

Avaliação Diagnóstica

A avaliação diagnóstica é uma etapa fundamental no processo de ensino-aprendizagem, pois permite ao professor identificar o conhecimento prévio dos alunos e as lacunas existentes antes do início de uma nova unidade curricular. Essa forma de avaliação possibilita o planejamento de estratégias pedagógicas mais eficazes e personalizadas, ajustando o ritmo e a abordagem das aulas conforme as necessidades individuais ou coletivas da turma.

No contexto formativo, a avaliação diagnóstica também oferece aos alunos uma oportunidade de refletir sobre seu próprio aprendizado, promovendo maior conscientização sobre suas dificuldades e pontos fortes. Dessa forma, ela atua como uma ferramenta de base para o desenvolvimento contínuo, garantindo que o ensino seja mais direcionado e significativo, aumentando as chances de sucesso escolar.

A avaliação diagnóstica é um dos aspectos presentes no processo de formação profissional com base em competências. Neste sentido cabe ao docente observar as funções da avaliação. Considerando a Metodologia Senai de Educação Profissional - MSEP, a avaliação da aprendizagem contempla 3 funções da avaliação na perspectiva do desenvolvimento de competências, da qual, uma delas é a função diagnóstica. Portanto, observa-se que a avaliação precisa ser planejada buscando mensurar, acompanhar, identificar, medir, investigar e analisar o aprendizado do aluno, assim como correlacionar com as competências previstas no itinerário formativo.

A avaliação também desempenha um papel de retroalimentar os processos de ensino e aprendizagem, para que se necessário, seja redirecionado ou corrigindo possíveis desvios de rota no processo de formação das capacidades profissionais. Estas funções da avaliação, de acordo com a Metodologia Senai de Educação Profissional - MSEP, são as seguintes: função diagnóstica, formativa e somativa. Especificamente sobre a função diagnóstica, tem-se que:

Busca consultar aquilo que os Alunos aprenderam antecipadamente e ao longo do curso, a respeito de um determinado conteúdo formativo, consentindo ao Docente identificar as necessidades de aprendizagem, ou seja, diagnosticar a condição do Aluno, considerado o contexto já consolidado e trazido de experiências anteriores (MSEP, 2019, p. 125).

Assim, ao planejar uma unidade curricular, é essencial que o professor tenha uma visão clara sobre o conhecimento prévio dos alunos, pois isso impacta diretamente no processo de ensino-aprendizagem. No contexto de um curso técnico em Desenvolvimento de Sistemas, é especialmente relevante identificar se os estudantes compreendem conceitos fundamentais de unidades curriculares prévias.

1. Contextualização

Neste trabalho foi escolhido as seguintes unidades curriculares para realizar este estudo: Lógica de Programação e Programação de Aplicativos. Isto significa que o professor está ministrando no momento atual a unidade curricular de Programação de Aplicativos e procura diagnosticar conhecimentos prévios da unidade curricular de Lógica de programação.

Será feito um levantamento de conceitos de modo geral, com uma ferramenta digital de coleta de informações, da unidade curricular de lógica de programação que são necessários para a unidade curricular de programação de aplicativos.

Posteriormente é feita uma aplicação de prototipação de uma tela de interface de usuário, que será empregada para um debate acerca dos conceitos prévios. Neste caso mais específico, a capacidade técnica a ser investigada será a de “Integrar banco de dados por meio da linguagem de programação” prevista no itinerário formativo do curso técnico de desenvolvimento de sistemas, localizado na coluna D, linha 240 do itinerário formativo. Neste caso, um dos conhecimentos prévios que os alunos precisam ter está relacionado aos conceitos de tipos de dados e de variáveis usados em linguagem de programação. No itinerário formativo de lógica de programação, estes conceitos aparecem na unidade de Lógica de Programação como fundamentos técnicos e científicos, descrito como “Identificar estruturas de dados para construção do algoritmo”, localizado na coluna C, linha 303 o itinerário formativo.

Portanto, será investigado o conhecimento prévio dos alunos relacionados aos conceitos gerais de lógica de programação e um estudo específico de tipos de dados e de variáveis, pois para gravar os dados em um banco de dados é preciso determinar o seu tipo.

2. Justificativa

A avaliação diagnóstica é um componente essencial no planejamento pedagógico, permitindo que os professores identifiquem o conhecimento prévio dos alunos e ajustem as práticas educativas de acordo com as necessidades da turma. Através de uma avaliação eficaz, o professor pode detectar quais conhecimentos e capacidades já foram consolidados e quais ainda precisam ser reforçados. No contexto do ensino técnico, especialmente em áreas como a programação, a avaliação diagnóstica facilita a transição entre unidades curriculares, assegurando que os alunos possuam as bases teóricas e práticas necessárias para acompanhar novos conteúdos.

Para facilitar essa coleta de dados e tornar o processo mais interativo e envolvente, diversas ferramentas digitais podem ser utilizadas. Elas oferecem ao professor a oportunidade de avaliar os alunos de maneira ágil e eficiente, promovendo a participação ativa dos estudantes e fornecendo dados em tempo real sobre o desempenho da turma. Essas ferramentas digitais, possibilitam que o professor crie um ambiente descontraído, incentivando o aluno a demonstrar seu conhecimento prévio, sem a pressão típica de uma prova formal. O uso dessas ferramentas é especialmente relevante em programação, onde a prática constante e a aplicação dos conceitos adquiridos são fundamentais para o aprendizado.

No contexto de um curso técnico em Desenvolvimento de Sistemas, é especialmente relevante identificar se os estudantes envolvidos compreendem conceitos fundamentais de unidades curriculares anteriores, de modos geral, assim como noções de variáveis vistas em Lógica de Programação, para aplicar esses conhecimentos em unidades mais avançadas, como Programação de Aplicativos.

A utilização de ferramentas de diagnóstico facilita o levantamento desse conhecimento prévio e possibilita um ensino mais direcionado e eficaz, sendo que existem diversas ferramentas digitais que podem auxiliar o professor na coleta de dados sobre o nível de conhecimento dos alunos. Essas ferramentas permitem, de maneira interativa e ágil, medir o entendimento dos alunos em relação a conceitos abordados em unidades anteriores.

As ferramentas que foram pesquisadas são:

Kahoot: é uma ferramenta muito utilizada em ambientes educacionais para a criação de quizzes interativos. Seu formato de gamificação envolve os alunos em

atividades dinâmicas, nas quais eles respondem perguntas em tempo real. Com o Kahoot, o professor pode criar questionários sobre variáveis e outros conceitos de programação, obtendo feedback imediato sobre o nível de conhecimento da turma. A plataforma também permite a criação de quizzes com múltiplas escolhas, verdadeiro ou falso, e até perguntas abertas, proporcionando flexibilidade para diferentes estilos de avaliação diagnóstica. Além disso, o Kahoot oferece relatórios que auxiliam na identificação de lacunas de conhecimento entre os alunos.

Quizizz: é outra plataforma de quizzes online que permite ao professor diagnosticar o conhecimento dos alunos de maneira interativa. Uma das vantagens do Quizizz é que os quizzes podem ser realizados de forma assíncrona, ou seja, o aluno pode responder no seu próprio tempo, ao contrário do Kahoot que geralmente é usado em tempo real. O Quizizz também fornece relatórios sobre o desempenho individual e coletivo dos alunos, permitindo ao professor identificar com precisão as áreas onde os alunos precisam de mais reforço. No contexto da unidade curricular de Programação de Aplicativos e sua relação com a unidade de Lógica de Programação, o Quizizz pode ser utilizado para revisar conceitos como tipos de variáveis, estruturas de dados e outros tópicos de lógica de programação, preparando o aluno para aplicar esses conceitos no desenvolvimento de soluções reais.

Google Forms: é uma ferramenta versátil para criar questionários e avaliações diagnósticas. Com ela, o professor pode elaborar quizzes sobre os tipos de variáveis e outros conceitos fundamentais da programação de maneira simples e eficiente. Os formulários permitem a personalização de perguntas em diversos formatos (múltipla escolha, dissertativa, listas suspensas), além de gerar relatórios automáticos com gráficos e tabelas que facilitam a análise dos resultados.

Mentimeter: é uma plataforma interativa que permite ao professor criar apresentações dinâmicas com perguntas, enquetes e quizzes em tempo real. Assim como o Kahoot, ele também utiliza o elemento de gamificação, mas com foco em apresentações mais estruturadas, onde os alunos podem responder a perguntas ao longo de uma explicação. Isso permite ao professor diagnosticar o conhecimento dos alunos enquanto apresenta um conteúdo, ajustando o ritmo da aula conforme o nível de compreensão observado.

Existem outras ferramentas que podem ser utilizadas, que podem ser encontradas como Plickers e Edpuzzle, mas que não serão abordadas aqui. Além destas ferramentas o professor pode lançar mão do seu conhecimento da unidade

curricular de Programação de Aplicativos, que em algum momento da unidade curricular, os alunos podem vir a desenvolver ou programar um formulário de coleta de dados, como por exemplo o cadastro de um cliente. Neste caso, a unidade curricular pode se utilizar de ferramentas de projeto de interface que vão estar associadas aos conceitos de tipos de dados e de variáveis usados em linguagem de programação. Neste caso, o professor pode explorar ferramentas como o Figma.

O Figma é uma ferramenta de design colaborativo amplamente usada no desenvolvimento de interfaces gráficas. No contexto educacional, o Figma pode ser adaptado como uma ferramenta diagnóstica ao solicitar que os alunos projetem telas ou protótipos de aplicativos que envolvem conceitos programáticos, como variáveis e estruturas de dados. Um exemplo prático seria pedir aos alunos que criem uma tela de cadastro de clientes, contendo campos como nome, telefone, CPF, email, e endereço completo. Com essa atividade, o professor consegue avaliar se o aluno compreende os tipos de variáveis necessárias para armazenar esses dados e como elas seriam representadas no código de um aplicativo. O Figma oferece um ambiente visual que favorece a criatividade e a aplicação prática dos conceitos, sendo uma excelente ferramenta para verificar o conhecimento prévio de forma mais aplicada.

3. Lista de Capacidades

Para o contexto deste trabalho serão observados os conhecimentos gerais associados às capacidades de lógica de programação que se encontram no itinerário formativo do curso técnico de desenvolvimento de sistemas que são:

- Aplicar lógica de programação para resolução dos problemas.
- Utilizar técnicas de abstração para resolução de problemas.
- Interpretar a simbologia das representações gráficas para definição do fluxo do algoritmo.
- Identificar estruturas de dados para construção do algoritmo.
- Utilizar expressões aritméticas, relacionais e lógicas para codificação do algoritmo.
- Codificar algoritmos na resolução de problemas.
- Aplicar técnica de ordenação e busca de dados para construção de algoritmo.
- Identificar padrão de nomenclatura de comentários para documentação do código fonte.
- Utilizar as estruturas de controle e repetição adequadas à lógica dos algoritmos.
- Utilizar padrões de nomenclatura e convenções de linguagem na codificação de algoritmos.

Além disso, foi escolhida a capacidade técnica de “Integrar banco de dados por meio da linguagem de programação” prevista no itinerário formativo do curso técnico de desenvolvimento de sistemas, para uma breve imersão, onde é avaliado o conhecimento prévio que está relacionado aos conceitos de tipos de dados e de variáveis usados em linguagem de programação, e que tem relação com a unidade de Lógica de Programação, onde aparece no itinerário formativo do curso como fundamentos técnicos e científicos, descrito como “Identificar estruturas de dados para construção do algoritmo”.

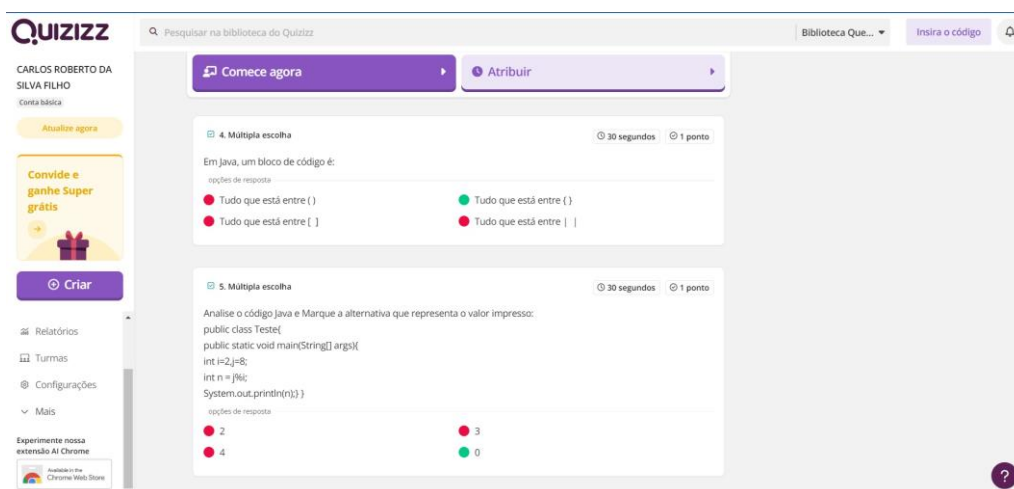
4. Instrumento Avaliativo

Como instrumento para realizar um diagnóstico mais descontraído com os alunos das capacidades previstas no itinerário formativo e evidenciar os conhecimentos prévios que os alunos possuem foi selecionado, dentre as opções pesquisadas, o Quizizz.

Com este instrumento foram criadas 20 questões no formato de múltipla escolha que buscam fornecer uma visão geral do conhecimento dos estudantes sobre as capacidades de Lógica de Programação que, de certo modo, em geral, são necessárias para a unidade curricular de Programação de Aplicativos.

Deve-se observar que para visualizar estas questões será necessário ter um perfil na plataforma. Para reforçar a evidência este desenvolvimento foi feito um “print” de tela das perguntas que aparecem no [link](#).

Figura 1 - Tela de questões referente ao levantamento dos conhecimentos prévios no Quizizz.



A figura 1 mostra 2 questões com o uso da linguagem de programação Java que busca levantar alguns conhecimentos sobre lógica de programação e sobre conhecimentos adquiridos, como por exemplo, tipos de variáveis e o resultado de uma operação em Java.

O uso de ferramentas digitais para essa finalidade é uma alternativa satisfatória. Plataformas interativas, como Quizizz, permitem avaliações rápidas e dinâmicas, tornando o processo mais engajante e acessível tanto para alunos quanto para professores.

5. Protótipo

Em termos de como empregar o Figma para este propósito de levantar e/ou identificar o conhecimento prévio dos alunos em relação a tipos de dados e de variáveis, o professor também pode criar uma tela de exemplo e debater o tema com os alunos os tipos de dados que seriam necessários para armazenar as informações presentes na tela de interface da aplicação. Buscando demonstrar este caso, foi criada uma tela de interface exemplo com alguns campos, onde os alunos precisam justificar os tipos de variáveis e tipos de dados a serem utilizados, e que, estes posteriormente poderiam ser armazenados em um banco de dados. A figura 2 a seguir, mostra esta tela exemplo.

Figura 2 - Tela de cadastro do cliente



O protótipo da tela de cadastro do cliente apresenta uma interface limpa e organizada. No topo, há uma barra de navegação azul escura com um ícone de seta para trás, o texto "Perfil Cliente" e um ícone de menu hambúrguer. Abaixo, o formulário é dividido em seções: "Cliente" com campos para "Nome completo" e "CPF"; "Contatos" com campos para "E-mail" e "Telefone ou celular"; e "Endereço" com campos para "Logradouro", "Número" e "Complemento" (em uma linha), "Cidade", "UF" (menu suspenso) e "CEP" (em uma linha). No final, há dois botões: "Concluir cadastro" em azul escuro e "Cancelar" em branco com contorno azul.

14:32 85%

< Perfil Cliente

Cliente

Nome completo

CPF

Contatos

E-mail

Telefone ou celular

Endereço

Logradouro

Número Complemento

Cidade

UF ▼ CEP

Concluir cadastro

Cancelar

Na figura 2 pode-se observar que vários tipos de dados distintos são solicitados no cadastro do cliente. Neste caso, cabe ao professor debater e conversar com os alunos sobre os tipos de dados que cada campo precisa armazenar e como estes dados serão direcionados para um banco de dados, quando este for integrado à aplicação que o aluno vai implementar.

6. Conclusão

O uso de ferramentas digitais como Quizizz e de protótipo como o Figma, além de outras inovações como Kahoot, Google Forms, Mentimeter, Plickers e Edpuzzle, possibilita um diagnóstico eficaz e dinâmico do conhecimento prévio dos alunos. Essas ferramentas não apenas facilitam a vida do professor, como também tornam o processo de avaliação mais envolvente e interativo para os estudantes.

A utilização dessas plataformas em um curso técnico de Desenvolvimento de Sistemas, especificamente para revisar conceitos de variáveis e lógica de programação, é uma excelente estratégia para garantir que os alunos estejam preparados para aplicar esses conhecimentos em unidades mais avançadas, como o desenvolvimento de aplicativos.

Assim, o professor consegue identificar com precisão as áreas que necessitam de reforço, personalizando o ensino para atender às necessidades de cada aluno e maximizando o potencial de aprendizagem.

7. Referências

LUCKESI, Cipriano Carlos. Avaliação da aprendizagem escolar: estudos e proposições. 22. ed. São Paulo: Cortez, 2011.

HAYDT, Regina Célia Cazaux. Avaliação do processo ensino-aprendizagem. 6. ed. São Paulo: Ática, 2006.