

Sistemas Simples para Problemas Reais

1. Nome do grupo e integrantes :

Equipe 3:

Erick, Vitor, João H., Maria E. e Felipe R.

2. Nome do sistema/empresa fictícia :

CIAOE (Centro de Integração de Aprendizes com Oportunidades de Efetivação).

3. Título do problema

INTEGRAÇÃO DO PROGRAMA JOVEM APRENDIZ COM OPORTUNIDADES DE EFETIVAÇÃO.

4. Descrição do problema real:

Empresas enfrentam desafios na implementação do programa jovem aprendiz, principalmente com a integração dos jovens no ambiente de trabalho. A falta de experiência e a dificuldade em encontrar jovens com o perfil desejado são obstáculos comuns. No entanto, o programa oferece benefícios como a inclusão social e a formação de futuros profissionais qualificados.

5. Como é tratado hoje :

O Senai desempenha um papel crucial na implementação de programas de jovem aprendiz e estágios, facilitando a inserção de jovens no mercado de trabalho e contribuindo para o desenvolvimento profissional. A instituição atua como ponte entre a formação teórica e a prática, oferecendo cursos e experiências práticas em empresas, o que aumenta as chances de efetivação dos jovens.

6. Solução proposta:

O CIAOE tem como principal objetivo o preparo técnico e prático dos aprendizes para o mercado de trabalho, propondo aprendizagens com histórico de notas em cada disciplina. No entanto, além de auxiliar os jovens aprendizes com a inserção na empresa, apresenta um histórico de notas e competências com indicação de qual vaga o futuro funcionário se encaixa, conforme as soft skills e hard skills, gerando um perfil profissional do mesmo. Desse modo, facilita a busca da empresa podendo analisar e discutir qual aluno se ingressaria melhor nas vagas disponíveis.

7. Explicação do código/lógica usada:

O código tem 4 arquivos java: GestorAlunos.java(Main onde temos o menu), aluno.java (para a criação do objeto dos alunos), vagas.java(Classe abstrata para possíveis vagas aos alunos) e Tecnico.java(Classe estendida de vagas.java, apenas um exemplo de como funcionaria em grande escala).

aluno.java:

Aqui temos todas as funções usadas no mains
public void cadastrar(): Responsável pelo cadastro dos alunos, que são adicionados no

ArrayList alunos, que tem como características seu nome e suas competências.

public void adicionarNotas(): Responsável por adicionar as notas dos alunos em determinada matéria, começa primeiro analisando se alunos.isEmpty(), caso seja verdade ele para. Após isso pergunta o aluno, e qual das matérias, que são armazenadas em MATERIAS[]. Após isso é usado um for loop para percorrer todos os espaços de determinada matéria na matriz onde são armazenadas double[][] notas, sendo o primeiro o espaço das matérias e o segundo as notas

public void adicionarCompetencias(): Mesma lógica do anteriores, só que com competências, ele verifica, pergunta, e através do JOptionPane ele pergunta quais as competências do aluno indefinidamente até o usuário digitar 0 ou sair

public void mostrarNotas(): Verifica, pergunta o aluno, passa um for loop através das notas de cada matéria. Para demonstrar visualmente foi usado String mensagem, onde através da concatenação foi adicionado através dos for loop, o for loop “externo” indica a matéria, e o for loop “interno” as notas dentro dessas matérias.

public void vagas(): Verifica, pergunta o aluno, e usa a função em Tecnico.java, que leva como atributos a média (não utilizada) e as competências do aluno, a partir daí ela analisa comparando se o que é preciso para vaga está presente no aluno. Há apenas uma vaga de exemplo.

GestorAlunos.java:

Basicamente o menu, onde são importadas as funções de aluno.java

vagas.java e Tecnico.java:

vagas.java é o modelo que todas as vagas seguem, e Tecnico.java é o exemplo usado no código para demonstrar como isso funcionaria. Foi usada uma classe abstrata e não apenas um polimorfismo pois dependendo da empresa ela teria diferentes maneiras de avaliar os alunos, então por conta de flexibilidade foi usada uma classe abstrata.

8. Melhorias futuras:

O estudo das competências antes de indicar a vaga. A consulta do aluno e a exibição da vaga pelas capacidades dele. Consulta de vagas pelas habilidades exigidas. O controle de indicação por vaga, mostrando um gráfico de quais competências estão em déficit. E simplificar os métodos e listas.