

KARITI: SISTEMA GRATUITO DE CORREÇÃO AUTOMATIZADA DE PROVAS OBJETIVAS

KARITI: FREE AUTOMATED CORRECTION FOR MULTIPLE-CHOICE TESTS

Mateus Santos Mendonça

Kleber Padovani de Souza

RESUMO

As avaliações pedagógicas são extremamente importantes ao processo de ensino-aprendizagem, uma vez que elas permitem a mensuração da compreensão dos estudantes a respeito dos objetos de estudo e, conseqüentemente, fornecem subsídios que podem nortear os planos de aula futuros dos professores. Contudo, as correções das avaliações realizadas por esses professores podem ser muito onerosas, especialmente quando a avaliação continuada faz parte das práticas pedagógicas dos docentes e quando se considera turmas com elevado número de estudantes — um cenário comum na realidade da educação brasileira. O presente trabalho, sensível a essa limitação prática e visando reduzi-la, disponibiliza, divulga e descreve um sistema gratuito e de códigos-fonte abertos para geração e correção de folhas de respostas de provas objetivas — denominado Kariti. O projeto foi concebido pensando em contribuir com a educação no país por meio da simplificação do trabalho de professores em escolas e universidades na correção de provas dessa natureza. O Kariti, além de poder ser utilizado como um gestor de turmas e notas, permite aos professores criarem folhas de respostas individuais, que, após serem preenchidas pelos estudantes e digitalizadas pelos professores, são automaticamente corrigidas pelo sistema. Com o intuito de viabilizar o seu uso por parte de instituições e professores e também visando permitir a incorporação de melhorias nos códigos por parte de desenvolvedores, o artigo em questão detalha todas as etapas de projeto e desenvolvimento do sistema e o disponibiliza de maneira pública e gratuita a toda a comunidade interessada.

Palavras-chave: Educação; Informática na educação; Visão computacional; Desenvolvimento web; Tecnologia.

ABSTRACT

Pedagogical evaluations are extremely important to the teaching-learning process, as they allow the measurement of students' understanding of the objects of study and, consequently, provide subsidies that can guide teachers' future lesson plans. However, corrections to the evaluations carried out by these teachers can be very costly, especially when continuous evaluation is part of the pedagogical practices of teachers and when considering classes with plenty of students --- a common scenario in Brazilian education. The present work, sensitive to this practical limitation and aiming to reduce it, makes available, discloses and describes a free and open source system for the generation and correction of objective test answer sheets for objective examinations --- called Kariti. The project was conceived thinking of

contributing to education in this country by simplifying the work of teachers in schools and universities in correcting objective exams. Kariti, in addition to being able to be used as a class and grade manager, allows teachers to create individual answer sheets, which, after being filled in by students and scanned by teachers, are automatically corrected by the system. In order to enable its use by institutions and teachers and also to allow developers to incorporate improvements in the codes, this article details all stages of the system's design and development and makes it all publicly available and free of charge for the entire interested community..

Keywords: Education; Informatics in education; Computer vision; Web development; Technology.

Introdução

O processo de avaliação é de extrema importância para determinar e aprimorar o desempenho dos alunos em sala de aula. Ele possibilita ao educador uma visão ampla sobre os seus métodos de ensino, permitindo alterações personalizadas quanto à prática pedagógica adotada pelo professor (SORDI; LUDKE, 2009).

Segundo Silva (2013), “a avaliação é o meio de desenvolvimento profissional e pessoal partindo do indivíduo como ponto-chave para a melhoria das suas práticas de ensino”. Damke e Gonçalves (2014) enfatizam que a avaliação é um meio importante para ampliar o conhecimento dos alunos e reconhecer o que eles aprenderam dos tópicos apresentados. No entanto, por mais que o processo de avaliação seja extremamente essencial ele não é uma tarefa ordinária, e pode consumir muito do tempo do educador – especialmente considerando as turmas com muitos estudantes. Nesse cenário, é comum o uso de avaliações coletivas, que, por mais que sejam dirigidas a muitas pessoas, ainda podem determinar o desempenho individual de cada um dos envolvidos nesse processo.

Existem várias estratégias de avaliação disponíveis ao docente, desde a observação até as provas, cada uma delas com suas vantagens e desvantagens. Dentro desse universo, encontram-se as provas objetivas. Segundo Gil (2020), quando realizadas de forma adequada, as provas objetivas podem contribuir para o processo ensino-aprendizagem, tendo como pontos positivos o oferecimento de retorno imediato ao estudante, a avaliação do trabalho docente e a identificação de deficiências específicas dos estudantes.

Contudo, o tempo para correção de provas, independentemente da modalidade, pode tornar-se impraticável quando se tem uma quantidade massiva de correções a se fazer. Considerando a realidade da educação brasileira, esse é um aspecto a ser levado em consideração quando se pensa em formas de aumentar a qualidade do processo ensino-aprendizagem no país, uma vez que a grande quantidade de estudantes é comum a todas as modalidades de ensino (MONTENEGRO, 2010; FLEURI, 2015).

Esse processo de correção pode se tornar praticável, no entanto, por meio do auxílio da tecnologia. Um exemplo são sistemas OMR (Optical Mark Recognition), que são sistemas baseados em processos que usam de técnicas de visão computacional e processamento de imagens para transformar dados brutos de uma imagem em dados que possam ser analisados por algoritmos em computadores

(TÜMER; KÜÇÜKKARA, 2018). Loke, Kasmiran e Haron (2018) relatam que esse método já vem sendo utilizado por quase 80 anos para diferentes fins, mas que se popularizou ao final do século passado.

Existem diversos sistemas que usam essa tecnologia atualmente para agilizar o processo de correção de provas objetivas, como: Remark Office OMR¹, Mestre GR², Merit³, GAPI⁴ e SGP⁵, no entanto, a utilização dessas plataformas implica em custos financeiros para os usuários. Ademais, existem outros sistemas que disponibilizam funções de correção gratuitas, como Prova Fácil⁶ e Minha Prova⁷, mas que são softwares que possuem seus códigos-fonte fechados — ou seja, apenas os proprietários detentores dos direitos de criação desses softwares podem adicionar funcionalidades ao produto.

Em alternativa a esses sistemas, existem os softwares livres, que são soluções computacionais que dão o livre direito a seus usuários de executar, redistribuir e modificar o produto, seja de forma gratuita ou não (BONILLA, 2014). Ou seja, qualquer um que tenha acesso ao software pode adicionar as modificações que desejar, tornando o mesmo mais robusto e democrático. Baseado nos princípios dessa filosofia e com o intuito de contribuir com processo ensino-aprendizagem, surgiu o sistema intitulado Kariti, apresentado por este artigo.

O Kariti é um sistema web gratuito e de códigos-fonte abertos que visa agilizar o trabalho de professores na correção de provas em escolas e universidades. Além de fornecer ao usuário funções de gestão de turmas, alunos, professores e provas, o Kariti é capaz de gerar folhas de respostas personalizadas para provas com questões objetivas e, posteriormente, corrigir tais folhas de respostas preenchidas pelos estudantes.

O software foi implementado utilizando as linguagens de programação PHP⁸ e Python⁹ — com fundamental suporte da biblioteca OpenCV¹⁰ para a geração e correção das folhas de resposta — e é destinado especialmente a instituições de ensino que desejam auxiliar seus professores com as avaliações educacionais. Para isso, basta a instalação do sistema proposto em servidor web acessível aos seus professores.

O projeto, desenvolvimento e implantação do Kariti serão apresentados neste artigo, que está organizado em 4 seções, sendo esta a primeira. Na próxima seção, é apresentada a metodologia adotada para o desenvolvimento do sistema Kariti. A terceira seção apresenta os resultados obtidos no desenvolvimento, apresentando algumas funcionalidades e interfaces com usuário disponíveis no sistema. Por fim, a última seção apresenta as conclusões do trabalho.

¹ <http://www.remarksoftware.com/>

² <https://mestreg.com.br>

³ <https://meritt.com.br>

⁴ <https://www.facebook.com/gapiapp>

⁵ <https://www.sgpsistema.com>

⁶ <https://www.provafacilnaweb.com.br>

⁷ <https://www.minhaprova.com.br>

⁸ <https://www.php.net>

⁹ <https://www.python.org>

¹⁰ <https://pypi.org/project/opencv-python>

2 METODOLOGIA

A metodologia aplicada no projeto foi criada pensando no ciclo completo da vida de um software, começando desde a análise de requisitos até a entrega do projeto final aos usuários, e foi dividida em 5 etapas, conforme segue.

2.1 Definição e documentação de funcionalidades, interfaces e demais especificidades do sistema

Inicialmente, para a definição e documentação de funcionalidades, foram listadas e documentadas todas as funcionalidades previstas para o projeto, a fim de poder ter uma visão ampla e mais descritiva do sistema a ser desenvolvido.

Em seguida, foi realizada a definição de alto nível de funcionamento do sistema, por meio da criação de diagramas para representar as conexões internas do sistema, como mostrado na Figura 1. Em seguida, criou-se um protótipo visual do sistema, como mostrado na Figura 2.

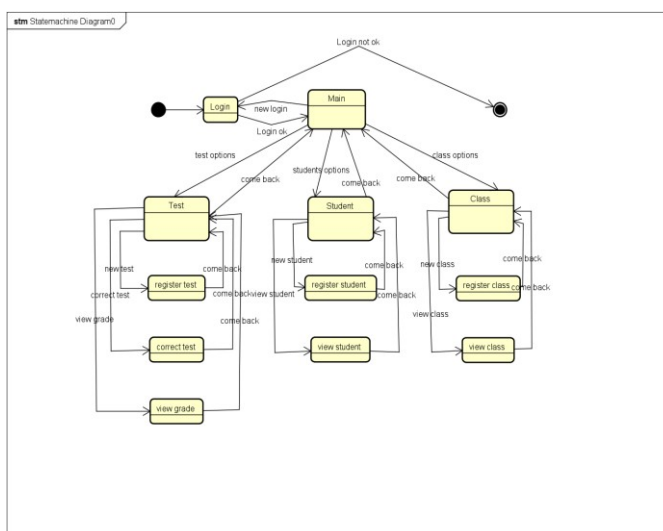


Figura 1 - Diagrama mostrando interações no sistema. (Autores)

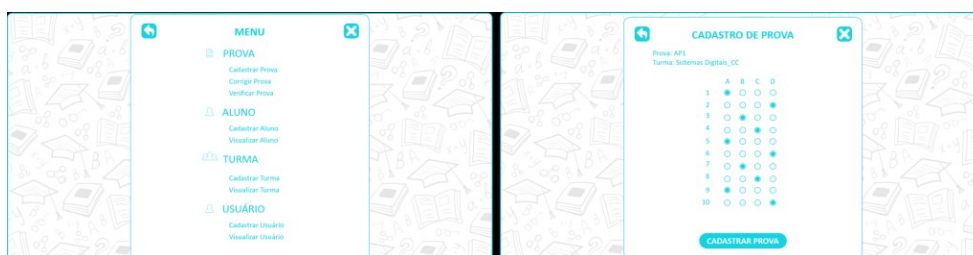


Figura 2 - Prototipagem do sistema. (Autores)

Por fim, foi elaborada a composição visual das folhas de respostas, por meio da criação de protótipos do leiaute das folhas de respostas, para que fossem posteriormente geradas automaticamente pelo sistema. A Figura 3(A) mostra o primeiro protótipo criado, após algumas modificações podemos ver o protótipo final na Figura 3(B).

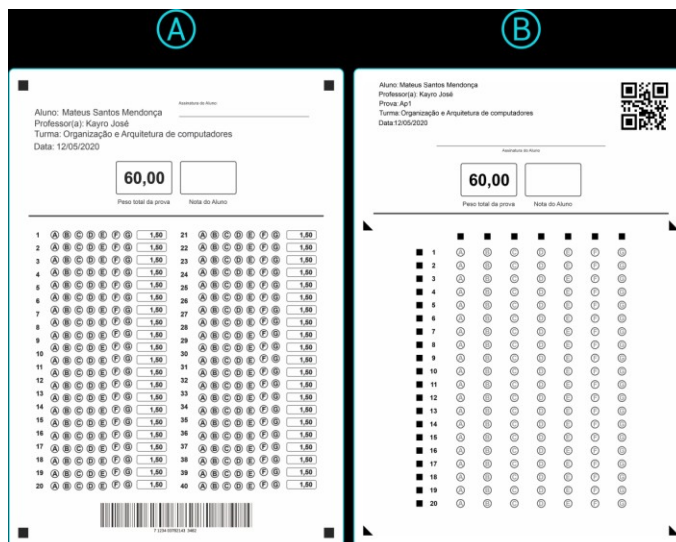


Figura 3 - (A) Primeiro protótipo da folha de respostas. (B) Último protótipo da folha de respostas. (Autores)

2.2 Implementação da geração automática das folhas de resposta

Após a obtenção do leiaute das folhas de respostas, foram inicialmente definidos os parâmetros de entrada necessários para a geração de folhas de respostas, sendo eles: o número identificador da prova e número identificador do aluno. Baseado nesses dois dados, é possível buscar as informações no banco de dados para geração das respectivas folhas de respostas.

Em seguida, foi implementada a geração de imagem para a folha de respostas usando a linguagem de programação Python, com suporte das bibliotecas `fpdf`¹¹, `PIL`¹², `pyQRcode`¹³ e `sqlite3`¹⁴. Assim, as imagens das provas foram então produzidas e armazenadas em um único arquivo PDF de maneira automática.

2.3 Implementação da leitura e correção folhas de resposta

Para a implementação do reconhecimento das marcações existentes nas folhas de respostas, foi utilizada a biblioteca `OpenCV` implementada em Python. Com ela, foi possível ler e reorientar a imagem correspondente à folha de respostas e extrair as demais informações necessárias para a sua correção. O processo de reconhecimento e correção de folhas de resposta é descrito a seguir em quatro etapas, conforme segue.

a. Ajuste de orientação da imagem

A primeira informação adquirida pelo algoritmo após ler a imagem é a localização dos marcadores em forma de triângulos, como destacados em vermelho na Figura 4. Esses triângulos permitirão ao sistema, conforme apresentado a seguir, identificar o ângulo de orientação da imagem e definir o eventual ajuste.

¹¹ <https://pyfpdf.readthedocs.io/en/latest>

¹² <https://pypi.org/project/Pillow>

¹³ <https://pypi.org/project/PyQRCode>

¹⁴ <https://docs.python.org/3/library/sqlite3.html>

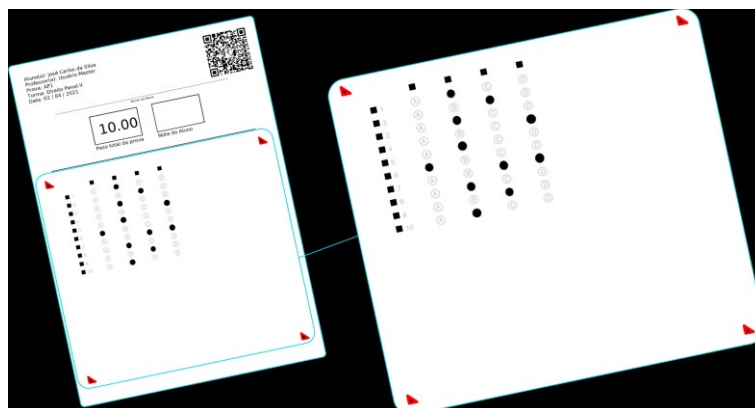


Figura 4 - Localização dos marcadores triangulares nas folhas de respostas. (Autores)

Para cada triângulo, o sistema identifica seus vértices [Figura 5(A)], traça uma reta r partindo do vértice contido entre os dois catetos até o ponto localizado no centro da hipotenusa e calcula o ângulo de inclinação entre r e uma reta horizontal de ângulo nulo [Figura 5(B)].

Então é feita a média entre o resultado dos valores dos 4 triângulos [Figura 5(C)], com intuito de eliminar possíveis ruídos. Desse resultado, são subtraídos 45° , para que assim seja possível encontrar a quantidade de graus de rotação para que a imagem fique na orientação esperada (na vertical) [Figura 5(D)].

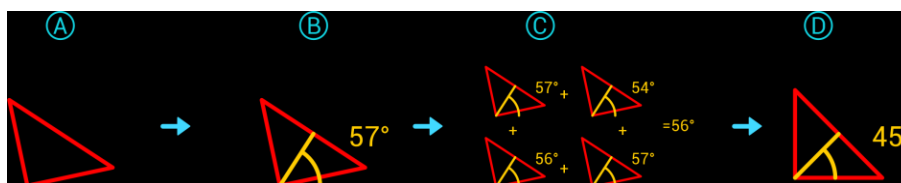


Figura 5 - (A) Identificação dos vértices. (B) Ângulo de inclinação. (C) Média entre os ângulos dos triângulos. (D) Triângulo rotacionado. (Autores)

b. Captura das respostas

Após ter a imagem alinhada corretamente, o algoritmo inicia a identificação dos outros marcadores óticos. Os primeiros marcadores a serem identificados são os marcadores quadrados, como destacados em vermelho na Figura 6(A).

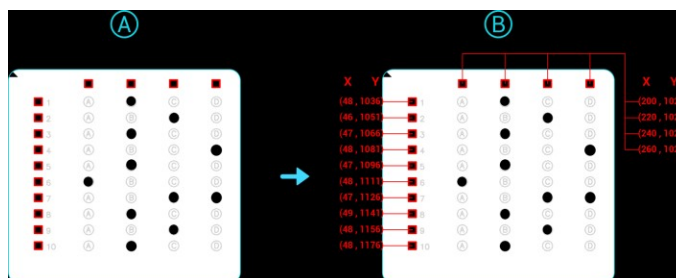


Figura 6 - (A) Identificação marcadores quadrados. (B) Classificação dos marcadores quadrados. (Autores)

Em seguida, esses marcadores são classificados em marcadores de questões (alinhados verticalmente) e marcadores de alternativas (alinhados horizontalmente). Os marcadores cujas coordenadas no eixo das abscissas (eixo x) forem semelhantes, serão armazenados em uma lista contendo os marcadores de questões; em contrapartida, os marcadores cuja coordenadas no eixo das ordenadas (eixo y) forem semelhantes, serão armazenados em uma lista contendo os marcadores de alternativas, conforme ilustra da figura 6(B).

Em seguida, cada uma das listas é ordenada em ordem crescente de acordo com a coordenada contrária à qual foram classificadas (os marcadores de questão são ordenados considerando suas localizações no eixo y e os marcadores de alternativas considerando suas localizações no eixo x). Sendo assim, o primeiro marcador da lista de questões equivale à primeira questão, o segundo à segunda e, assim, sucessivamente. Similarmente, na lista de alternativas, o primeiro elemento da lista equivale à primeira alternativa (alternativa A), o segundo à segunda alternativa (alternativa B), e assim sucessivamente.

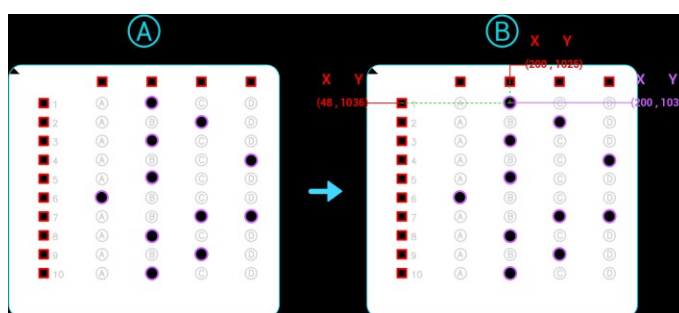


Figura 7 - (A) Identificação marcadores circulares. (B) Classificação dos marcadores circulares. (Autores)

Posteriormente, os marcadores a serem identificados são os marcadores circulares, referentes às respostas apresentadas pelos estudantes — destacados em roxo na Figura 7(A). Para cada marcador circular, são feitas as seguintes verificações: se sua coordenada no eixo y é semelhante à coordenada no mesmo eixo de algum marcador de questão — isso determina a qual questão essa resposta pertence; e se a coordenada no eixo x é semelhante à coordenada no mesmo eixo de algum dos marcadores de alternativas — isso determina a qual alternativa equivale aquela resposta, conforme mostrado na Figura 7(B).

c. *Teste de leitura de folhas de respostas*

A fim de determinar a eficácia dos processos executados pelo algoritmo de reconhecimento de marcações, foram realizados testes com 30 provas em diferentes situações, explorando ângulos diferentes de digitalização das imagens e marcações diferentes. Devido à restrição de aulas presenciais nas instituições de ensino brasileiras, imposta pelas medidas de combate à pandemia de COVID-19, não foi possível a realização de testes em ambientes reais de provas — para correção de cartões de resposta de uma prova real.

d. Implementação do validador de respostas

Após capturar todas as repostas em uma prova, o algoritmo identifica o número identificador da prova e do estudante através da leitura do QRCode contido no cartão de respostas. Posteriormente, busca-se o gabarito da prova na base de dados, compara cada questão e, caso esteja em consonância com o gabarito, a questão recebe o peso da questão para essa questão — a ser definido pelo professor; caso esteja incorreta, sem nenhuma alternativa marcada ou com mais de uma resposta, a questão recebe a pontuação zero. Ao final do processo, as repostas da prova são armazenadas na base de dados do sistema.

2.4 Implementação da leitura e correção folhas de resposta

Com o intuito de permitir a utilização do sistema por parte dos professores, utilizando a linguagem de programação PHP, foram criadas páginas de internet que oferecem meios para o usuário administrar provas, alunos, turmas e usuários do sistema. Posteriormente, o sistema de correção de folhas de resposta foi integrado à interface gráfica.

2.5 Documentação e disposição dos códigos-fonte de maneira pública

Para facilitar a utilização do usuário e também a instalação do sistema, foi elaborado um manual detalhando e instruindo a usar todas as funções do sistema, contendo ainda os passos para a instalação do Kariti. Adicionalmente, são disponibilizados gratuitamente os códigos-fonte do sistema em dois repositórios de dados, para maior acesso ao sistema. O primeiro é o GitLab, cujo repositório é <https://gitlab.com/kariti/kariti.git>; o segundo é o GitHub, cujo repositório é <https://github.com/karitisystem/kariti.git>.

3 RESULTADOS E DISCUSSÕES

Todas as funções planejadas para o sistema foram implementadas e testadas. As subseções a seguir apresentam, respectivamente, as funcionalidades implementadas e disponíveis no Kariti e os testes realizados no sistema.

3.1. Funcionalidades do sistema

Para ter acesso às funcionalidades do sistema, é necessário que o usuário (professor) tenha um perfil previamente cadastrado na base de dados do sistema, para em seguida inserir suas informações na Tela de Login. Na tela de menu, o usuário tem à sua disposição todas as funções referentes a prova, aluno e turma. Apenas o usuário com a chave principal, definida no momento da instalação do sistema, tem acesso à sessão de administração de outros usuários.

O Kariti possui algumas funções específicas, que são funções usadas unicamente por um módulo, e funções gerais, que são funções presentes em diversos módulos.

3.1.1. Funcionalidades específicas

No cadastro de provas o usuário insere, respectivamente, o nome da prova, turma da prova, quantidade de questões (limitada a 20) e quantidade de alternativas (limitada a 7). Após clicar no botão GERAR PROVA, ele é redirecionado para a tela de cadastro de gabarito. Para cadastrar o gabarito, basta inserir as informações: data da prova, resposta para cada questão, peso para cada questão e clicar no botão CADASTRAR PROVA.

A tela de correção de provas é onde as imagens das folhas de respostas preenchidas e digitalizadas devem ser inseridas para a correção. Após a correção, essas provas aparecem na área de busca, onde é possível obter um detalhamento de uma prova clicando em VERIFICAR PROVA ou gerar as folhas de resposta em branco clicando em GERAR CARTÃO RESPOSTA. A Figura 8 ilustra os processos geração, preenchimento e correção de uma prova.

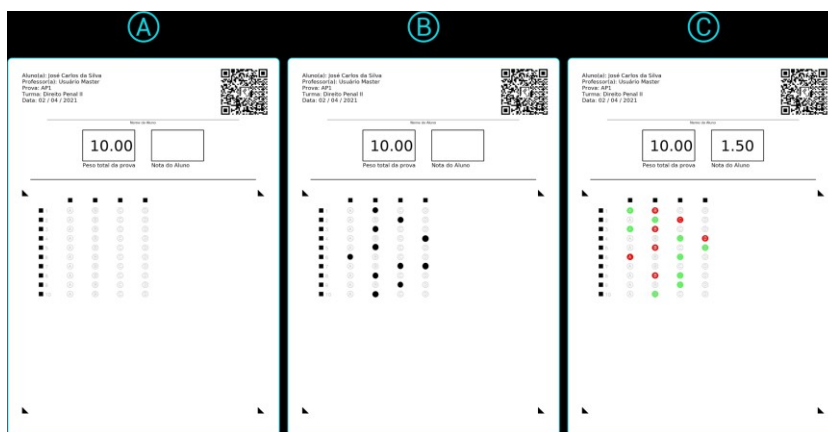


Figura 8 - (A) Prova gerada pelo sistema. (B) Prova preenchida. (C) Prova corrigida pelo sistema. (Autores)

Na tela de verificação das provas é possível saber qual a pontuação de um aluno e quantas questões ele acertou. Ademais, essa tela possui mais quatro funções. a) **BAIXAR PROVAS**: gera e baixa a prova de todos os alunos selecionados, mostrando seus erros e acertos em cada questão; b) **BAIXAR CSV**: gera e baixa um arquivo CSV (separados por vírgula) contendo as notas dos alunos selecionados; c) **ENVIAR POR E-MAIL**: envia para cada aluno um e-mail com um resumo do seu desempenho e suas respectivas provas corrigidas; d) **DELETAR PROVA**: deleta a prova em questão.

Na tela cadastro de alunos é possível cadastrar vários alunos de uma vez usando um arquivo CSV, cujo modelo é disponibilizado ao clicar no botão **BAIXAR MODELO CSV**. Para fazer a inclusão, basta inserir as informações solicitadas no arquivo CSV, fazer o upload do arquivo, clicar em **PROCURAR ARQUIVO**, e clicar no botão **CADASTRAR CSV**. Também é possível cadastrar um aluno por vez, bastando inserir: nome, e-mail e número de matrícula do aluno e clicar em **CADASTRAR ALUNO**.

Na tela de cadastro de turma, para criar uma turma, basta inserir as informações: nome da turma, nome do curso e clicar nos nomes dos alunos que irão fazer parte dessa turma. Para auxiliar na tarefa de selecionar os alunos, o usuário

tem a opção de baixar um arquivo CSV modelo, inserir as informações de nome, e-mail ou número de matrícula dos alunos, fazer o upload desses arquivos e clicar em CADASTRAR CSV. Após isso, o sistema irá selecionar os alunos automaticamente.

Na tela de cadastro de usuário (professor), os professores são inseridos apenas pelo usuário principal (criado no momento da instalação do sistema). Para cadastrar um professor basta apenas inserir as informações: nome de usuário (usado para realizar o login), nome completo do professor e uma senha para acessar o sistema.

3.1.2. Funcionalidades gerais

Como telas que compartilham funções gerais temos as telas de visualização de: aluno, turma e usuário, compartilhando as funções: a) Pesquisar, onde é possível inserir um nome na barra de pesquisa para realizar a pesquisa; b) Editar, onde é possível editar as informações do registro clicando no ícone de lápis e c) Excluir, onde é possível excluir os registros (com exceção do usuário principal).

3.2. Testes com o sistema

Após a finalização da construção da plataforma, foram realizados os testes do sistema. Inicialmente, foram validadas as correções realizadas pelo Kariti nas 30 folhas de resposta previamente preenchidas. O sistema apresentou 100% de acurácia nas correções realizadas.

Adicionalmente, foram realizados testes de instalação e usabilidade do sistema em diferentes plataformas. O primeiro teste foi realizado em um servidor web com sistema operacional Linux. O usuário de teste foi guiado por um dos desenvolvedores, que o instruiu a testar todas as funções presentes no sistema, a fim de encontrar possíveis falhas e melhorias. Nesse teste, foi constatado que algumas telas do sistema não estavam sendo apresentadas no tamanho adequado e o problema foi corrigido.

Similarmente, o segundo teste foi realizado no sistema instalado em um servidor Linux. O usuário de teste também foi guiado por um dos desenvolvedores. Neste teste, foi identificado que o sistema não estava corrigindo corretamente as provas. Foi diagnosticado que uma das etapas de processamento de imagens não estava sendo realizada. Após análise, o problema foi corrigido.

O terceiro teste foi realizado com o sistema instalado em um servidor Windows. O usuário de teste foi também guiado por um dos desenvolvedores. Neste último teste, foi identificado erro no envio das provas aos estudantes, devido à falta de conexão com a internet; no entanto, a mensagem exibida pelo sistema não era amigável. Foi realizado o tratamento do erro e corrigida a falha.

Ao final dos testes com usuários, foram feitos testes internos para comprovar a consistência do sistema. Os próprios desenvolvedores interagiram com a plataforma e verificaram que ela está pronta e adequada para executar as tarefas para a quais fora desenvolvida.

4 CONSIDERAÇÕES FINAIS

O presente trabalho mostrou a importância da avaliação dos estudantes no processo de ensino- aprendizagem. Contudo, é fato que, para os professores, o trabalho mecânico de correção dessas avaliações pode ser um tanto oneroso devido à grande demanda de alunos usual do sistema de ensino. Como alternativa, foi apresentado o Kariti, um sistema web de criação e correção de provas objetivas que tem o intuito de auxiliar os professores nessa árdua tarefa diária.

O Kariti usa técnicas de visão computacional e processamento de imagens para gerar e corrigir folhas de respostas de maneira automática, para que os professores possam usar seu tempo de maneira mais eficaz. Além dessas funções, o sistema também permite organizar os alunos em turmas, aplicar provas a essas turmas e mostrar as notas dos alunos nessa turma de forma organizada.

O sistema é direcionado a instituições de ensino interessadas em fornecer o serviço aos seus professores. Contudo, o próprio professor — caso possua conhecimentos técnicos necessários à instalação — também pode fazer uso de forma independente. A instalação requer do interessado apenas um computador com sistema operacional Linux ou Windows. Todavia, o processo de correção mediado pelo Kariti ganha ainda mais celeridade caso o professor conte com equipamento de digitalização automática das folhas de resposta, como realizado por algumas impressoras multifuncionais existentes no mercado.

O sistema é particularmente interessante para instituições, uma vez que pode ser utilizado pela maioria dos professores, senão todos, dado que as provas objetivas podem ser aplicadas em diferentes áreas do conhecimento. Como o professor insere apenas o número de questões e o número de alternativas das questões — e não os enunciados das questões, o sistema se ajusta a qualquer prova objetiva, respeitados os limites nos números de questões e alternativas existentes.

Ademais, em consonância com a filosofia de software livre, é possível que novas funcionalidades possam ser agregadas ao Kariti, visto que o sistema está disponível gratuitamente na internet e permite a redistribuição e modificação de terceiros. Essa iniciativa contribui para a democratização das tecnologias educacionais para que, assim, o ensino se torne cada vez mais eficiente e de qualidade.

REFERÊNCIAS

BONILLA, M. H. S. **Software livre e educação: uma relação em construção. Perspectiva**, 32(1), 205–234, Abril. 2014. Retrieved from <https://doi.org/10.5007/2175-795x.2014v32n1p205> doi:10.5007/2175-795x.2014v32n1p205

DAMKE, A. S., & GONÇALVES, J. P. **Cadernos de educação. Cadernos de Educação.**, 94–105, 2014.

FLEURI, R. M. **Perfil profissional docente no brasil: Metodologias e categorias de pesquisas**. In (Vol. 40, p. 55), 2015. Brasília: Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (Inep).

GIL, A. C. **Metodologia do ensino superior**. In (5th ed., p. 111-112), 2020. Atlas.

LOKE, S. C., KASMIRAN, K. A., & HARON, S. A. **A new method of mark detection for software-based optical mark recognition**. PLOS ONE, 13(11), e0206420, November. 2018. Retrieved from <https://doi.org/10.1371/journal.pone.0206420> doi:10.1371/journal.pone.0206420

MONTENEGRO, I. P. **Ser professor: uma pesquisa sobre o que pensa o docente das principais capitais brasileiras**. In (p. 43), 2010, 5. São Paulo: Fundação Victor Civita.

SILVA, M. D. O. **A importância da observação de aulas no processo de avaliação de desempenho docente: concepções de professores**. *Gestão e Desenvolvimento*, n. 21, 2013. Retrieved from <https://revistas.ucp.pt/index.php/gestaoedesenvolvimento/article/view/254> doi:10.7559/GESTAOEDESENVOLVIMENTO.2013.254

SORDI, M. R. L. D., & LUDKE, M. **Da avaliação da aprendizagem à avaliação institucional: aprendizagens necessárias**. *Avaliação: Revista da Avaliação da Educação Superior (Campinas)*, 14(2), 313–336, Julho. 2009. doi:10.1590/s1414-40772009000200005

TÜMER, A., & KÜÇÜKKARA, Z. **An image processing oriented optical mark recognition and evaluation system**. *International Journal of Applied Mathematics, Electronics and Computers*, 6, 59-64, 2018, 12. doi:10.18100/ijam ec.2018447788