

Projeto Trilha Certa

O Centro de Ensino Médio precisa automatizar a confirmação de matrícula dos alunos pelas trilhas que eles escolherem. Com o novo ensino médio, a escola precisa oferecer trilhas. Das 19 trilhas possíveis, ela escolheu 4:

1. ENEGRE-SER, cujo enfoque é social;
2. DINHEIRO NA MÃO É VENDEVAL, que trata de finanças pessoais;
3. ADMIRÁVEL MUNDO NOVO, que trata das tecnologias digitais; e
4. AGROECOLOGIA cujo título já diz tudo.

A escola possui 8 turmas do 1º ano, de A a H, todas no turno vespertino. No 1º ano, os alunos têm uma visão do que se trabalha em cada trilha. No final do ano, mais precisamente no mês de novembro, eles têm que escolher uma trilha para se matricular no 2º ano. Mas existe uma restrição. No 2º ano, existem 4 turmas no turno matutino (A, B, C e D) e 4 no vespertino (E, F, G e H). As turmas A e E são da trilha 1, B e F da trilha 2, C e G da trilha 3 e D e H da trilha 4. Todas as turmas têm que ter o mesma quantidade de alunos, exceto quando o total de alunos não for múltiplo de 8. Exemplo: Se forem 240 alunos, cada turma terá exatamente 30 alunos. Se forem 208, cada turma terá 26 alunos. Agora, se forem 220 alunos, 4 turmas terão exatamente 28 alunos e 4 turmas terão exatamente 27 alunos.

Cada turma só pode ter no máximo 30 alunos, totalizando 240 alunos nas 8 turmas! Não pode ter overbooking!

Histórias do usuário:

- A escola cadastra os CPF dos alunos do 1º ano. Só os CPF cadastrados conseguem entrar no sistema. Existirão no máximo 240 CPF.
- Um aluno entra no sistema usando seu CPF. Em seguida, o sistema mostra apenas os turnos com vagas disponíveis para matrícula. Caso um turno esteja lotado, indica o turno que está lotado e mostra o turno disponível para o aluno escolher. O aluno escolhe o turno que quer cursar. Em seguida o sistema mostra apenas as trilhas com vagas disponíveis para matrícula. Caso algumas trilhas estejam com todas as vagas ocupadas, indica isto e mostra as trilhas com vagas disponíveis para o aluno escolher. O aluno escolhe a sua trilha e o sistema confirma a matrícula.
- Ao final do período de matrículas, o sistema só manda uma mensagem de prazo encerrado, não permitindo o acesso dos alunos.
- Caso um aluno que já escolheu entre com seu CPF, o sistema deve mostrar a escolha dele, que não pode ser alterada. Após o prazo da matrícula o sistema não pede mais o CPF, só mostra a mensagem de prazo encerrado.
- A qualquer momento o administrador do CEM pode consultar o relatório das matrículas, inclusive após o período das matrículas.

Requisitos não funcionais:

- O sistema será instalado em um notebook com 4 GB RAM e 100 GB de HD com acesso à internet.
- O serviço deve ser oferecido pela internet por meio de um servidor HTTP.
- A solução deve prever o maior paralelismo possível entre os acessos dos alunos para a escolha. A solução trivial seria serializar todos os alunos em uma única fila, de acordo com a chegada, mas isso pode acarretar um tempo de espera muito longo para os últimos da fila, caso os alunos entrem quase ao mesmo tempo. Por isso, o sistema deve permitir que um número máximo de alunos possa escolher a sua trilha ao mesmo tempo, sem ultrapassar a quantidade máxima de alunos em cada turmas.
- Para evitar uma longa espera, as vagas mostradas para os alunos escolherem terão um tempo de validade, por volta de 30 segundos. Após este prazo, o aluno terá que recomençar o processo de matrícula novamente. Neste caso, o sistema deve informar

que as ofertas não são mais válidas e aluno deve recomeçar a escolha desde o início, ou seja, escolhendo o turno para ver as trilhas ainda disponíveis.

- Caso a trilha pretendida no turno escolhido esteja lotada, o aluno pode trocar o turno, liberando as vagas mostradas e ver se a trilha do outro turno tem vagas, que ficam reservadas por 30 segundos, para ele escolher uma trilha.
- Os dados devem ser persistidos apenas em arquivos do tipo texto. Usar estrutura de dados interna aos arquivos mais adequada ao problema. Não pode ter condições de corrida no acesso a cada registro do arquivo e deve-se ter o maior paralelismo possível.
- A solução deve ser implementada como um processo Web (servidor) que recebe várias requisições em paralelo. Usar um framework web full stack no backend baseado em Python. O frontend pode ser React. A comunicação pode ser JSON/REST.

Modelar e implementar a solução evitando condições de corrida, deadlock e starvation com o maior paralelismo possível. Upload da solução no Moodle e apresentação da solução para a turma. Deverá haver 3 sprints. Ao final de cada sprint o trabalho será avaliado parcialmente.