

Avaliação Desenvolvimento Full Stack - Pleno/Senior - Pontos



Data de emissão: 19/02/2024



Sumário

[Sumário](#)

[Introdução](#)

[Desafio 1](#)

[Observações](#)

Introdução

Neste documento propomos a avaliação para **Desenvolvimento Full Stack Pleno/Senior** na **Logique Sistemas**. A avaliação é composta por **um** desafio.

Na **Logique Sistemas**, utilizamos algumas tecnologias nas nossas soluções, sendo elas: *Java, Python, React JS, Angular, PostgreSQL*, etc. Sugerimos que utilize algumas destas tecnologias nas suas resoluções ou então fique a vontade para utilizar alguma que se sinta mais confortável.

Sinta-se a vontade para desenvolver e apresentar os resultados do desafio da forma que preferir.

Boa sorte e boa avaliação!

Desafio 1

Empresas com colaboradores (em regime CLT) fazem uso de ferramentas para registro de pontos de jornadas de trabalho para que tanto os colaboradores quanto a empresa tenham o controle das horas trabalhadas. Fazer esse acompanhamento manualmente é uma tarefa complexa e passível de erros. Além disso, ao longo do dia a dia, podem surgir uma série de imprevistos, como a entrada um pouco atrasada por algum colaborador, ou a necessidade de ausência parcial ou integral de outro colaborador. Estas situações, tornam o controle de jornadas ainda mais complexo.

Sendo assim, desenvolva um sistema de **gestão de pontos de jornada de trabalho**, onde os seguintes requisitos sejam atendidos:

- O sistema deve ser protegido por login e senha, e somente após autenticado o usuário poderá visualizar os dados;
- Os dados de usuários e pontos devem ser persistidos em banco de dados;
- Um ponto representa o registro de entrada ou saída de expediente durante a jornada de trabalho de um usuário (colaborador);
- O sistema deve ter suporte a dois tipos de regimes de jornadas de trabalho, são eles:
 - Regime de 6 horas contínuas de trabalho, sem previsão de pausas;
 - Regime de 8 horas de trabalho, com previsão de pausa mínima de 1 hora para almoço;
- O sistema deve ter flexibilidade para aceitar entradas de registros de pontos em qualquer horário, dada a situações de imprevistos, atrasos ou cargas extras de trabalho;
- Exemplos de registros de pontos de jornada de trabalho válida para um regime de 8 horas:
 - 08:28 - Horário de início de expediente,
 - 12:15 - Horário de saída para pausa para almoço,
 - 13:15 - Horário de retorno de pausa para almoço,
 - 17:28 - Horário de saída de expediente.

- O sistema deve ter suporte para múltiplos registros de pontos em um dia, não se restringindo ao limite de quatro pontos. Por exemplo: 08:28 (início de expediente) → 12:15 (pausa para almoço) → 13:15 (retorno da pausa para almoço) → 15:30 (saída esporádica) → 16:30 (retorno da saída esporádica) → 18:28 (fim de expediente);
- O sistema deve ter suporte para dois tipos de usuários: Usuário Administrador e Usuário Comum;
- O usuário administrador terá permissão para cadastro de novos usuários (usuário comum);
- Ao cadastrar um novo usuário comum, o administrador deverá informar o tipo de regime de jornada de trabalho daquele colaborador;
- O usuário comum não poderá cadastrar novos usuários;
- O usuário comum terá acesso as seguintes funcionalidades:
 - Registrar ponto: O usuário poderá informar a data/hora para registro de um novo ponto de trabalho na jornada;
 - Resumo de jornada do dia atual: Com base nos pontos cadastrados para o dia, o usuário poderá verificar quais pontos foram contemplados, bem como se a jornada prevista para o dia está completa ou não (entende-se completa quando a duração do expediente com base nos pontos atende o regime de jornada de trabalho do colaborador);
 - Previsão para completar jornada: O sistema deverá apresentar, com base nos pontos, resumo de jornada e regime de jornada do colaborador, a quantidade de horas restantes para completar a jornada do dia;
 - Horas excedidas da jornada: O sistemas deverá apresentar, com base nos pontos, resumo de jornada e regime de jornada do colaborador, a quantidade de horas excedidas (extras) na jornada do dia;

Fique a vontade para aplicar melhorias e customizações que preferir no sistema. Apresente e descreva-os durante a apresentação. ;)

Observações

- O projeto deverá ser entregue com o link do repositório no Github ou Gitlab contendo o projeto versionado e acessível publicamente;
- O projeto deverá vir documentado no repositório (arquivo README.md) contendo as versões das linguagens, principais bibliotecas/frameworks, como executar o projeto, funcionalidades implementadas e funcionalidades idealizadas, mesmo que não desenvolvidas.
- Implemente módulos de testes unitários, principalmente para validação de fluxos de processamento das datas dos pontos;
- Se possível, disponibilize o sistema desenvolvido em algum serviço *cloud* e que esteja acessível para avaliação prévia;
- Junto às documentações, apresente diagrama de entidade/relacionamento (DER) que reflete a arquitetura de dados aplicada na solução;
- Boas práticas de Git, testes unitários e qualidade de código serão considerados na avaliação.