

## INTRODUÇÃO

Neste trabalho, desenvolvemos a estrutura de backend utilizando a linguagem Python, aplicando princípios sólidos de orientação a objetos e diversos padrões de projeto para garantir um sistema modular, escalável e de fácil manutenção. O backend é responsável pela gestão de usuários e reservas, centralizando as regras de negócio, a segurança e a persistência de dados. O backend foi projetado para garantir a validação de dados essenciais, como e-mails e senhas, além de implementar segurança básica com o uso de hashing para armazenar credenciais de forma segura. Assim, garantimos não apenas a funcionalidade necessária, mas também boas práticas de desenvolvimento.

## OBJETIVO

O principal objetivo do backend deste sistema é oferecer uma base robusta e confiável para o gerenciamento de usuários e reservas, centralizando as operações críticas de forma segura e eficiente. Ele foi projetado para abstrair a complexidade das regras de negócio, proporcionando uma interface clara e bem definida para as funcionalidades essenciais, como cadastro de usuários, autenticação, criação e cancelamento de reservas. Além disso, buscou-se garantir a extensibilidade e a manutenção facilitada do sistema por meio da aplicação de boas práticas de desenvolvimento e padrões de projeto.

## METODOLOGIA

Para o desenvolvimento do backend deste sistema, foi adotada uma abordagem orientada a objetos, aliada à aplicação de diversos padrões de projeto, como Factory Method, Singleton, Strategy e Decorator. Essa escolha metodológica teve como foco principal garantir a modularidade, a escalabilidade e a reutilização de código, facilitando futuras evoluções e manutenções. Com sua execução inicial sendo realizada diretamente pelo próprio terminal, o sistema utiliza arquivos JSON para a persistência de dados, armazenando informações relacionadas a usuários e reservas.

## RESULTADOS E DISCUSSÕES

Com a implementação do backend, foi possível atingir os principais objetivos estabelecidos no início do projeto, como a gestão eficiente de usuários e reservas, garantindo segurança, modularidade e facilidade de manutenção. A adoção de padrões de projeto como Factory Method e Decorator contribuiu significativamente para a organização e expansão das funcionalidades, permitindo, por exemplo, a adição de logs ou a criação de perfis administrativos sem necessidade de alterar as classes principais.

Figura 1 - output do terminal

```
Escolha: 2
=== Lista de Usuários ===
ID: 6533163-9221-4d61-9190-373bccc268a9 Nome: tentivel | Email: tentivel@gmail.com | Tipo: comum | Status: Ativo
ID: 8803c0d8-6322-4a09-9319-3d9c6d69ffef Nome: tentativo | Email: tentativo@gmail.com | Tipo: comum | Status: Ativo
ID: 20f6d7bd-6f48-4c45-8782-8ba6e4e73385 Nome: Raquel fda | Email: dnjesj@gmail.com | Tipo: comum | Status: Ativo
ID: b18016c2-ee43-4dca-9337-8d6c7d2a8d2c Nome: teste terminal | Email: terminal@gmail.com | Tipo: comum | Status: Ativo
ID: 11c718c-ch19-4bd3-bdcb-f7843fa75edc Nome: 123 | Email: 123@gmail.com | Tipo: comum | Status: Ativo
ID: 47f9a6e9-386e-4d3b-9388-870224f6ac37 Nome: a | Email: teste@gmail.com | Tipo: comum | Status: Ativo
ID: 52e77888-375b-45c5-b565-93508d3d15fd Nome: testeADM | Email: ADM@gmail.com | Tipo: admin | Status: Ativo
ID: ada4b731-41d7-4e09-a595-675ea3937877 Nome: test3 | Email: test@gmail.com | Tipo: comum | Status: Ativo
ID: 9ac222ab-4e3e-4d66-8968-580896ee3a09 Nome: test03 | Email: test@gmail.com | Tipo: comum | Status: Ativo
ID: f495db2b-6dc5-4348-b458-2ad5b7f110be Nome: test02 | Email: test@gmail.com | Tipo: comum | Status: Ativo
ID: 2a377bac-c8d4-49db-9db3-bda72eb13580 Nome: ultimoTest | Email: ultimoTEST@gmail.com | Tipo: admin | Status: Ativo

=== Menu Geral ===
1. Cadastrar usuário
2. Listar usuários
3. Login
4. Listar reservas
5. Fazer reserva
6. Sair
Escolha: 4
=== Lista de Todas as Reservas ===
2025-05-10 | 08:00-10:00 | Sala: 101 | Usuário: Ana Oliveira | ana.oliveira@example.com | Matéria: Matemática | Status: DESATIVADA
2025-05-11 | 10:00-12:00 | Sala: 102 | Usuário: Bruno Costa | bruno.costa@example.com | Matéria: Física | Status: DESATIVADA
2025-05-12 | 14:00-16:00 | Sala: 103 | Usuário: Carla Souza | carla.souza@example.com | Matéria: Química | Status: DESATIVADA
2025-05-13 | 18:30-22:00 | Sala: 104 | Usuário: Daniel Rocha | daniel.rocha@example.com | Matéria: Biologia | Status: ATIVA
2025-05-14 | 13:30-15:00 | Sala: 105 | Usuário: Eduardo Lima | eduarda.lima@example.com | Matéria: História | Status: ATIVA
2025-05-15 | 15:00-17:00 | Sala: 106 | Usuário: Lucas Martins | lucas.martins@example.com | Matéria: Geografia | Status: ATIVA
2025-05-16 | 17:00-19:00 | Sala: 107 | Usuário: Marina Dias | marina.dias@example.com | Matéria: Filosofia | Status: ATIVA
2025-05-17 | 08:00-12:00 | Sala: 008 | Usuário: 123 | 123@gmail.com | Matéria: gabriel | Status: ATIVA
2025-05-09 | 09:00-09:30 | Sala: d07 | Usuário: 123 | 123@gmail.com | Matéria: teste2 | Status: ATIVA
2025-05-20 | 07:00-12:00 | Sala: d03 | Usuário: ultimoTest | ultimoTEST@gmail.com | Matéria: padrões | Status: ATIVA
```

Fonte: “os autores.”

## CONCLUSÃO

O desenvolvimento do backend deste projeto permitiu consolidar importantes conceitos de programação orientada a objetos, segurança e boas práticas de arquitetura de software. A implementação de um sistema de gerenciamento de usuários e reservas, com persistência em arquivos JSON e execução pelo terminal, atendeu aos requisitos iniciais, proporcionando uma solução funcional, modular e de fácil manutenção. Assim abrindo espaço para futuras melhorias e expansões para um sistema mais robusto.

## REFERÊNCIAS

MILLER, Frederic P.; VANDOME, Agnes F.; McBREWSTER, John (Ed.). **GRASP (object-oriented design)**. [S.l.]: Alphascript Publishing, 2011.

<sup>1</sup>Estudante do Curso Bacharelado em Ciência da Computação – e-mail: gabriel2332rodrigues@gmail.com;

<sup>2</sup>Professor do Curso Bacharelado em Ciência da Computação – e-mail: ricardo.ladeira@ifc.edu.br;