



## **Desafio SCI Sistemas Contábeis**

### **Programador Web Full Stack - Pleno/Sênior**

O Syndkos é um sistema da SCI Sistemas Contábeis que tem a ideia de criar uma plataforma única para conectar a administradora de condomínios aos síndicos, conselheiros e condôminos, proporcionando uma experiência de integração, agilidade e geração de confiança através da tecnologia. O intuito do sistema é atender as necessidades financeiras e gerenciais de seus clientes de forma intuitiva e moderna.

O dinamismo dos processos e o ganho de produtividade são proporcionais pelo uso das mais modernas tecnologias de desenvolvimento de sistemas disponíveis no mercado, garantindo segurança e mobilidade.

No Syndkos, uma das funcionalidades oferecidas aos síndicos e condôminos é o cadastro e reserva de ambientes, onde o síndico de um condomínio realiza o cadastro dos ambientes disponíveis no sistema junto com as datas e horários que podem ser reservados. Enquanto isso, o condômino pode efetuar reservas desses ambientes que podem simbolizar, por exemplo, a área de festas, a quadra de esportes ou a piscina de um condomínio.

Neste desafio, você irá criar uma versão simplificada do cadastro e reserva de ambientes.

#### **Critérios essenciais**

- 1) A solução deve ser desenvolvida em Python (sinta-se livre para utilizar um framework de sua preferência: Django, Flask, FastAPI, etc);
- 2) Utilizar um framework de front-end (React, Angular, Vue, etc);
- 3) Criar uma interface agradável e intuitiva (utilizando frameworks CSS, tais como: Bootstrap, Materialize, Material UI, etc);
- 4) Implementar testes automatizados;
- 5) Disponibilizar uma aplicação em forma de API para consulta e manipulação das reservas de ambientes;
- 6) Disponibilizar a documentação da API (Swagger, ReDoc, etc);
- 7) A solução deve possibilitar o cadastro, visualização, atualização e exclusão de uma reserva de ambiente;
- 8) Os dados devem ser salvos em um dos bancos de dados: Postgres, MySQL ou MariaDB;
- 9) A reserva de ambiente deve ter os seguintes campos:
  - a) Nome do ambiente;
  - b) CPF da pessoa responsável pela reserva;
  - c) E-mail da pessoa responsável pela reserva;
  - d) É pago uma taxa por uso do ambiente?
  - e) Valor cobrado (caso exista taxa por uso);
  - f) Capacidade máxima de pessoas no ambiente;



- g) Em quais dias da semana podem ser reservados os ambientes, sendo os dias possíveis:
- i) Domingo
  - ii) Segunda-feira
  - iii) Terça-feira
  - iv) Quarta-feira
  - v) Quinta-feira
  - vi) Sexta-feira
  - vii) Sábado
- 10) Quando um ambiente é reservado, vamos considerar que a reserva é válida para o dia inteiro;
- 11) O projeto deverá conter validações para que o CPF e e-mail informados sejam válidos;
- 12) O condômino não deve conseguir reservar em dias já reservados por outros condôminos;
- 13) Caso o ambiente seja reservado num dia da semana, não deverá ser possível reservar o próximo dia, pois devem ser realizadas as atividades de manutenção e limpeza do ambiente.
- a) Exemplo: se a área de festas foi reservada no sábado, no domingo não será possível realizar a reserva por nenhum morador, pois o dia posterior ficará reservado para limpeza do ambiente.
- 14) Deve ser possível realizar a busca de uma reserva do ambiente pelo CPF do responsável pela reserva (frontend);
- 15) As validações devem serem aplicadas tanto no frontend quanto na API disponibilizada pelo backend;
- 16) O projeto deverá ser disponibilizado em repositório online, como Github, Gitlab, etc;
- 17) O repositório deve conter um arquivo README explicando os requisitos para executar o projeto em ambiente local, como configurar ambiente e como executar localmente;
- 18) Utilize boas práticas no código, na estrutura e no repositório git;
- 19) Qualquer funcionalidade extra será bem vinda.

### **Critérios opcionais**

- 1) Adicione screenshots (imagens) do sistema ao final do arquivo README;
- 2) Demonstração de uso de pipelines para CI/CD para executar e exibir coverage dos testes unitários;
- 3) Utilização do ambiente de desenvolvimento em Docker;
- 4) Fazer deploy do projeto (Heroku, Surge.sh, Hostinger, Pythonanywhere, AWS, Google Cloud, etc) e disponibilizar a URL de acesso.