SISTEMA DE MONTAGEM DE PROVA COM BANCO DE QUESTÕES E CORREÇÃO AUTOMATIZADA COM VISÃO COMPUTACIONAL

Jean Machado Brasil, Vitor Doring Leitão, Lucas Baruel Cestaro, Alberson Wander Sá dos Santos, Hélio Lourenço Esperidião Ferreira.

Colégios UNIVAP Unidade Centro, São José dos Campos-SP, Brasil, jeanmachadobrasil@gmail.com;

Resumo

Este trabalho apresenta o desenvolvimento de um sistema web para montagem, organização e correção automatizada de avaliações escolares, com o objetivo de reduzir o tempo despendido por docentes em atividades extraclasse e minimizar falhas associadas a processos manuais. A solução foi estruturada em arquitetura modular, com interface desenvolvida em HTML, CSS e Vue.js, backend em Python com Flask, suporte do Node.js, persistência de dados em MongoDB e rotinas de Visão Computacional implementadas com OpenCV. O sistema emprega QR Code individual para associar cada avaliação ao aluno, ao modelo de prova e ao respectivo gabarito. O protótipo integra banco de questões, montagem de provas, validação pelo processo pedagógico, impressão e fluxo de correção, além de apresentar uma interface com edição enriquecida e pré-visualização do documento. Os resultados demonstram a viabilidade da integração das tecnologias utilizadas e indicam potencial para aumentar a padronização, a confiabilidade e a eficiência dos processos avaliativos em instituições de ensino.

Palavras-chave: Correção automatizada. OpenCV. QR Code. Banco de dados não relacional. Visão computacional.

Área do Conhecimento: Ciências Exatas e da Terra - Ciência da Computação.

Introdução

As transformações no ambiente educacional ampliaram as atividades realizadas pelos docentes para além do período em sala de aula, incluindo planejamento pedagógico, elaboração de materiais e processos avaliativos. Barbosa (2021) observaram que aproximadamente 97% dos professores participantes de seu estudo dedicavam tempo adicional às atividades profissionais além da carga horária prevista. Entre as tarefas que contribuem para essa ampliação da jornada estão a elaboração, a aplicação e a correção de avaliações. Nesse contexto, sistemas automatizados de correção de provas objetivas constituem uma alternativa para diminuir o tempo empregado nessas atividades (Fileno, 2025), enquanto as Tecnologias da Informação e Comunicação (TIC) permitem organizar dados, automatizar rotinas e tornar o fluxo de trabalho mais eficiente (Pinto; Molina; Paletta, 2022).

O QR Code (Quick Response Code) oferece alta capacidade de armazenamento, leitura em diferentes direções e mecanismos de correção de erros, características que favorecem a identificação automática de documentos (Tiwari, 2016). Paralelamente, técnicas de Visão Computacional possibilitam interpretar imagens e reconhecer padrões, sendo aplicáveis à leitura de cartões-resposta e à automatização de avaliações objetivas (Menoncin; Antonello, 2017). A integração dessas tecnologias permite relacionar uma prova impressa aos dados digitais correspondentes e reduzir etapas manuais suscetíveis a falhas.

Dessa forma, o objetivo deste trabalho é desenvolver um sistema computacional capaz de integrar a montagem, o gerenciamento, a validação, a impressão e a correção automatizada de provas objetivas. A proposta utiliza um banco de questões compartilhado, identificação individual por QR Code e processamento de imagens para reconhecer marcações, buscando aumentar a confiabilidade do processo avaliativo, reduzir erros humanos e otimizar o tempo de trabalho dos docentes em cenários com grande volume de avaliações.

Metodologia

O desenvolvimento adotou uma arquitetura computacional modular. No backend, utilizou-se Python com o microframework Flask, estruturado segundo o padrão MVC (Model-View-Controller) e organizado em uma API REST. A camada de serviço concentra as regras de negócio, enquanto o padrão DAO (Data Access Object) é responsável pela comunicação com o MongoDB, separando o acesso aos dados das demais funcionalidades da aplicação. O controle de autenticação e autorização é realizado pelo próprio backend, diferenciando as permissões de professores e do processo pedagógico, sem a utilização de JWT.

Na camada de apresentação, foram empregados HTML, CSS e Vue.js, permitindo a construção de interfaces responsivas e componentes reutilizáveis. O Node.js foi integrado como suporte às operações assíncronas e à manipulação de arquivos. Para a persistência, utilizou-se o MongoDB, banco de dados não relacional orientado a documentos, escolhido pela flexibilidade na representação das estruturas de questões e avaliações (Chorodow, 2019).

O fluxo de correção utiliza duas tecnologias principais. O QR Code identifica individualmente cada avaliação e permite associá-la ao aluno, à versão da prova e ao respectivo gabarito (Tiwari, 2016). Em seguida, rotinas de Visão Computacional implementadas com a biblioteca OpenCV processam as imagens digitalizadas, identificando os campos de resposta e as marcações realizadas pelos estudantes (Kaehler; Bradski, 2017; Menoncin; Antonello, 2017).



A modelagem funcional definiu dois perfis principais de acesso. O Professor pode autenticar-se, cadastrar, editar e excluir questões próprias, consultar e filtrar o banco geral, montar avaliações e encaminhá-las ao processo pedagógico. O Processo Pedagógico concentra a validação das provas, o gerenciamento de usuários e avisos, a importação de arquivos escaneados, o acionamento da correção

automatizada, a impressão e a exportação dos resultados. A navegação foi estruturada para separar os escopos de cada perfil e centralizar as funções relacionadas a questões, provas e configurações.

Como complemento ao desenvolvimento, foi elaborada uma estimativa do custo de capital humano. O valor da hora técnica de cada função foi obtido a partir do salário mensal médio, convertido em valor diário e dividido por uma jornada de oito horas. Esse valor foi multiplicado pelo total de horas dedicadas ao projeto. A estimativa resultou em R$ 184,92 para a atividade de análise de sistemas, R$ 19.250,00 para desenvolvimento backend e R$ 7.124,04 para desenvolvimento frontend, totalizando R$ 26.558,96.

Tabela 1 - Estimativa de custo do desenvolvimento do sistema.

Função	Salário por hora	Horas	Custo
Analista de Sistemas	R$ 15,41	12	R$ 184,92
Desenvolvedor Back End Júnior Flask	R$ 16,60	1155	R$ 19.250,00
Desenvolvedor Front End Júnior Vue	R$ 15,42	462	R$ 7.124,04
Total			**R$ 26.558,96**

Fonte: Autores (2026).

Resultados

Para fins de validação técnica do protótipo, todas as simulações de fluxo de correção foram conduzidas utilizando gabaritos mock e dados de estudantes fictícios, sem a coleta de informações pessoais ou envolvimento de seres humanos reais, o que isenta esta pesquisa de submissão ao Comitê de Ética em Pesquisa (CEP).

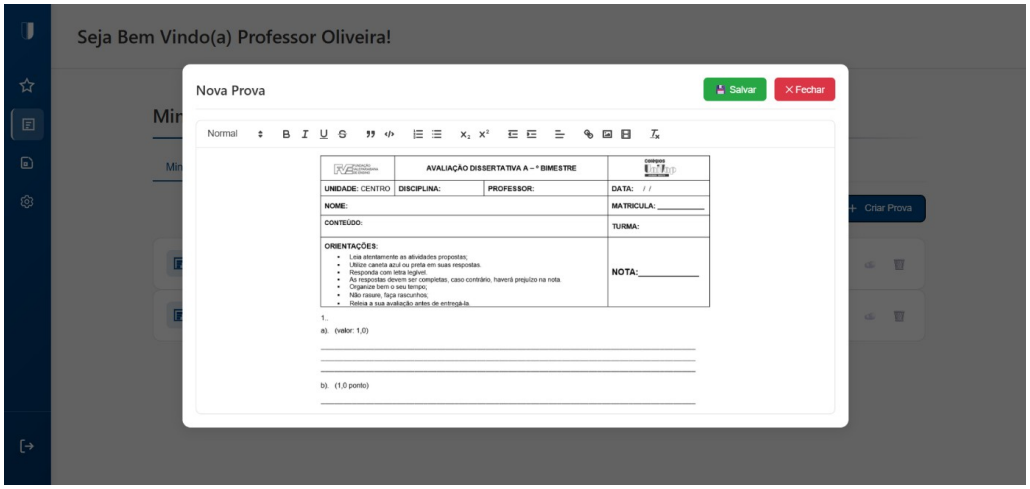
O protótipo desenvolvido integrou os principais módulos definidos na modelagem funcional. No ambiente do Professor, a plataforma reúne consulta e cadastro de questões, filtros do banco, visualização de provas, montagem assistida e envio para validação. No ambiente do Processo Pedagógico, foram concentradas funções de avisos, gestão de usuários, importação e impressão de provas, validação do material recebido e acionamento do fluxo de correção. Essa organização separa responsabilidades e mantém as etapas do processo avaliativo em uma única aplicação.

A interface de montagem denominada “Nova Prova” consolidou, em uma mesma janela, a personalização do documento e a inserção de itens de avaliação. O sistema apresenta uma pré-visualização da folha, cabeçalho institucional com dados de unidade, disciplina, professor, data e estudante, além de campos para nota e orientações. Um editor de texto enriquecido permite formatação tipográfica, inserção de mídias, hiperlinks e equações, enquanto as questões podem ser adicionadas a partir do banco de dados ou de autoria do próprio professor. O salvamento centralizado no MongoDB mantém o modelo da prova disponível para continuidade e posterior validação.

A integração entre Flask, Node.js, MongoDB e os componentes da interface permitiu implementar o fluxo de persistência e manipulação dos dados da avaliação. O projeto também incorporou a estrutura necessária para identificação por QR Code e processamento das imagens digitalizadas com OpenCV. A estimativa financeira apresentada na Tabela 1 totalizou R$ 26.558,96 em capital humano, considerando análise, backend e frontend.

Para avaliar o desempenho da rotina de correção automatizada, foram realizados ensaios funcionais com gabaritos preenchidos corretamente utilizando canetas nas cores preta e azul. Nos testes realizados, o sistema apresentou uma taxa de acerto de 98% na identificação das marcações. Além da precisão, foi avaliado o desempenho de processamento da rotina de Visão Computacional, que apresentou capacidade de correção de aproximadamente 57 provas por segundo nas condições utilizadas nos ensaios. Os testes foram executados em uma máquina equipada com processador Intel Core i5 de 12ª geração, 32 GB de memória RAM DDR4 e placa de vídeo NVIDIA GeForce RTX 4070. Esses resultados fornecem uma avaliação quantitativa preliminar do funcionamento da solução e demonstram a capacidade do protótipo de realizar a correção automatizada em alta velocidade sob as condições controladas do experimento.

Figura 1 - Tela principal do sistema durante a montagem de uma nova prova.



Fonte: Autores (2026).

Discussão

A centralização do banco de questões e da montagem de provas reduz a dependência de arquivos isolados e de operações manuais de busca, cópia e formatação. No processo tradicional descrito no trabalho, essas etapas podem provocar erros de numeração, quebras de página e perda de padronização. Ao reunir edição, armazenamento e validação em uma mesma aplicação, o protótipo aproxima-se do papel atribuído às TIC na gestão e organização da informação, favorecendo a automatização de rotinas administrativas e pedagógicas (Pinto; Molina; Paletta, 2022).

A associação entre QR Code e Visão Computacional também amplia a rastreabilidade do processo de correção. O código individual estabelece o vínculo entre a prova impressa e seus dados digitais, enquanto a análise das imagens permite identificar as marcações do cartão-resposta. Essa abordagem apresenta relação direta com trabalhos anteriores de correção automatizada de avaliações objetivas. Menoncin e Antonello (2017), por exemplo, desenvolveram um sistema baseado em visão computacional e OpenCV capaz de identificar um QR Code para associar a avaliação ao seu respectivo gabarito e, posteriormente, reconhecer as respostas marcadas no cartão-resposta. De forma semelhante, o presente trabalho utiliza a identificação automatizada e o processamento de imagens para reduzir etapas manuais do processo de correção. Entretanto, enquanto o trabalho de Menoncin e Antonello (2017) concentra-se principalmente no reconhecimento do cartão-resposta e na obtenção da nota, o protótipo desenvolvido neste estudo busca integrar essa funcionalidade a um fluxo mais amplo de gerenciamento de questões, montagem de avaliações, armazenamento dos dados e organização dos resultados. Essa integração constitui uma das principais diferenças da proposta, uma vez que a correção deixa de ser tratada como uma etapa isolada e passa a fazer parte de um processo centralizado de gestão das avaliações.

Os ensaios funcionais realizados com a rotina de correção automatizada forneceram indicadores quantitativos preliminares sobre o desempenho da solução. Considerando gabaritos preenchidos corretamente com canetas pretas e azuis, o sistema apresentou taxa de acerto de 98% na identificação das marcações. Além disso, foi observada uma capacidade de processamento de aproximadamente 57 provas por segundo nas condições do experimento. O desempenho foi obtido em uma máquina equipada com processador Intel Core i5 de 12ª geração, 32 GB de memória RAM DDR4 e placa de vídeo NVIDIA GeForce RTX 4070. Esses resultados indicam que a abordagem empregada apresenta potencial para processar grandes volumes de avaliações em um intervalo reduzido de tempo, além de fornecer uma medida inicial de precisão para a rotina de Visão Computacional.

Apesar dos resultados obtidos, os ensaios foram realizados em condições controladas e com gabaritos preenchidos corretamente, não abrangendo de forma completa situações que podem ocorrer durante a utilização prática do sistema, como iluminação fraca ou irregular, rasuras, preenchimentos incompletos, marcações fora da área esperada e outras variações no preenchimento das respostas. Dessa forma, embora a taxa de acerto de 98% represente um resultado quantitativo inicial, são necessários testes adicionais para avaliar a robustez da rotina diante de diferentes condições de captura e preenchimento. A análise de desempenho também pode ser ampliada por meio de testes comparativos com diferentes dispositivos de captura e configurações de hardware, permitindo verificar a influência desses fatores no tempo de processamento.

Durante o desenvolvimento, os principais desafios estiveram na integração entre módulos, na modelagem flexível do banco e na precisão das rotinas de processamento de imagens. Os resultados quantitativos obtidos contribuem para caracterizar tecnicamente o protótipo, mas a avaliação permanece preliminar devido ao escopo controlado dos ensaios. Como continuidade, o projeto prevê incorporar reconhecimento óptico de caracteres, aprendizado de máquina e redes neurais, além de mecanismos de análise estatística do desempenho dos estudantes, geração de relatórios pedagógicos e integração com sistemas acadêmicos existentes.

Conclusão

O desenvolvimento demonstrou a viabilidade de uma solução computacional voltada à elaboração, organização e correção automatizada de avaliações escolares. A integração entre Vue.js, Flask, Python, Node.js, MongoDB, QR Code e OpenCV permitiu estruturar um sistema modular que reúne banco de questões, montagem, gerenciamento, validação, impressão e processamento das provas, contribuindo para maior padronização e confiabilidade do fluxo avaliativo.

Os objetivos propostos foram atendidos no escopo funcional apresentado pelo protótipo, que oferece uma alternativa tecnológica para reduzir tarefas manuais e modernizar os processos de avaliação. Os refinamentos futuros deverão concentrar-se na validação quantitativa da correção automatizada, no aumento da precisão do reconhecimento das marcações e na integração com outros sistemas institucionais, ampliando a capacidade de uso da plataforma em ambientes educacionais.

Referências

BARBOSA, A. et al. Tempo de trabalho e de ensino: composição da jornada de trabalho dos professores paulistas. Educação e Pesquisa, v. 47, 2021.

CHODOROW, K. MongoDB: The Definitive Guide: Powerful and Scalable Data Storage. 3. ed. Sebastopol: O'Reilly Media, 2019.

FILENO, G. C. Sistema automatizado de provas de múltipla escolha com embaralhamento inteligente de gabaritos. 2025. Trabalho de Conclusão de Curso (Curso Técnico em Informática) - Colégio Técnico Antônio Teixeira Fernandes, São José dos Campos, 2025.

KAEHLER, A.; BRADSKI, G. Learning OpenCV 3: Computer Vision in C++ with the OpenCV Library. Sebastopol: O'Reilly Media, 2017.

MENONCIN, L. C.; ANTONELLO, R. Correção automática de avaliações objetivas com leitura do cartão-resposta por visão computacional. 2017. Trabalho de Conclusão de Curso (Curso Técnico em Informática) - Instituto Federal Catarinense, Campus Luzerna, 2017.

PINTO, D.; MOLINA, L. G.; PALETTA, F. C. Uso das tecnologias da informação e comunicação na gestão da informação e do conhecimento nas organizações. Perspectivas em Gestão & Conhecimento, v. 12, n. 1, 2022.

TIWARI, S. An introduction to QR code technology. In: INTERNATIONAL CONFERENCE ON INFORMATION TECHNOLOGY (ICIT), 2016. Proceedings [...]. Bhubaneswar: IEEE, 2016. p. 39-44.