



INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE GOIÁS

PRÓ-REITORIA DE ENSINO

CÂMPUS GOIÂNIA

BACHARELADO EM SISTEMAS DE INFORMAÇÃO

ALEX BATISTA ARANTES NETO

Aplicação para Auxílio à Correção de Atividades Avaliativas

Goiânia, 2023.



INSTITUTO FEDERAL
Goiás

MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO
SISTEMA INTEGRADO DE BIBLIOTECAS

TERMO DE AUTORIZAÇÃO PARA DISPONIBILIZAÇÃO NO REPOSITÓRIO DIGITAL DO IFG - ReDi IFG

Com base no disposto na Lei Federal nº 9.610/98, AUTORIZO o Instituto Federal de Educação, Ciência e Tecnologia de Goiás, a disponibilizar gratuitamente o documento no Repositório Digital (ReDi IFG), sem ressarcimento de direitos autorais, conforme permissão assinada abaixo, em formato digital para fins de leitura, download e impressão, a título de divulgação da produção técnico-científica no IFG.

Identificação da Produção Técnico-Científica

- | | |
|--|---|
| ☐ Tese | ☐ Artigo Científico |
| ☐ Dissertação | ☐ Capítulo de Livro |
| ☐ Monografia – Especialização | ☐ Livro |
| ☒ TCC - Graduação | ☐ Trabalho Apresentado em Evento |
| ☐ Produto Técnico e Educacional - Tipo: _____ | |

Nome Completo do Autor: Alex Batista Arantes Neto

Matrícula: 20191011090315

Título do Trabalho: Aplicação para Auxílio à Correção de Atividades Avaliativas

Autorização - Marque uma das opções

1. (X) Autorizo disponibilizar meu trabalho no Repositório Digital do IFG (acesso aberto);
2. () Autorizo disponibilizar meu trabalho no Repositório Digital do IFG somente após a data ____/____/____ (Embargo);
3. () Não autorizo disponibilizar meu trabalho no Repositório Digital do IFG (acesso restrito).

Ao indicar a opção **2 ou 3**, marque a justificativa:

- () O documento está sujeito a registro de patente.
() O documento pode vir a ser publicado como livro, capítulo de livro ou artigo.
() Outra justificativa: _____

DECLARAÇÃO DE DISTRIBUIÇÃO NÃO-EXCLUSIVA

O/A referido/a autor/a declara que:

- i. o documento é seu trabalho original, detém os direitos autorais da produção técnico-científica e não infringe os direitos de qualquer outra pessoa ou entidade;
- ii. obteve autorização de quaisquer materiais inclusos no documento do qual não detém os direitos de autor/a, para conceder ao Instituto Federal de Educação, Ciência e Tecnologia de Goiás os direitos requeridos e que este material cujos direitos autorais são de terceiros, estão claramente identificados e reconhecidos no texto ou conteúdo do documento entregue;
- iii. cumpriu quaisquer obrigações exigidas por contrato ou acordo, caso o documento entregue seja baseado em trabalho financiado ou apoiado por outra instituição que não o Instituto Federal de Educação, Ciência e Tecnologia de Goiás.

Alex Batista Arantes Neto

Goiânia, 07/07/2023.
Local Data

Assinatura do Autor e/ou Detentor dos Direitos Autorais

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE GOIÁS

PRÓ-REITORIA DE ENSINO

CÂMPUS GOIÂNIA

BACHARELADO EM SISTEMAS DE INFORMAÇÃO

ALEX BATISTA ARANTES NETO

Aplicação para Auxílio à Correção de Atividades Avaliativas

Trabalho de Conclusão de Curso apresentado à Coordenação do Curso de Bacharelado em Sistemas de Informação como requisito parcial para obtenção do título de Bacharel em Sistemas de Informação.

Orientador: Prof. Dr. Raphael de Aquino Gomes

Goiânia, 2023.

A662a Arantes Neto, Alex Batista.
 Aplicação para auxílio à correção de atividades avaliativas / Alex Batista Arantes Neto. –
 Goiânia: Instituto Federal de Educação, Ciência e Tecnologia de Goiás, Câmpus Goiânia, 2023.
 60 f.: il.

 Orientador: Prof. Dr. Raphael de Aquino Gomes.

 TCC (Trabalho de Conclusão de Curso) – Bacharelado em Sistemas de Informação, Instituto
 Federal de Educação, Ciência e Tecnologia de Goiás, Câmpus Goiânia.

 1. Fireworks (Programa de computador) – atividades avaliativas – processo de ensino-
 aprendizagem. I. Gomes, Raphael de Aquino (orientador). II. Instituto Federal de Educação,
 Ciência e Tecnologia de Goiás, Câmpus Goiânia. III. Título.

CDD 006.68

Ficha catalográfica elaborada pelo Bibliotecário Alisson de Sousa Belthodo Santos CRB1/ 2.266
Biblioteca Professor Jorge Félix de Souza,
Instituto Federal de Educação, Ciência e Tecnologia de Goiás, Câmpus Goiânia.



INSTITUTO FEDERAL
Goiás

MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE GOIÁS
CÂMPUS GOIÂNIA

ALEX BATISTA ARANTES NETO

Aplicação para Auxílio à Correção de Atividades Avaliativas

Trabalho de Conclusão de Curso apresentado ao Curso de Bacharelado em Sistemas de Informação do Instituto Federal de Educação, Ciência e Tecnologia de Goiás, como parte das exigências para obtenção do Título de Bacharel em Sistemas de Informação.

Goiânia, 30 de junho de 2023.

BANCA EXAMINADORA

Prof. Dr. Raphael de Aquino Gomes

Orientador - IFG/Câmpus Goiânia

Me. Daniela Hilda de Souza Siqueira

Membro Externo da Banca Examinadora - IFG/Câmpus Goiânia

Prof. Me. Carlos Augusto da Silva Cabral

Membro Interno da Banca Examinadora - IFG/Câmpus Goiânia

Documento assinado eletronicamente por:

- Daniela Hilda de Souza Siqueira, TÉCNICO EM ASSUNTOS EDUCACIONAIS, em 30/06/2023 17:41:51.
- Carlos Augusto da Silva Cabral, PROFESSOR ENS BASICO TECN TECNOLÓGICO, em 30/06/2023 15:55:27.
- Raphael de Aquino Gomes, PROFESSOR ENS BASICO TECN TECNOLÓGICO, em 30/06/2023 15:11:51.

Este documento foi emitido pelo SUAP em 30/06/2023. Para comprovar sua autenticidade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifg.edu.br/autenticar-documento/> e forneça os dados abaixo:

Código Verificador: 425750
Código de Autenticação: cde448e946



Instituto Federal de Educação, Ciência e Tecnologia de Goiás
Rua 75, nº 46, Centro, GOIÂNIA / GO, CEP 74055-110
Sem Telefones cadastrados

Dedicatória

Dedico este trabalho aos meus pais e minhas irmãs, que sempre me deram apoio em tudo.

Agradecimentos

A todos os professores que foram parte da minha formação e que lecionaram com a preocupação no desenvolvimento crítico e no respeito de seus alunos.

Ao meu orientador Raphael, por ter me guiado no desenvolvimento deste trabalho com muita dedicação e atenção.

À minha família e meus amigos que de grande suporte foram durante todo esse período.

Usar a tecnologia e não ser usados ou manipulados docilmente como objetos por ela – não que a tecnologia tenha vida por si própria, mas ela pode ser usada para manipular e estar a serviço de uma concepção de mundo que não é emancipadora

Anderson Fernandes de Alencar,
2005.

Resumo

Título: Aplicação para Auxílio à Correção de Atividades Avaliativas

Autor: Alex Batista Arantes Neto

Orientador: Dr. Raphael de Aquino Gomes

A correção de atividades avaliativas é uma tarefa que pode demandar bastante tempo e foco da pessoa que a realiza. Quando o meio utilizado para o registro das respostas é o papel, a execução pode ser ainda mais trabalhosa, já que exige também do docente a manipulação, o armazenamento e o controle das entregas. Além disso, pode ser bastante cansativa a inserção de comentários a respeito da resposta de cada entrega. Com base nisso, este trabalho tem como o objetivo apresentar uma solução que auxilie a correção de atividades realizada em papel, de modo que o processo seja realizado digitalmente, permitindo a implementação de diversas funcionalidades que ajudem o docente. A ferramenta foi implementada na forma de uma aplicação web que armazena as entregas a partir das imagens das mesmas e permite um fluxo de correção por questão ou por atividade, agrupamento de respostas e recebimento das entregas e envio das correções via e-mail.

Palavras-chave

correção de atividades, aplicação web, automação

Abstract

Title: Application for Assistance in the Correction of Evaluative Activities

Author: Alex Batista Arantes Neto

Advisor: Dr. Raphael de Aquino Gomes

The correction of evaluation activities is a task that can demand a lot of time and focus from the person who performs it. When paper is used to record responses, execution can be even more laborious, as it also requires the teacher to handle, store, and control deliveries. In addition, it can be quite tiring to insert comments about the response of each delivery. Based on this, this work aims to present a solution that helps correct activities carried out on paper so that the process is carried out digitally, allowing the implementation of several functionalities that help the teacher. The tool was implemented in the form of a web application that stores deliveries based on their images and allows a correction flow by question or activity, grouping responses and receiving deliveries, and sending corrections via email.

Keywords

activity correction, web application, automation

Lista de Figuras

2.1	Diagrama de uma Arquitetura Cliente-servidor.	21
2.2	Passos na requisição de uma página web dinâmica.	23
3.1	Etapas do fluxo da aplicação.	25
3.2	Exemplo de uma região de resposta obtida de uma entrega.	26
3.3	Fluxo de correção questão por questão	27
3.4	Modelo de Entidade-Relacionamento da aplicação.	27
3.5	Arquitetura da aplicação.	28
3.6	Diagrama de navegação das páginas da aplicação	29
3.7	Diagrama de componentes da aplicação.	30
3.8	Exemplo de criação de uma atividade.	31
3.9	Página inicial com atividade criada.	31
3.10	Exemplo de uma página de atividade.	32
3.11	Página de uma atividade com uma única questão criada.	33
3.12	Página de a atividade com três questões criadas.	33
3.13	Selecionando imagens das entregas.	34
3.14	Imagens das entregas selecionadas.	34
3.15	Página da atividade com as entregas inseridas.	35
3.16	Página de uma entrega	36
3.17	Editando a região de uma resposta.	36
3.18	Resposta desalinhada (parte superior) e resposta com região corrigida (parte inferior).	37
3.19	Exemplo de página de questão.	38
3.20	Exemplo de criação de um grupo de respostas.	39
3.21	Exemplo de seleção de respostas para inserir em um grupo.	39
3.22	Respostas selecionadas após serem inseridas em um grupo.	40
3.23	Definindo nota e comentário para um grupo de respostas.	40
3.24	Respostas do grupo com a nota e comentário definidos.	41
4.1	Diagrama de classes do modelo no <i>back-end</i>	43
4.2	Fluxo das mensagens no <i>back-end</i>	45

Lista de Quadros

2.1 Principais elementos da avaliação diagnóstica, somativa e formativa	20
---	----

Lista de Códigos de Programas

4.1	<i>Exemplo de um serializador</i>	43
4.2	<i>Exemplo de ModelAndViewSets</i>	44
4.3	<i>View utilizada para obtenção de um arquivo de correção</i>	45
4.4	<i>Exemplo de um agrupador</i>	46

Lista de Abreviaturas e Siglas

API	<i>Application Programming Interface</i>
AWS	<i>Amazon Web Services</i>
DRY	<i>Don't Repeat Yourself</i>
HTML	<i>Hypertext Markup Language</i>
HTTP	<i>Hypertext Transfer Protocol</i>
IP	<i>Internet Protocol</i>
JSON	<i>JavaScript Object Notation</i>
MVC	<i>Model, View, Controller</i>
PISA	Programa Internacional de Avaliação de Estudantes
SPA	<i>Single-page application</i>
SQL	<i>Structured Query Language</i>
URI	<i>Universal Resource Identifier</i>
URL	<i>Uniform Resource Locator</i>
VoIP	<i>Voice-over-IP</i>

Sumário

1	INTRODUÇÃO	16
1.1	Objetivos	17
1.2	Estrutura do trabalho	18
2	REFERENCIAL TEÓRICO	19
2.1	Processo Avaliativo	19
2.2	Internet e Aplicações Distribuídas	20
2.2.1	World Wide Web	21
2.2.2	Ferramentas para desenvolvimento web	23
3	ARQUITETURA DA APLICAÇÃO E <i>front-end</i>	25
3.1	Fluxo de ações	25
3.2	Arquitetura da aplicação	27
3.3	Front-end	29
3.3.1	Componentes desenvolvidos	29
3.3.2	Detalhes sobre a implementação dos componentes	30
3.3.2.1	Página principal	30
3.3.2.2	Página de atividade	31
3.3.2.3	Página de entrega	35
3.3.2.4	Página de questão	37
4	BACK-END E VALIDAÇÃO	42
4.1	Persistência de dados	42
4.2	Tratamento de dados de requisições	43
4.3	Implementação de operações	44
4.4	Avaliação da aplicação	46
5	CONSIDERAÇÕES FINAIS	49
	REFERÊNCIAS BIBLIOGRÁFICAS	51
	APÊNDICE A – EXEMPLO DE ENTREGA DE ATIVIDADE	55
	APÊNDICE B – EXEMPLO DE FOLHA DE QUESTÕES DE UMA ATIVIDADE	56
	APÊNDICE C – EXEMPLO DE FOLHA DE RESPOSTA	57
	APÊNDICE D – EXEMPLO DE ARQUIVO DE CORREÇÃO	58

Introdução

Em todos os níveis de formação é muito comum que professores lecionem disciplinas para uma alta quantidade de alunos. Como é apontado na pesquisa Programa Internacional de Avaliação de Estudantes (PISA) de 2015 (SANTILLANA, 2016), no Brasil a média de estudantes por professor é de 29, sendo um dos maiores obtidos no estudo. Quando um docente enfrenta esta situação, acaba por ter que lidar com a correção de uma alta quantidade de atividades avaliativas, processo que demanda tempo e bastante foco. Além de uma maior demora para realizar a correção, a demanda massiva de atividades também torna a avaliação mais suscetível a apresentar inconsistências ou ausência de padrão de correção entre um estudante e outro.

A correção pode ser ainda mais prejudicada quando a atividade é realizada e entregue em papel, meio ainda bastante comum em diversas instituições, seja por limitações técnicas ou por escolha do método didático. Com isso, a necessidade de manipulação dos materiais torna mais difícil o armazenamento e submissão das atividades, assim como o registro das entregas e o desempenho dos alunos. Outro fator a ser considerado nessas situações é o esforço que deve ser empregado para que comentários construtivos a respeito das respostas sejam apresentados pelo professor para cada uma das atividades, uma vez que estes devem ser realizados através da escrita manual, de forma repetitiva.

Considerando essas dificuldades e limitações, diversas ferramentas ou ambientes virtuais de aprendizagem, como Moodle (MOODLE, 2022) e Classroom (CLASSROOM, 2022) se baseiam na ideia da transposição do meio físico para o digital visando auxiliar o processo de correção. A partir desta transferência, diferentes abordagens podem ser tomadas, embora se baseiem majoritariamente no uso de questões objetivas.

Apesar das atividades objetivas já serem uma opção de avaliação que torna a correção mais rápida, algumas ferramentas foram desenvolvidas para tornar o processo avaliativo ainda mais eficiente e menos exaustante através da detecção e correção automática das respostas dos alunos. Estas alternativas realizam a correção automática a partir da digitalização dos cartões-resposta, seja por um *scanner*, como é feito no MarkReader (2022), ou através da câmera do celular, como no EvalBee (2022).

Quando apenas questões de múltipla escolha não é a melhor solução para avaliar

o desempenho dos alunos, alternativas como Bloomfield (2010) e Paredes e Hsiao (2021) permitem uma correção livre em cima das imagens digitalizadas das atividades, sendo uma experiência semelhante à correção realizada no papel, porém realizada de maneira totalmente digital com a distribuição realizada de forma online.

Uma opção bastante útil presente em algumas aplicações é a de delimitação do local de resposta nas imagens digitalizadas, permitindo que o fluxo de correção seja realizado de forma agregada para todos os estudantes, considerando cada questão. Esta estratégia é aplicada por Llamas-Nistal *et al.* (2013) e Crowdmark (2022). Indo além dessa abordagem, a plataforma Gradescope (2022) oferece também a funcionalidade de agrupar respostas semelhantes, permitindo que a mesma nota e mesmas rubricas sejam aplicadas apenas uma vez para todas elas.

Considerando todos os pontos característicos dessas aplicações, o Gradescope (2022) é a solução mais completa, oferecendo a possibilidade de correção de avaliações aplicadas em papel no formato digital, com flexibilidade para poder lidar com questões de tipos variados, com o fluxo de correção de questão em questão, e o agrupamento de respostas. Porém, além de ser uma aplicação de código fechado, também necessita de pagamento com custo elevado para desfrutar da ferramenta de correção. Visando oferecer uma solução alternativa, neste trabalho é proposto o desenvolvimento de uma plataforma que permita a correção de avaliações aplicadas em papel de forma digital, implementando abordagens que tornam o gerenciamento das atividades mais fácil e que fazem com que o processo seja menos desgastante. A solução proposta é uma alternativa de código aberto, que permite flexibilidade nos tipos de questões que podem ser corrigidas, e que pode ser facilmente aprimorada e personalizada para que se adapte à necessidade de implementação. O foco do desenvolvimento é direcionado na lógica da aplicação, fornecendo a base para futuros aprimoramentos.

1.1 Objetivos

O objetivo geral deste trabalho é o desenvolvimento de uma plataforma de auxílio à correção de atividades avaliativas aplicadas em papel, através da implementação de funcionalidade que facilitam a correção. São objetivos específicos:

- propor uma estratégia para armazenamento das entregas dos alunos;
- implementar uma interface para correção das entregas de uma atividade que permita inserir comentários e a nota atribuída;
- desenvolver uma solução que elimine tarefas repetitivas da correção;
- projetar uma aplicação que seja facilmente extensível para a inclusão de novas funcionalidades.

1.2 Estrutura do trabalho

O restante do trabalho é organizado em 4 capítulos.

No Capítulo 2 são apresentados alguns conceitos teóricos que ajudarão a compreender o tema e a aplicação desenvolvida.

O Capítulo 3 apresenta a arquitetura da aplicação e a estrutura do que compõe sua interface, através de uma demonstração de seu uso.

O Capítulo 4 descreve a infraestrutura da aplicação, com foco na manipulação dos dados. Também é apresentada uma validação da solução desenvolvida.

Por fim, no Capítulo 5 são apresentadas considerações finais a respeito do que foi alcançado no trabalho desenvolvido.

Referencial Teórico

Este capítulo tem como objetivo apresentar conceitos teóricos que são importantes para a compreensão do trabalho. Inicialmente são discutidas algumas definições relacionadas ao processo avaliativo¹. Em seguida, tecnologias voltadas ao desenvolvimento de aplicações Web são apresentadas.

2.1 Processo Avaliativo

A avaliação é um instrumento tradicionalmente utilizado por professores, que pode cumprir diferentes propósitos no processo educativo. Segundo Silva, Matos e Almeida (2014), existem 3 (três) métodos avaliativos utilizados pelos docentes com finalidades distintas: avaliação diagnóstico, formativa e somativa.

Na avaliação diagnóstico, o docente conhece melhor os alunos, suas expectativas, conhecimentos prévios, dificuldades, etc. Normalmente é aplicada no início de um ciclo de ensino, pois com estas informações o docente pode verificar o preparo dos estudantes para o processo de aprendizado a que serão submetidos.

A avaliação formativa tem a intenção de identificar pontos frágeis dos estudantes, permitindo ao docente realizar melhores ajustes na estratégia de ensino individualmente ou coletivamente. Através do *feedback*, este método também comunica aos estudantes seus erros, oferecendo a oportunidade de melhorar seu próprio desempenho.

Já as avaliações somativas tradicionalmente são utilizadas para classificar a aprendizagem dos alunos, sendo normalmente o resultado obtido nestas atividades que determinará o desempenho dos estudantes em uma disciplina.

O Quadro 2.1 descreve os principais elementos de cada um destes métodos avaliativos.

A avaliação é um instrumento de muita importância no processo educativo. Porém, diversas dificuldades podem surgir na hora de se realizar uma avaliação. Como

¹Vale acrescentar que o presente trabalho não possui foco na área de estudo da educação, visando apenas o desenvolvimento de uma solução para uma situação recorrente enfrentada por docentes

Quadro 2.1: *Principais elementos da avaliação diagnóstica, somativa e formativa*

Tipo de Avaliação	Função	Pra que serve?	Quando Aplicar
Diagnóstica	Permite que o professor entenda e identifique conteúdos em que os alunos possuem aptidão e possíveis defasagens	Permite que o professor entenda e identifique conteúdos em que os alunos possuem aptidão e possíveis defasagens	Antes de iniciar o processo de aprendizagem
Formativa	Promove o acompanhamento, com o intuito de verificar se os alunos estão alcançando os objetivos propostos	Para fornecer aos alunos e professores os chamados <i>feedback</i> quanto ao progresso durante o processo de aprendizagem	Durante o processo de aprendizagem
Somativa	Promove a classificação dos alunos, de acordo com os níveis de aproveitamento previamente estabelecidos	Para medir através de notas e conceitos o aprendizado dos alunos. Indicados por meio de seus resultados	Ao final de um conteúdo, de um período ou ao final de uma etapa educativa

Fonte: (JOVENSGENIOS, 2021)

aponta Machado (2019), as práticas de avaliação estão suscetíveis a critérios internos ou externos do objeto de avaliação, podendo ser de forma consciente ou inconsciente por parte do avaliador.

Uma avaliação possui normalmente dois tipos de questões, as objetivas e as discursivas. As questões objetivas são aquelas que são acompanhadas das opções de respostas, que limitam o universo possível de “solução” para a questão, sendo bastante comum no formato de alternativas. Já as questões discursivas são as que permitem ao aluno desenvolver uma resposta de forma mais livre, mais aberto à dissertação.

As questões objetivas muitas vezes são preferíveis pela facilidade e rapidez que oferecem na correção, podendo inclusive desfrutar de diversas ferramentas que realizam a correção automática dessas questões. Já as questões discursivas normalmente não possuem uma correção tão trivial, já que exigem interpretação e análise da resposta do aluno.

2.2 Internet e Aplicações Distribuídas

Uma rede de computadores é uma infraestrutura que conecta dispositivos entre si, sendo a partir dessa conexão que as máquinas de uma rede podem se comunicar, permitindo que arquivos e dados sejam compartilhados. Para esta troca de mensagens acontecer, é necessário o uso de protocolos, definidos por Kurose e Ross (2012) como o

formato e a ordem das mensagens trocadas entre duas entidades, além das ações tomadas pelas mesmas.

Atualmente as redes de computadores são bastante comuns em residências e ambientes corporativos, sendo a Internet a maior e mais conhecida, conectando dispositivos do mundo inteiro. É utilizando a infraestrutura global da internet que diversos serviços são realizados, como o correio eletrônico, *streaming* de vídeo e *Voice-over-IP* (VoIP). Estes são descritos como aplicações distribuídas, já que acontecem através da troca de mensagens entre diferentes sistemas finais de uma rede (KUROSE; ROSS, 2012).

A forma como essas aplicações são estruturadas nos dispositivos é chamada de arquitetura da aplicação. Uma das mais comuns utilizada na internet é a cliente-servidor. Nesta arquitetura, como o nome indica, existem dois atores, ambos dispositivos finais da rede. O servidor cumpre o papel de receber e responder às requisições feitas pelo sistema cliente. A Figura 2.1 ilustra esta arquitetura.

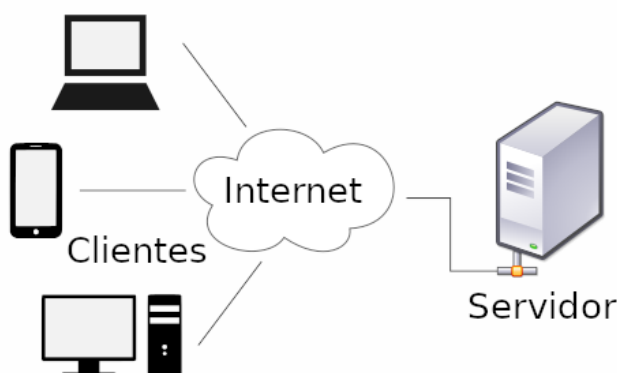


Figura 2.1: Diagrama de uma Arquitetura Cliente-servidor.

Fonte: Adaptado de (VIGNONI; CALIMO; MGYUGCHA, 2011)

Para permitir que as mensagens sejam encaminhadas para o dispositivo destino correto, cada um dos aparelhos conectados na rede recebe um identificador único chamado de endereço *Internet Protocol* (IP). Além disso, também é necessário que seja identificado na comunicação qual o processo de um sistema final deve receber a mensagem, e para isso é informado junto ao endereço IP um número chamado de porta.

2.2.1 World Wide Web

A World Wide Web (também conhecida como apenas Web ou W3) foi desenvolvida para ser um repositório para o conhecimento humano (BERNERS-LEE *et al.*, 1994), e consiste de uma aplicação distribuída, implantada sobre a internet, permitindo o compartilhamento de documentos e outros recursos (também chamados de objetos).

A Web implementa a arquitetura cliente-servidor, sendo que do lado do cliente utiliza-se uma aplicação denominada de navegador web (ou *web browser*) e do lado

do servidor executa-se um servidor web. Através de um navegador, um recurso que esteja disponível em um servidor web pode ser requisitado e visualizado. Para identificar este recurso, utiliza-se de um *Universal Resource Identifier* (URI), que consiste em um identificador que localiza um determinado objeto em um servidor. A requisição é feita conforme o protocolo *Hypertext Transfer Protocol* (HTTP) (FIELDING *et al.*, 1999a).

Existem dois tipos de mensagens no protocolo HTTP, uma de requisição, enviada por clientes, e uma de resposta, enviada por servidores web como resultado de uma requisição. Cada requisição possui um método que indica uma ação que está sendo solicitada ao servidor. Mais comumente são utilizados os métodos GET, utilizado para obter um recurso, POST, para inserir um novo objeto como recurso disponível, PUT, para inserir ou substituir um recurso, e DELETE, para requisitar a exclusão de um recurso (FIELDING *et al.*, 1999b).

Muitos dos recursos são disponibilizados no formato de páginas web, que são documentos hipertexto, que permitem que um conteúdo seja dividido em fragmentos conectados, de forma que seja gerenciado de forma não linear (AGOSTI, 1996). Essas páginas são escritas utilizando a linguagem *Hypertext Markup Language* (HTML). Uma coleção de páginas relacionadas, normalmente hospedadas em um mesmo servidor web, é chamada de website, ou apenas, site.

Quando uma página web é enviada ao cliente que a requisitou da forma que ela é armazenada no servidor web, ela é categorizada como uma página estática. São conteúdos fixos, que não recebem muita alteração, e são iguais para todos os clientes que os requisitam. Em contraste às páginas estáticas, existem as dinâmicas, que são “montadas” durante o processamento da requisição, conforme ilustrado na Figura 2.2.

Os websites serviam majoritariamente para o propósito de comunicar informações, porém cada vez mais é comum ver surgir também as aplicações web. Essas podem ser definidas por uma aplicação cliente-servidor que utiliza um navegador web como seu programa cliente (SHKLAR; ROSEN, 2009). Um dos motivos que explica este crescimento é a maior facilidade de portabilidade de uma aplicação para diferentes dispositivos.

Estas aplicações web são comumente divididas em duas partes, chamadas de *front-end* e *back-end*, que apesar de funcionarem em conjunto, agem de maneira separada. O *front-end* é a interface da aplicação e é executada no cliente, ou seja, no navegador do dispositivo que a acessa. Já o *back-end* é normalmente executado em um computador dedicado, e não só cumpre as funções de fornecer ao cliente páginas e outros recursos que compõe o *front-end*, mas também a de lidar com o estado da aplicação, ou seja, inserindo, persistindo, alterando e encaminhando para o cliente os dados relacionados à aplicação.

Atualmente, muitas aplicações web são implementadas como uma *Single-page application* (SPA). Em uma SPA, uma página web é requisitada apenas uma vez, de forma que qualquer mudança de conteúdo é realizada através de requisições assíncronas apenas

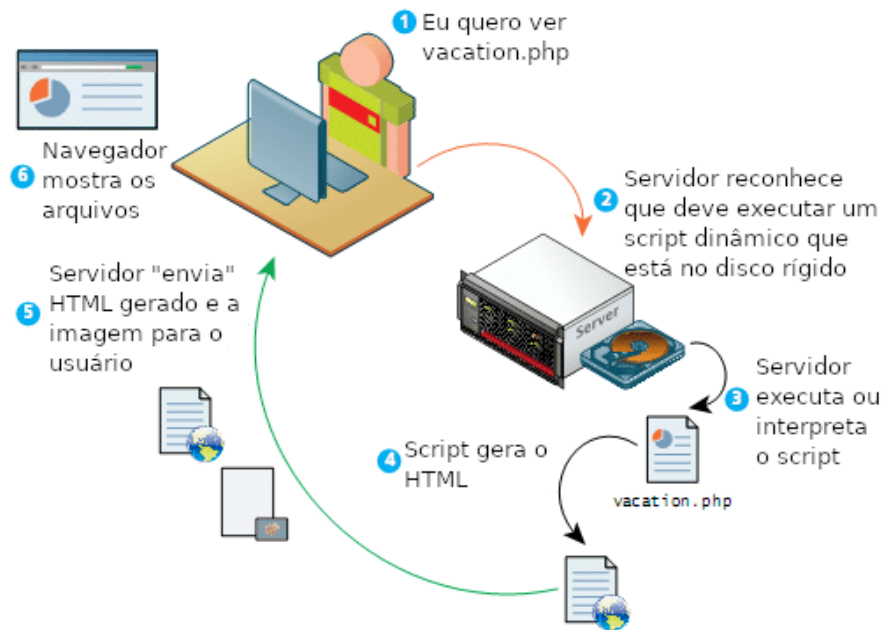


Figura 2.2: Passos na requisição de uma página web dinâmica.

Fonte: Adaptado de (SAIHOOD; ELAMEER, 2016)

dos dados necessários ao servidor, através de uma *Application Programming Interface* (API). As vantagens de uma SPA são tornar a aplicação mais dinâmica, com menos atualizações da estrutura da página e ganho em desempenho.

2.2.2 Ferramentas para desenvolvimento web

Entre as ferramentas mais comumente utilizadas para desenvolvimento de aplicações web, Django (2023) é um *framework* da linguagem Python (2023), voltado para desenvolvimento web. Foi desenvolvido por Adrian Holovaty and Simon Willison (RUBIO, 2017) em 2003 com o objetivo de permitir implementar aplicações web de forma mais rápida enquanto trabalhavam em um jornal (DJANGO, 2023). Em 2005, decidiram tornar o software *open-source* (DJANGO, 2023).

Com o lema “o framework web para perfeccionistas com um prazo máximo”, o Django foi pensado com um conjunto de filosofias de projeto que tornam seu processo de desenvolvimento web bastante produtivo (RUBIO, 2017). Algumas dessas filosofias são:

- *Don't Repeat Yourself* (DRY): traduzida para “não repita a si mesmo”, diz respeito à normalização do código, evitando redundâncias.
- *Explicit is better than implicit*: que significa “Explícito é melhor do que implícito”, foi herdada da linguagem Python e visa tornar o código mais fácil de ser entendido.
- Acoplamento fraco: garante maior independência entre componentes.

Essas filosofias, além de propiciar maior produtividade durante o desenvolvimento, também favorecem a criação de aplicações com uma melhor manutenibilidade. O

acoplamento fraco e o uso da arquitetura *Model, View, Controller* (MVC) (LEFF; RAYFIELD, 2001) também tornam o Django uma boa opção para o desenvolvimento de APIs.

Outra ferramenta bastante utilizada no desenvolvimento web é ReactJS (ou apenas React) (REACT, 2022), uma biblioteca da linguagem JavaScript (EICH, 1995) de código aberto para desenvolvimento de interfaces de usuário no *front-end* de aplicações web. Pode também ser considerado um *framework* que separa a criação da interface de usuário em partes isoladas chamadas de componentes, encorajando o desenvolvimento destes de forma que eles possam ser reutilizados em diferentes partes da aplicação.

O *state* de um componente guarda os dados que descrevem a situação atual do mesmo. Quando o *state* é alterado, o componente é renderizado novamente na interface. Isso faz do React uma ferramenta para a criação de aplicações web do tipo SPA.

Arquitetura da aplicação e *front-end*

Este capítulo e o próximo têm como objetivo descrever a aplicação desenvolvida. Inicialmente, neste capítulo, será descrita a arquitetura geral da aplicação e, em seguida, os elementos de sua interface.

3.1 Fluxo de ações

Para a aplicação foi considerado um caso de uso que define o fluxo básico que um usuário percorre para realizar a correção de uma atividade. Seguindo a ilustração da Figura 3.1, as etapas são:

1. Criação de nova atividade.
2. Definição das questões da atividade.
3. Inserção das entregas dos alunos.
4. Correção das respostas das entregas.
5. Obtenção ou envio das correções.

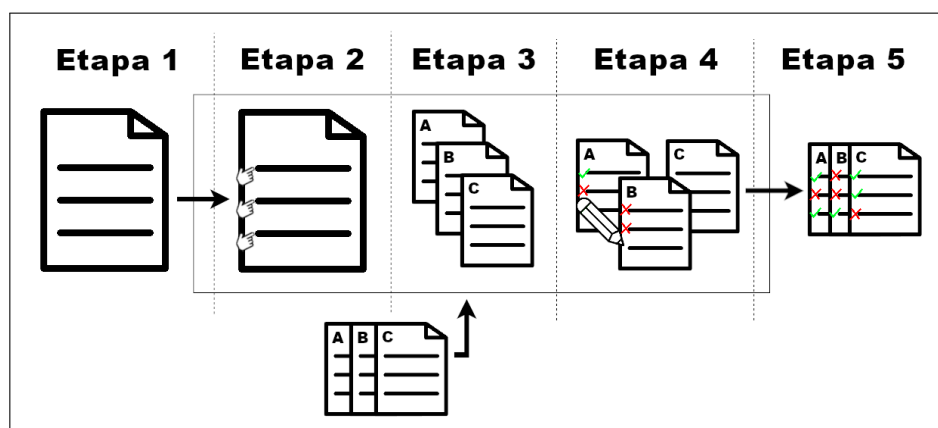


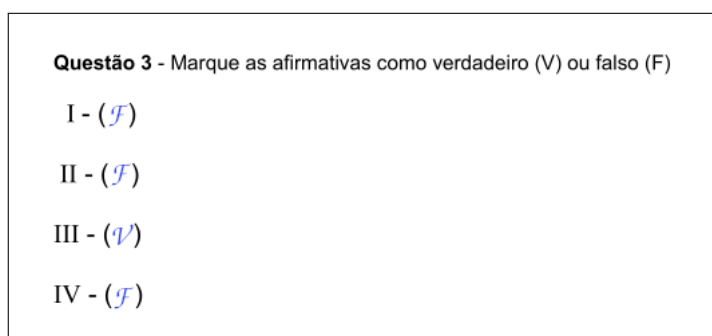
Figura 3.1: *Etapas do fluxo da aplicação.*

Fonte: Elaboração própria.

Para criar a nova atividade, na Etapa 1, deve-se definir um título e uma folha de respostas, aonde os alunos escrevem as respostas. Em vista de uma implementação mais

simples, a folha de respostas deve ter apenas uma página. Utilizando como exemplo a atividade do Apêndice B, uma folha de resposta sugerida seria semelhante à apresentada no Apêndice C.

Na Etapa 2, para cada questão presente na atividade, deve-se determinar um título, uma nota e a região da imagem da folha de respostas que contém a resposta da questão. Esta região consiste em coordenadas que determinam o espaço de resposta de cada questão, e é utilizada para obter recortes das respostas de cada questão das entregas para, assim, facilitar a visualização de uma resposta. Com esta seleção todo o resto da imagem da entrega não será mostrado ao avaliador. O recorte da região das respostas nas entregas é feito utilizando as coordenadas relativas às proporções da imagem da entrega, permitindo que estas tenham tamanhos diferentes da imagem folha de resposta. A Figura 3.2 mostra um exemplo de região selecionada a partir de uma entrega apresentada no Apêndice A.



Questão 3 - Marque as afirmativas como verdadeiro (V) ou falso (F)

I - (F)

II - (F)

III - (V)

IV - (F)

Figura 3.2: Exemplo de uma região de resposta obtida de uma entrega.

Fonte: Elaboração própria.

A inserção das entregas dos alunos, na Etapa 3, consiste na seleção das imagens das folhas de respostas respondidas e um respectivo identificador único de cada aluno, podendo ser, por exemplo, o número de matrícula.

A Etapa 4 pode ser realizada através das formas: i) entrega por entrega ou ii) questão por questão. No formato de entrega por entrega, o usuário corrige todas as respostas de uma entrega por vez, enquanto na forma questão por questão são corrigidas todas as respostas de uma questão por vez, seguindo um fluxo representado pela Figura 3.3. Para o formato de questão por questão pode ser utilizado um agrupamento de respostas semelhantes, para que recebam a mesma nota e comentário.

Para a obtenção das correções, na Etapa 5, é possível obter um arquivo PDF da correção de cada entrega, além de uma planilha com a correção e notas das entregas. A opção de enviar as correções é realizada através do envio de um e-mail para cada aluno, com o arquivo de correção da sua respectiva entrega em anexo.

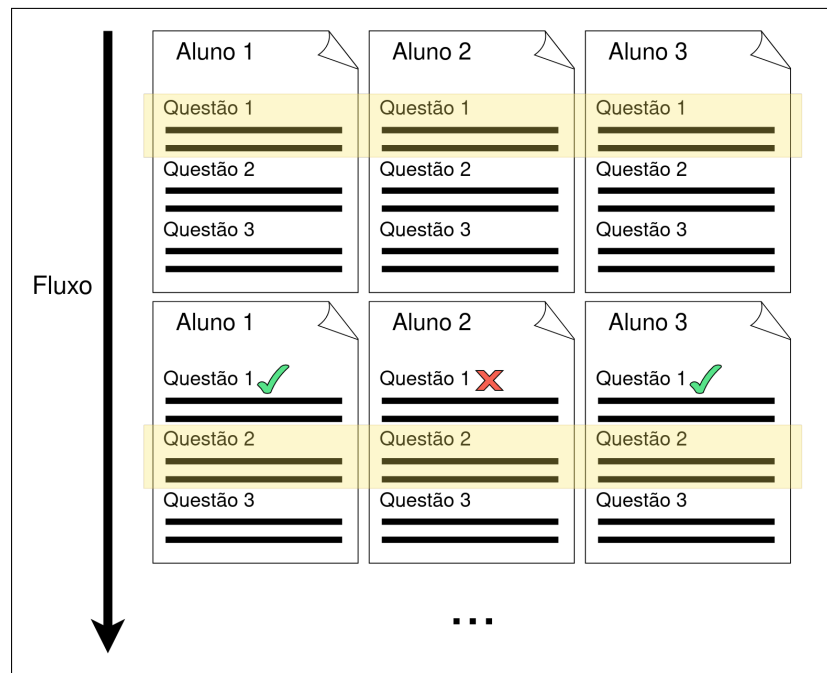


Figura 3.3: Fluxo de correção questão por questão

Fonte: Elaboração própria.

3.2 Arquitetura da aplicação

A partir do fluxo de ações descrito na seção anterior, foram identificadas as entidades básicas que compõem a aplicação: atividade, questão, entrega, resposta e grupo de respostas. Estas entidades se relacionam conforme o diagrama da Figura 3.4.

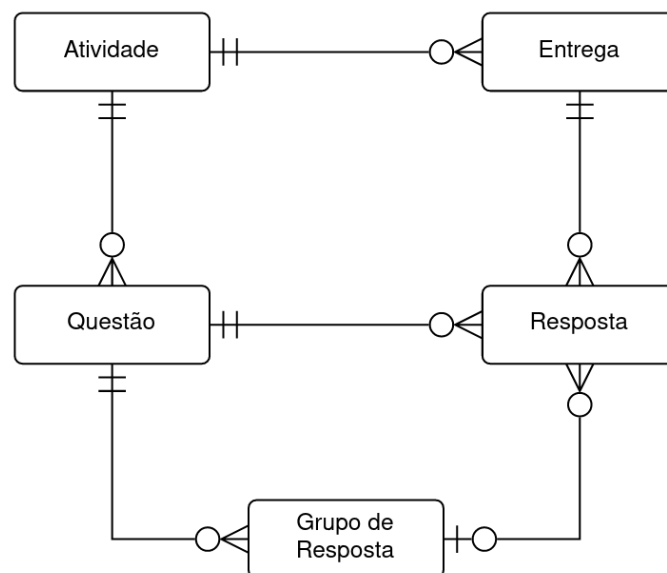


Figura 3.4: Modelo de Entidade-Relacionamento da aplicação.

Fonte: Elaboração própria.

Uma *Atividade* representa uma atividade avaliativa, composta por um título, nota

e imagem da folha de resposta. Cada Atividade possui *Questões*, que tem como atributos uma descrição, nota e coordenadas que determinam a região da imagem da folha de resposta onde a resposta da questão se encontra. Uma *Entrega* de uma Atividade tem um identificador único para o estudante e uma imagem da folha de resposta respondida. Para cada uma das *Questões*, são criadas as *Respostas*, a partir das Entregas. Uma Resposta possui uma imagem, gerada a partir da região da *Questão*, uma nota, um comentário, e pode ou não fazer parte de um *Grupo de Resposta*, que está vinculado a uma *Questão*, e também possui nota e comentário.

Para o desenvolvimento da plataforma foi escolhido o tipo aplicação web, tornando-a de fácil compatibilidade em diferentes dispositivos. Desta forma, também se torna possível que a aplicação seja implementada para uso local ou de forma distribuída, sendo acessada remotamente e de forma compartilhada por múltiplos usuários. A desvantagem desta escolha é uma maior vulnerabilidade a ataques e necessidade de conexão à rede, quando for utilizada de forma distribuída. Foram utilizados elementos de SPA, para garantir um uso mais dinâmico, fácil e intuitivo da aplicação.

Para o desenvolvimento do *front-end* da aplicação, foi escolhido o *framework* ReactJS (REACT, 2022), por ser uma boa ferramenta para a criação de SPA cujo modelo de componentes permite o isolamento e reusabilidade de elementos da plataforma. Para o *back-end* foi escolhido o *framework* Django (2023) da linguagem Python (2023).

Os componentes React “montam” a interface e funcionalidade das páginas. Enquanto o servidor irá acessar o sistema de arquivos para lidar com requisições do navegador de páginas e outros recursos, as APIs irão acessar o banco de dados para poder realizar as operações requisitadas pelos componentes React, como inserção ou obtenção de dados. Estas requisições são feitas para *endpoints* que correspondem à ação a ser realizada pelo servidor. O diagrama da Figura 3.5 ilustra esta comunicação na arquitetura da aplicação.

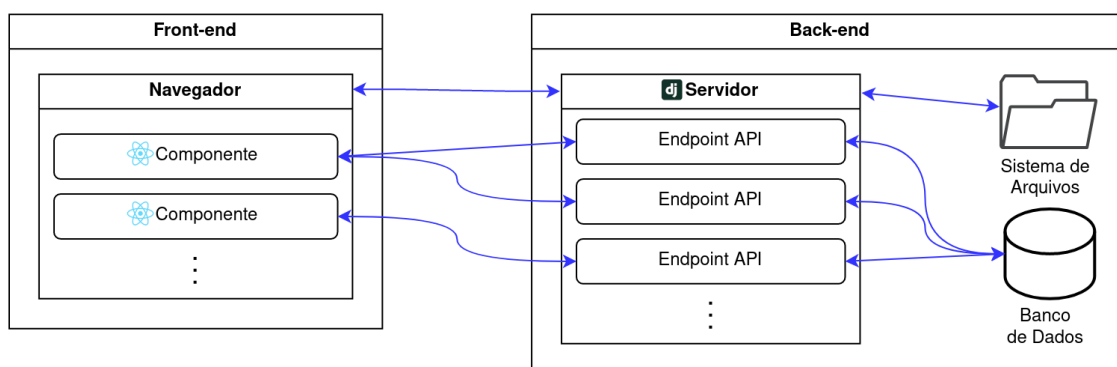


Figura 3.5: Arquitetura da aplicação.

Fonte: Elaboração própria

O *front-end* da aplicação é descrito na próxima seção, ao passo que seu *back-end*

é apresentado no próximo capítulo.

3.3 Front-end

Para satisfazer o fluxo de ações descrito na Seção 3.1, a aplicação possui 4 (quatro) páginas, que são: página principal, página de atividade, página de questão e página de entregas. O fluxo da navegação entre as páginas é ilustrado na Figura 3.6. A definição deste modelo se deve ao fato de que um número pequeno de páginas ajuda a tornar a navegação menos labiríntica e com caminhos menores.

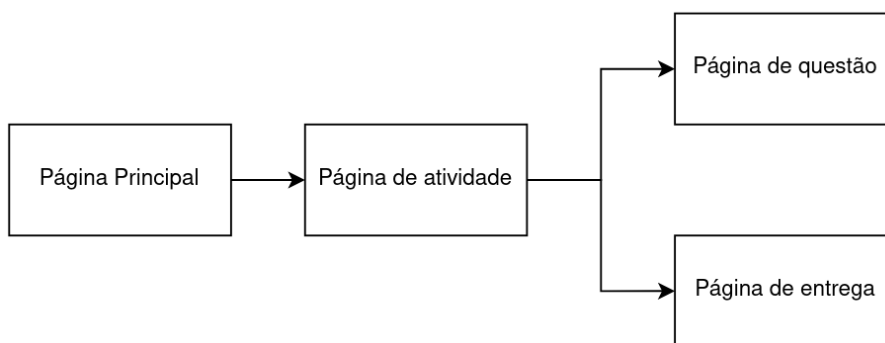


Figura 3.6: Diagrama de navegação das páginas da aplicação

Fonte: Elaboração própria

A página principal é aonde se cria e se acessa as avaliações, e também se obtém ou envia as correções das entregas. Na página de atividade as questões são criadas e as entregas inseridas. Através da página de entrega pode ser realizada a correção entrega por entrega, enquanto pela página de questão pode ser realizada a correção questão por questão, além do agrupamento de respostas.

3.3.1 Componentes desenvolvidos

Cada página é um componente composto de outros componentes. Para cada uma das entidades, exceto grupo de respostas, existe um componente de lista do mesmo, sendo que para questões e respostas cada elemento da lista é um componente à parte. Também existe para cada entidade, com exceção de respostas, um componente com formulário de criação, como para criar atividades e questões. Os componentes e sua respectiva hierarquização são mostradas na Figura 3.7. Com o objetivo de padronizar o código-fonte em um único idioma os componentes foram nomeados em inglês traduzindo atividade, questão, entrega, resposta e grupo de resposta para *assignment*, *field*, *submission*, *answer* e *answer group*, respectivamente.

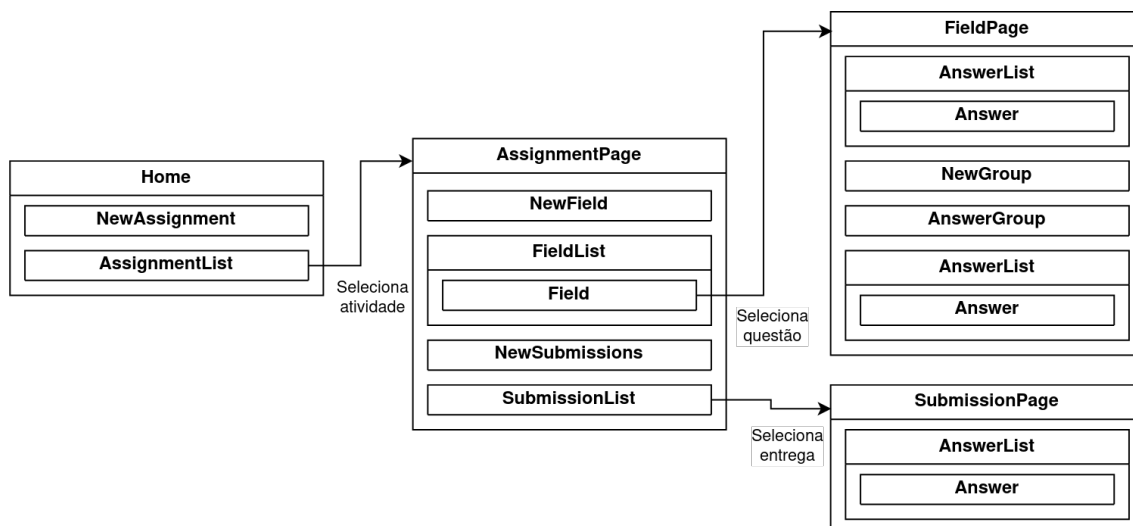


Figura 3.7: Diagrama de componentes da aplicação.

Fonte: Elaboração própria

Além dos componentes mostrados no diagrama da Figura 3.7, um componente extra chamado de `SelectionTool` também é utilizado. Este serve o propósito de reutilizar a ferramenta de seleção de regiões em imagens nos componentes `NewField` e `Answer`.

As requisições de obtenção de dados da aplicação são realizadas pelos componentes equivalentes às páginas, sendo que os dados são “transmitidos” para os componentes filhos, de forma que esses componentes sejam mais simples e contidos e, consequentemente, reaproveitáveis. Isso faz com que o estado da aplicação fique contido em apenas um componente, facilitando a manutenção e o controle. Contudo, esta estratégia torna os componentes relativos às páginas mais complexos, já que têm que lidar também com a comunicação com seus componentes filhos.

3.3.2 Detalhes sobre a implementação dos componentes

Nesta subseção a implementação dos componentes propostos será detalhada, tendo como foco o uso da aplicação. O código-fonte da aplicação pode ser obtido em <https://github.com/alexbatistaarantes/auxilio-avaliacao>, sendo que aqui serão discutidos apenas aspectos de mais alto nível.

3.3.2.1 Página principal

Na página principal, o formulário para criação de atividade é mostrado na parte superior, enquanto abaixo deste formulário está a lista de atividades já criadas. Para criar uma nova atividade, o usuário preenche o formulário de criação, inserindo o título e selecionando a imagem da folha de resposta no formato JPEG ou PNG, como mostrado na Figura 3.8.

Auxílio Avaliação

Nova atividade

Título

Folha de resposta da Atividade Folha de Resposta.png

Todas as atividades

Figura 3.8: *Exemplo de criação de uma atividade.*

Fonte: Elaboração própria.

A Figura 3.9 mostra a página com a atividade criada na lista de atividades. Mais detalhes sobre as atividades poderiam ser mostradas na lista, porém apenas o título e a imagem da folha de respostas são mostradas, visando a obtenção de uma interface mais enxuta.

Auxílio Avaliação

Nova atividade

Título

Folha de resposta da Atividade No file selected.

Todas as atividades

[Rede de Computadores](#)

Figura 3.9: *Página inicial com atividade criada.*

Fonte: Elaboração própria.

3.3.2.2 Página de atividade

Clicando em uma das atividades listadas na página principal, a página de atividade correspondente é apresentada. Conforme pode ser visto na Figura 3.10, a interface

da página de atividade é dividida em 3 (três) painéis. No primeiro, detalhes da aplicação são mostrados, sendo possível excluir a atividade, baixar a planilha de correção ou enviar as correções por e-mail. O segundo painel contém o formulário para criar novas questões e a lista de questões já criadas. No terceiro painel é possível inserir as entregas, sendo também onde a lista de entregas é mostrada. Foi escolhido mostrar todos esses elementos em uma mesma janela para evitar maiores navegações para visualizar o que se deseja.

The screenshot displays the 'Auxílio Avaliação' web application interface, which is divided into three main panels:

- Rede de Computadores:** This panel shows the activity details. It includes the title 'Rede de Computadores', the current value 'Valor da atividade: 0', and links for 'Baixar planilha de correção', 'Enviar correção por e-mail', and 'Excluir Atividade'. Below this, there is a preview of the activity sheet, showing the title, the activity name, and a list of questions with their respective options.
- Campos:** This panel is used for creating new questions. It features a 'Novo campo' section with a 'Questão 1' input field and a '2' dropdown menu. There is a 'Criar campo' button. Below this, there is a 'Folha de Resposta' section showing a preview of a question with multiple-choice options (a, b, c, d, e) and a 'Mostrar' button.
- Entregas:** This panel is used for managing submissions. It includes a button for 'Inserir entregas a partir do e-mail', a 'Nova Entrega' section with a 'Browse...' button and a 'Criar' button, and a 'Mostrar' button.

Figura 3.10: Exemplo de uma página de atividade.

Fonte: Elaboração própria.

O valor da atividade, mostrado no primeiro painel, é a nota máxima da atividade e é definido a partir da soma do valor de suas questões. Para criar as questões, o usuário deve inserir um título, um valor e a região de resposta da questão, na folha de resposta, a partir do formulário de nova questão da página de atividade, como é mostrado na Figura 3.10. Nas Figuras 3.11 e 3.12 é possível ver que cada questão criada aparece na lista de questões e a nota da atividade é atualizada com o novo valor.

Auxílio Avaliação
Rede de Computadores

Título: Rede de Computadores
Valor da atividade: 2
[Baixar planilha de correção](#)
[Enviar correção por e-mail](#)
[Excluir Atividade](#)

Matrícula: _____

Atividade - Redes de Computadores
Folha de Resposta

Questão 1 - Marque a resposta correta.

a) _____
b) _____
c) _____
d) _____
e) _____

Questão 2 - Escreva a resposta.

Questão 3 - Marque as afirmativas como verdadeiras (V) ou falsas (F).

I - ()
II - ()
III - ()

Campos
Novo campo
Legenda Points
Criar campo

Matrícula: _____

Atividade - Redes de Computadores
Folha de Resposta

Questão 1 - Marque a resposta correta.

a) _____
b) _____

Mostrar

[Questão 1](#)
2 points

Entregas
[Inserir entregas a partir do e-mail](#)
Nova Entrega
[Browse...](#) No files selected. [Criar](#)

Figura 3.11: Página de uma atividade com uma única questão criada.

Fonte: Elaboração própria.

Auxílio Avaliação
Rede de Computadores

Título: Rede de Computadores
Valor da atividade: 10
[Baixar planilha de correção](#)
[Enviar correção por e-mail](#)
[Excluir Atividade](#)

Matrícula: _____

Atividade - Redes de Computadores
Folha de Resposta

Questão 1 - Marque a resposta correta.

a) _____
b) _____
c) _____
d) _____
e) _____

Questão 2 - Escreva a resposta.

Questão 3 - Marque as afirmativas como verdadeiras (V) ou falsas (F).

I - ()
II - ()
III - ()

Campos
Legenda Points
Criar campo

Matrícula: _____

Atividade - Redes de Computadores
Folha de Resposta

Questão 1 - Marque a resposta correta.

a) _____
b) _____

Mostrar

[Questão 1](#)
2 points
[Questão 2](#)
6 points
[Questão 3](#)
2 points

Entregas
[Inserir entregas a partir do e-mail](#)
Nova Entrega
[Browse...](#) No files selected. [Criar](#)

Figura 3.12: Página de a atividade com três questões criadas.

Fonte: Elaboração própria.

A interface do processo de inserção das entregas dos alunos necessita de poucas etapas. Como mencionado no início desta seção, o formulário de inclusão de entregas está no terceiro painel da página de atividade. Para inserir as entregas recebidas a partir do e-mail, é necessário apenas clicar no botão “Inserir entregas a partir do e-mail”, para que uma requisição seja feita para o *back-end*, solicitando a obtenção dos e-mails com as entregas para a atividade, mediante cadastro da conta nas configurações do servidor. Uma função irá se conectar ao servidor de e-mail solicitando mensagens não lidas, aonde serão

obtidas as entregas a partir das imagens anexadas aos e-mails que possuírem o nome da atividade como título.

Para inserir as entregas a partir da seleção das imagens salvas no computador, é preciso clicar no campo de seleção de arquivos, inserir os arquivos equivalentes, como mostrado na Figuras 3.13.

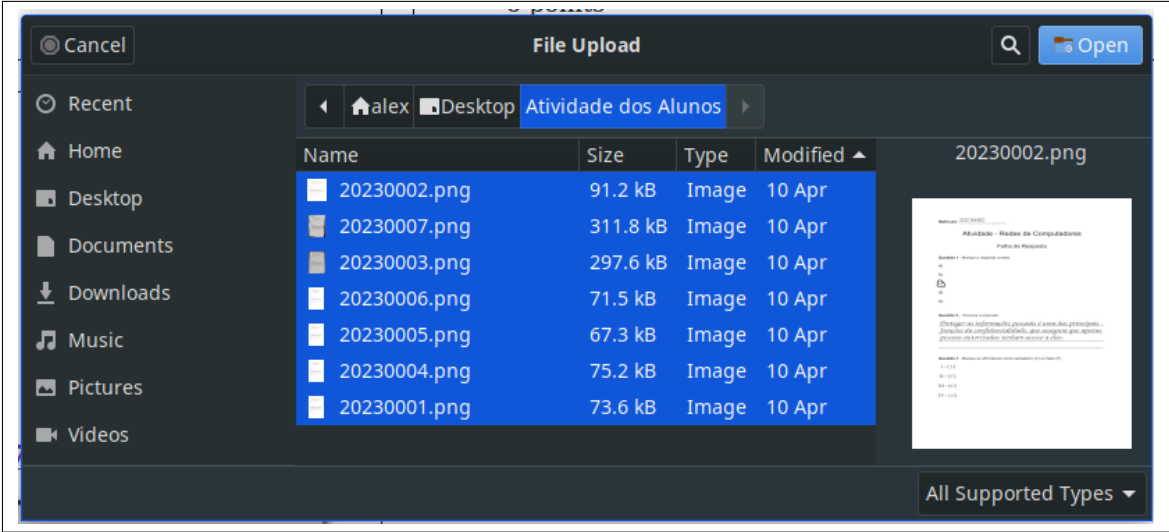


Figura 3.13: Selecionando imagens das entregas.

Fonte: Elaboração própria

Após a seleção dos arquivos, a página exibe quantos arquivos foram selecionados. Como próxima etapa é necessário clicar no botão “Clicar” exibido ao lado da mensagem, como mostrado na Figura 3.14.



Figura 3.14: Imagens das entregas selecionadas.

Fonte: Elaboração própria

As entregas aparecerão na lista, como é mostrado no lado esquerdo da Figura 3.15, contendo a imagem da entrega, um link para baixar a correção individual, e um identificador único do aluno, extraído a partir do nome da imagem inserida, ou do endereço de e-mail que a enviou. Clicando em uma entrega, a página da entrega é acessada. Novamente, prioriza-se uma interface mais enxuta para a lista de entregas, em detrimento de uma mais detalhada.

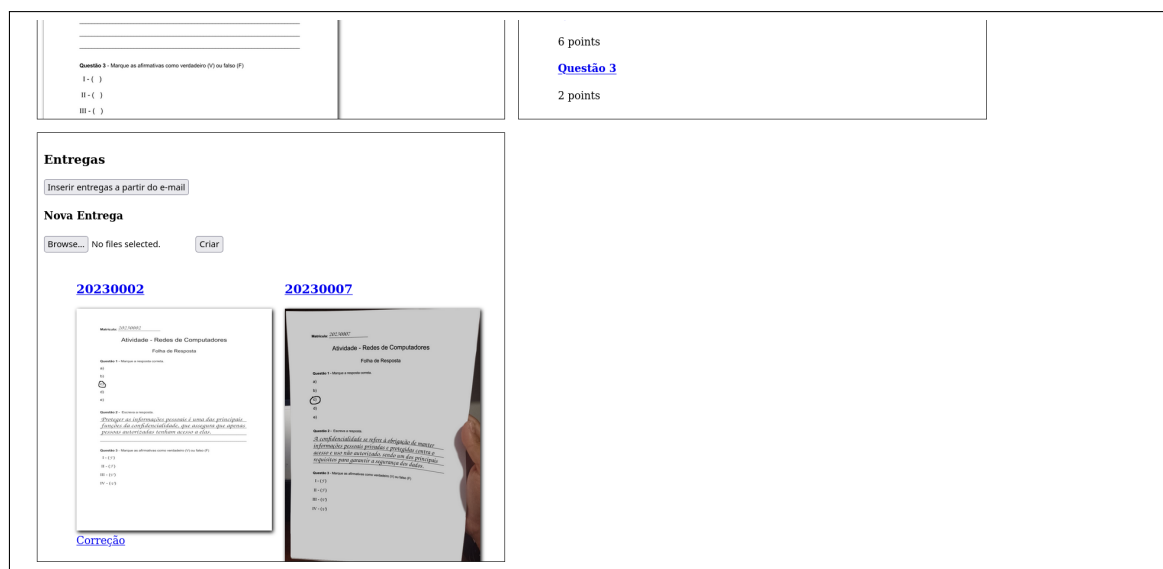


Figura 3.15: Página da atividade com as entregas inseridas.

Fonte: Elaboração própria.

3.3.2.3 Página de entrega

A página de entrega é composta por um painel com detalhes da entrega, um botão para baixar o arquivo de correção, e outro para remover a entrega. Além disso, há um painel com uma lista de respostas desta entrega. A Figura 3.16 mostra um exemplo desta página. Cada resposta da lista possui campos para definir a nota e um comentário, tornando possível a correção usando o tipo entrega por entrega.

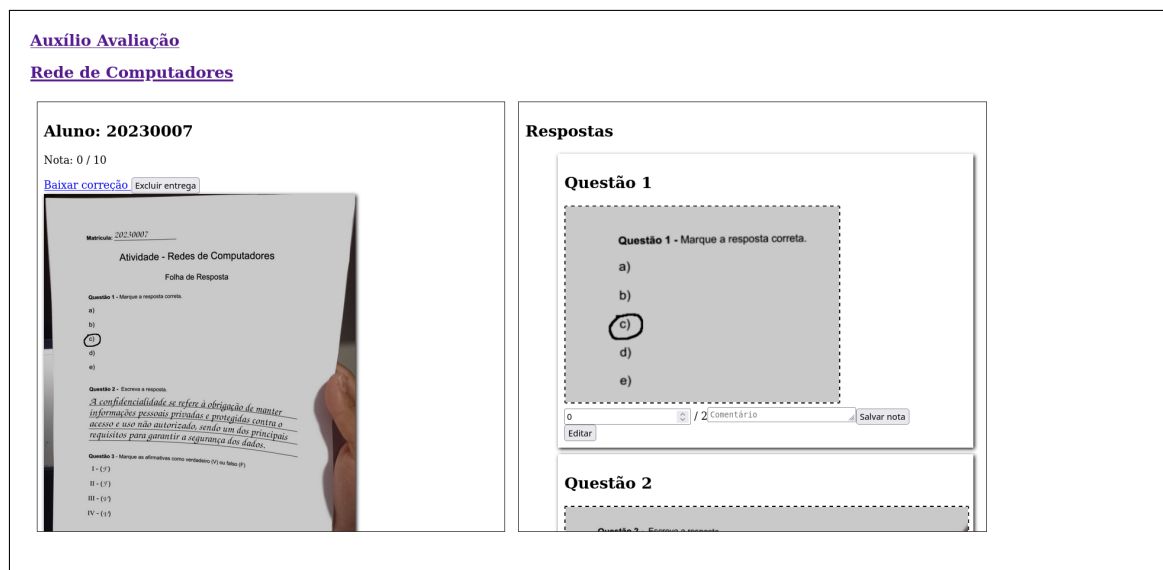


Figura 3.16: Página de uma entrega

Fonte: Elaboração própria

Cada resposta da lista também possui um formulário para editar a região da resposta, na imagem da entrega. Este formulário existe para corrigir eventuais desalinhamentos presentes em entregas com desenquadramento na imagem. Para realizar este enquadramento, basta que o usuário clique no botão “Editar”, presente logo abaixo da resposta da lista. A Figura 3.17 demonstra a utilização deste formulário.

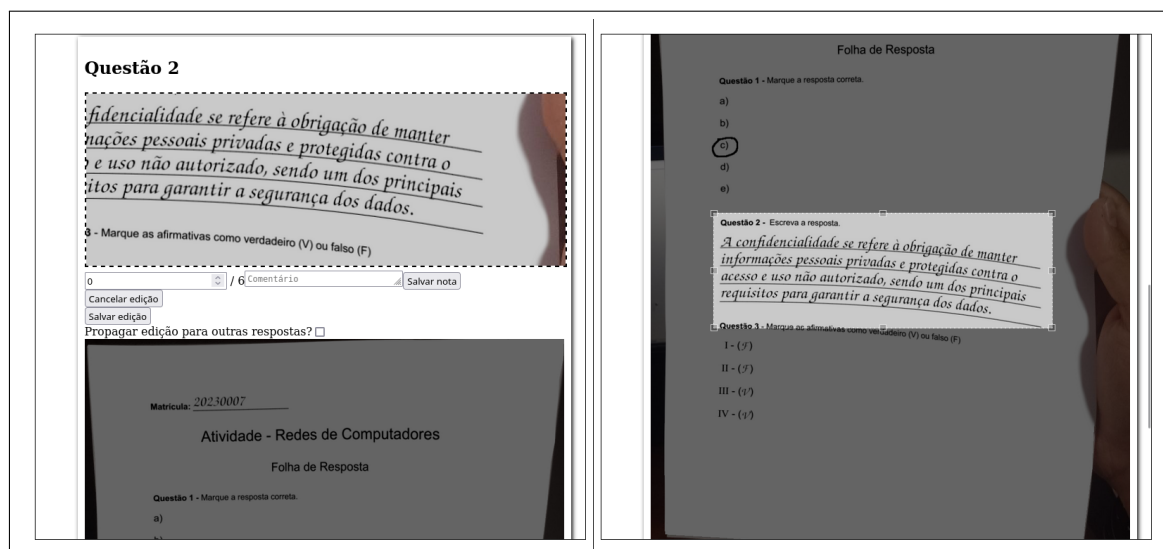


Figura 3.17: Editando a região de uma resposta.

Fonte: Elaboração própria.

Uma caixa de seleção é inclusa no formulário para que o usuário escolha se deseja que a modificação realizada na região seja proporcionalmente aplicada nas outras

respostas da mesma entrega. Na figura 3.18 é possível ver o antes e depois de um enquadramento.

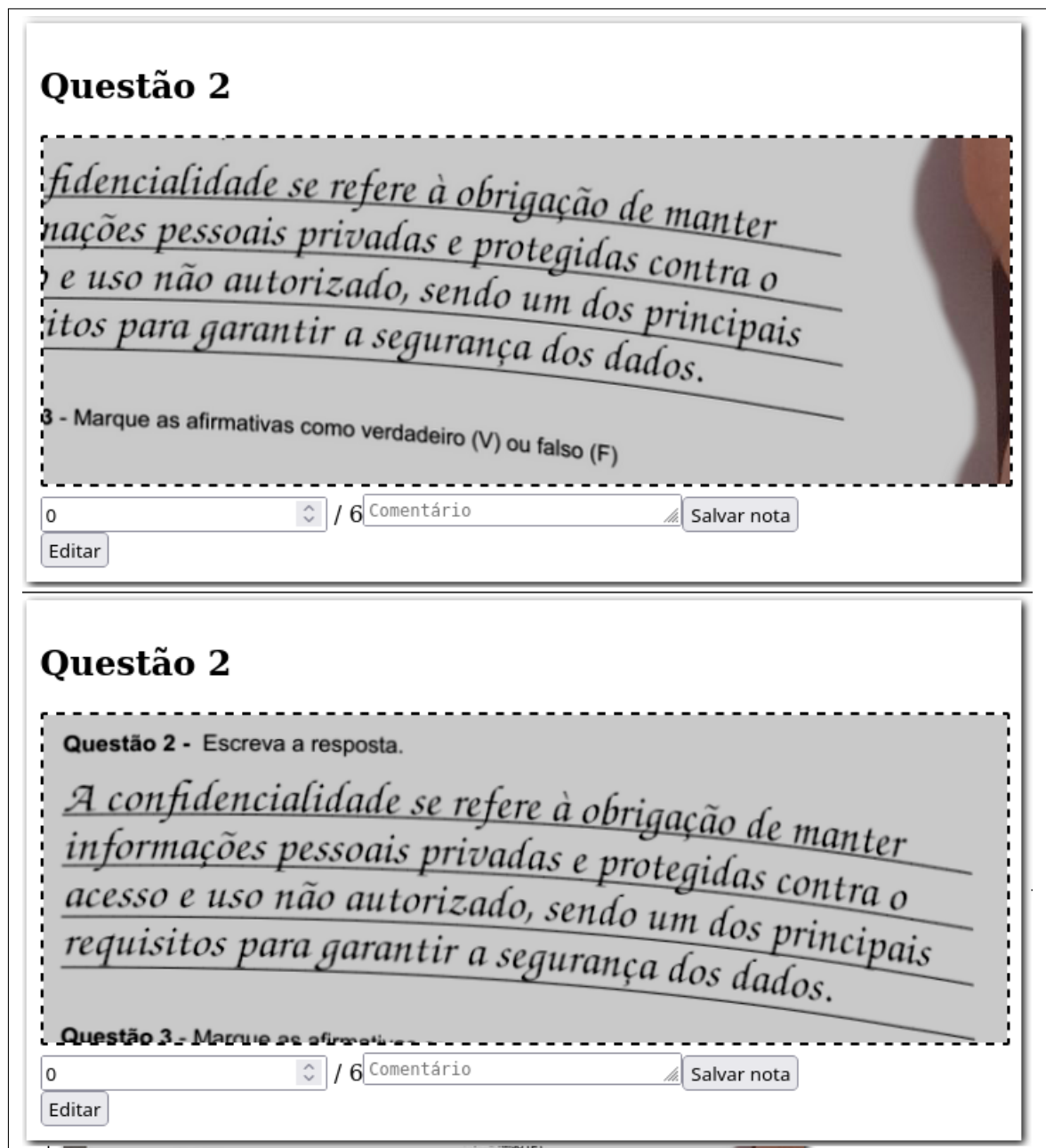


Figura 3.18: Resposta desalinhada (parte superior) e resposta com região corrigida (parte inferior).

Fonte: Elaboração própria.

3.3.2.4 Página de questão

Como pode ser visto na Figura 3.19, a página de questão possui dois painéis. No primeiro existe um botão para deletar a questão e uma lista de respostas, contendo todas as respostas para essa questão que não estão em nenhum grupo de resposta. É também neste

painel que uma correção do tipo questão por questão é realizada, percorrendo e corrigindo cada resposta.

Auxílio Avaliação
Rede de Computadores

Questão 2
Deletar campo

Questão 2 - Escreva a resposta.
Proteger as informações pessoais é uma das principais funções da confidencialidade, que assegura que apenas pessoas autorizadas tenham acesso a elas.

0 / 6 Comentário Salvar nota
Editar

Questão 3 - Marque as afirmativas corretas.
A confidencialidade se refere à obrigação de manter informações pessoais privadas e protegidas contra o acesso e uso não autorizado, sendo um dos principais requisitos para garantir a segurança dos dados.

0 / 6 Comentário Salvar nota
Editar

Grupos
Novo Grupo
Não citou combate a divulgaçã Criar

Figura 3.19: Exemplo de página de questão.

Fonte: Elaboração própria.

O segundo painel (parte direita da Figura 3.19) consiste em um formulário para criação de um novo grupo de respostas. Quando um grupo de respostas criado está selecionado, um outro formulário para definir a nota e comentário para este grupo é apresentado, com uma lista das respostas presentes no mesmo. Esta interface de dois painéis permite que respostas de uma questão possam ser visualizadas e corrigidas em uma mesma tela, evitando navegações, além de tornar mais fácil a seleção múltipla de respostas, para serem inclusas ou excluídas de um grupo.

Para criar um grupo, deve ser definida uma descrição, como apresentado na Figura 3.19. Ao clicar no botão “Criar”, o grupo aparece abaixo, porém com a lista vazia como mostrado na Figura 3.20.

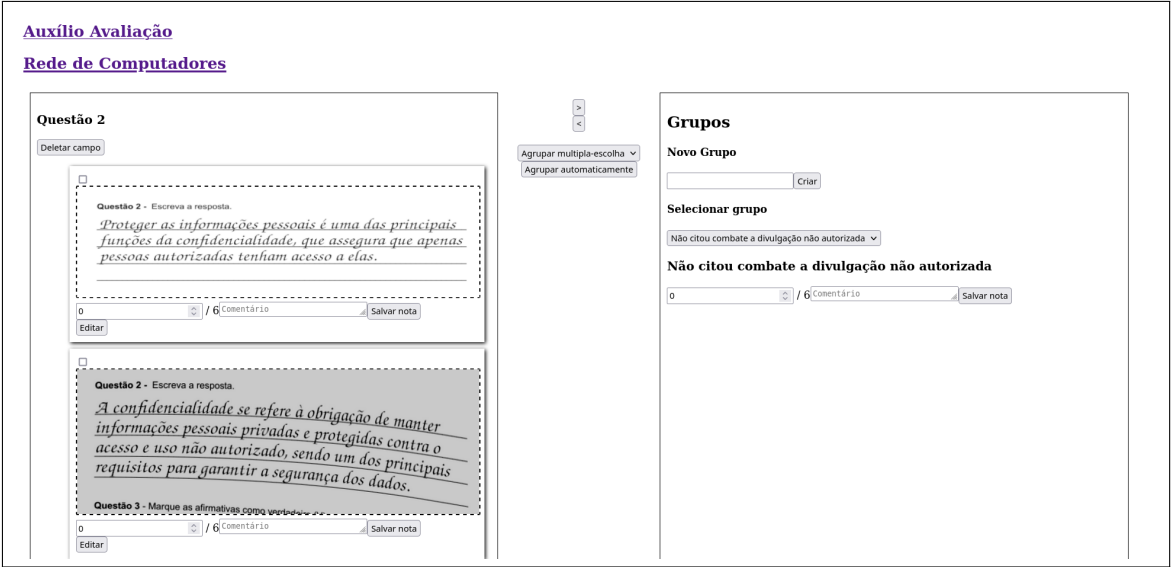


Figura 3.20: Exemplo de criação de um grupo de respostas.

Fonte: Elaboração própria

Para inserir respostas no grupo, deve-se seleciona-las da lista do primeiro painel, como mostrado na Figura 3.21, e clicar no botão com a seta para a direita, presente entre os painéis. Após isso, as respostas selecionadas serão incluídas no grupo, como mostra a Figura 3.22. De maneira equivalente, ao selecionar as respostas da lista de um grupo e clicar no botão com a seta para a esquerda as respostas selecionadas são retiradas do grupo.

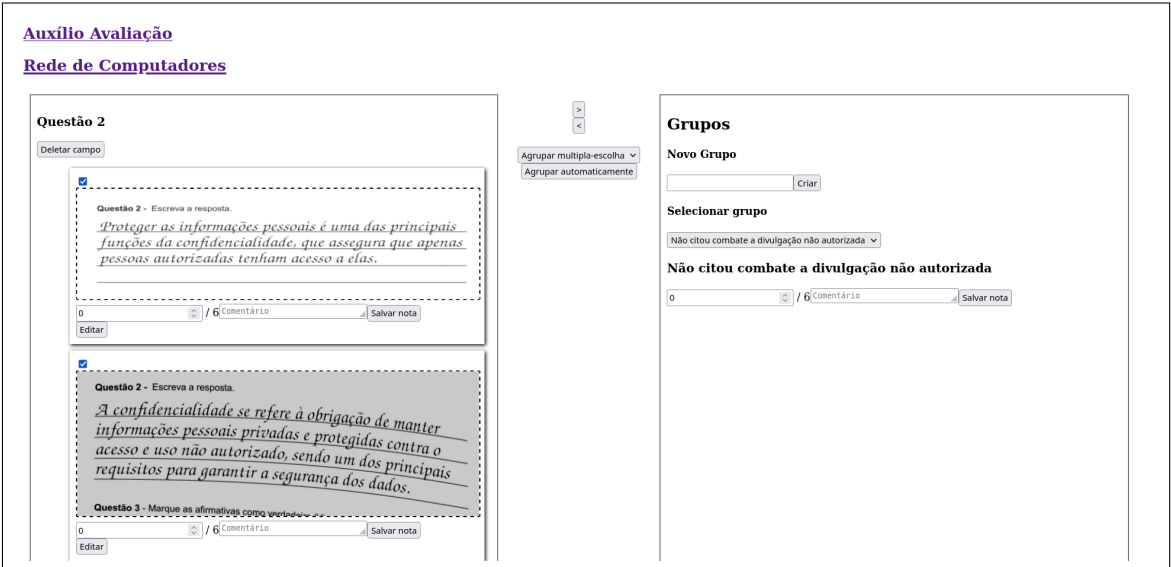


Figura 3.21: Exemplo de seleção de respostas para inserir em um grupo.

Fonte: Elaboração própria.

The screenshot shows the 'Auxílio Avaliação' interface. On the left, under 'Rede de Computadores', there are two question cards for 'Questão 2'. The first card shows a question about confidentiality and has a selected answer: 'Garantindo que apenas pessoas autorizadas tenham acesso às informações, a confidencialidade é uma medida de segurança que tem como objetivo proteger dados sensíveis contra divulgação não autorizada a terceiros.' The second card shows a question about confidentiality and has a selected answer: 'A confidencialidade é uma medida de segurança que visa proteger informações sensíveis contra divulgação não autorizada a terceiros.' On the right, the 'Grupos' section shows a 'Novo Grupo' button and a 'Selecionar grupo' dropdown. Below this, there is a group of selected answers for 'Questão 2' with a score of 0 and a comment field.

Figura 3.22: Respostas selecionadas após serem inseridas em um grupo.

Fonte: Elaboração própria.

Para dar nota para as respostas agrupadas, é necessário preencher o formulário do grupo com a nota e um comentário, como é mostrado na Figura 3.23.

The screenshot shows the 'Auxílio Avaliação' interface. On the left, under 'Rede de Computadores', there are two question cards for 'Questão 2'. The first card shows a question about confidentiality and has a selected answer: 'Garantindo que apenas pessoas autorizadas tenham acesso às informações, a confidencialidade é uma medida de segurança que tem como objetivo proteger dados sensíveis contra divulgação não autorizada a terceiros.' The second card shows a question about confidentiality and has a selected answer: 'A confidencialidade é uma medida de segurança que visa proteger informações sensíveis contra divulgação não autorizada a terceiros.' On the right, the 'Grupos' section shows a 'Novo Grupo' button and a 'Selecionar grupo' dropdown. Below this, there is a group of selected answers for 'Questão 2' with a score of 3 and a comment field. A tooltip is visible over the 'Salvar nota' button, indicating that the user must cite the concept of confidentiality, which also includes protecting sensitive data from unauthorized disclosure.

Figura 3.23: Definindo nota e comentário para um grupo de respostas.

Fonte: Elaboração própria.

Ao clicar no botão “Salvar nota”, a nota e o comentário são aplicados para cada uma das respostas que fazem parte do grupo como mostrado na Figura 3.24.

The image shows a web interface with three question cards. Each card has a title 'Questão 2 - Escreva a resposta.', a question text, a student's answer, a score, and a comment. The first card shows a question about confidentiality, a student's answer, a score of 0, and a comment. The second card shows a question about confidentiality, a student's answer, a score of 0, and a comment. The third card shows a question about confidentiality, a student's answer, a score of 3, and a comment.

Figura 3.24: Respostas do grupo com a nota e comentário definidos.

Fonte: Elaboração própria.

Após finalizada a correção, é possível obter o arquivo com o resultado para cada uma das entregas. Este arquivo contém a nota final e a nota e comentário de cada uma das respostas, como mostrado no Apêndice D, na página da atividade ou páginas das entregas. Também é possível encaminhar por e-mail o arquivo de correção, para os alunos em que o identificador único for o seu endereço de e-mail, clicando no botão “Enviar correção por e-mail” na página da atividade.

Back-end e Validação

Continuando a apresentação da aplicação iniciada no capítulo anterior, este capítulo descreve detalhes de implementação do *back-end*. A implementação foi baseada no *framework* Django devido à sua usabilidade e ampla variedade de bibliotecas e ferramentas disponíveis, características estas que propiciam extensibilidade para a aplicação.

Além de descrever o *back-end*, este capítulo também apresenta uma avaliação realizada para validar a aplicação desenvolvida.

4.1 Persistência de dados

Para interagir com o banco de dados necessário para a persistência dos dados no Django, utiliza-se as classes `Models`. Estas classes mapeiam os dados da aplicação para o banco de dados, permitindo que as ações sejam realizadas utilizando-se de programação orientada a objetos, ao invés de consultas em *Structured Query Language* (SQL).

Acompanhando as entidades adotadas no *front-end*, foram definidas 5 (cinco) entidades, apresentadas no diagrama da Figura 4.1. Novamente, o nome das entidades e seus atributos foram definidos em inglês, para padronização do código.

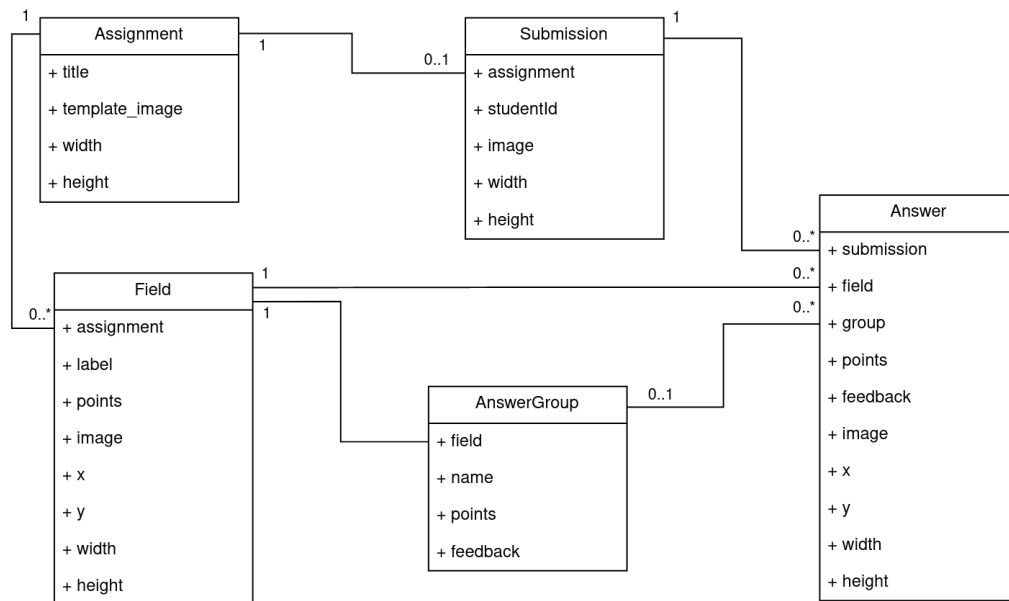


Figura 4.1: Diagrama de classes do modelo no back-end.

Fonte: Elaboração própria.

4.2 Tratamento de dados de requisições

Como as requisições recebidas pelo cliente para criação e edição de entidades são no formato *JavaScript Object Notation* (JSON), é necessário converter os atributos recebidos para tipos nativos do Python. Para isso, são utilizadas classes chamadas de *Serializers*, do *framework* Django REST framework (ENCODE OSS LTD, 2023). Neste caso é criado um serializador para cada uma das entidades do modelo, como mostrado no Código 4.1 para a entidade *Assignment*.

Código 4.1 Exemplo de um serializador

```

1 class AssignmentSerializer(serializers.ModelSerializer):
2     fields = serializers.PrimaryKeyRelatedField(many=True, read_only=True)
3     total_points = serializers.FloatField(read_only=True)
4
5     class Meta:
6         model = Assignment
7         fields = '__all__'
  
```

4.3 Implementação de operações

Cada mensagem enviada ao servidor tem como objetivo requisitar um serviço como o envio de um determinado conjunto de dados, a criação de novas instâncias de uma entidade, a alteração de instâncias já existentes ou alguma outra funcionalidade. A execução destes serviços é realizada em classes ou funções chamadas de `views`.

Para cada entidade existe uma `view`, usada para criar, alterar, ou deletar uma instância. Também existem `views` para obter listas de instâncias de uma entidade usando como parâmetro outra entidade como, por exemplo, respostas com relação a uma entrega específica.

O modelo de implementação adotado faz com que, para se obter os dados de uma página, seja necessário a realização de múltiplas requisições para diferentes *endpoints*. Contudo, como vantagem os *endpoints* são mais contidos, com foco em apenas uma entidade, favorecendo futuras extensões da aplicação.

Na implementação foram utilizadas classes chamadas de `ModelViewSet`, oferecidas pelo Django REST Framework, que ajudam a reduzir redundâncias e diminuir o código das `views` pois servem como um arcabouço que implementa funções que lidam com métodos de criação, leitura, atualização e remoção, que são implementados de forma equivalente para todas as entidades. No Código 4.2 uma `ViewSet` para manipulação de atividades e outra para campos de uma determinada atividade são implementadas.

Código 4.2 Exemplo de *ModelViewSets*

```
1 class AssignmentViewSet(viewsets.ModelViewSet):  
2     queryset = Assignment.objects.all()  
3     serializer_class = AssignmentSerializer  
4  
5 class AssignmentFieldsViewSet(viewsets.ModelViewSet):  
6     serializer_class = FieldSerializer  
7  
8     def get_queryset(self):  
9         assignment_id = self.kwargs['assignment_id']  
10        return Field.objects.filter(assignment__id=assignment_id)
```

Também são em `views` que são implementadas outras funcionalidades da aplicação como o envio dos arquivos através de e-mail, a obtenção de entregas por e-mail e a geração do arquivo de correção. O Código 4.3 apresenta a implementação desta última funcionalidade.

Código 4.3 View utilizada para obtenção de um arquivo de correção

```
1 def download_submission_grading(request, submission_id):
2
3     submission = get_object_or_404(Submission, pk=submission_id)
4
5     pdf = get_submission_grading(submission)
6
7     response = HttpResponse(bytes(pdf), content_type='application/pdf')
8     response['Content-Disposition'] = f'inline; filename="{submission.assignment.title}-{
9         submission.studentId}.pdf"'
10
11     return response
```

Para que as *views* se tornem acessíveis para o *front-end* na forma de uma API cada uma deve ser vinculada a uma *Uniform Resource Locator* (URL) que é utilizada como o endereço das requisições HTTP. Desta forma, o caminho percorrido pelas requisições e respostas no *back-end* pode ser ilustrado conforme o diagrama da Figura 4.2.

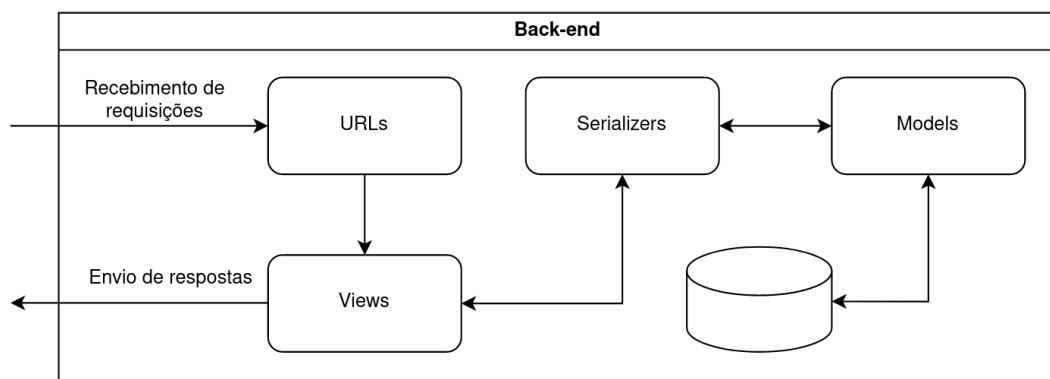


Figura 4.2: Fluxo das mensagens no back-end.

Fonte: Elaboração própria.

Cada uma das partes do *back-end* é implementada em um arquivo próprio nomeado de acordo com a parte em questão como, por exemplo, um arquivo *models.py* para os *models* e *views.py* para as *views*. Essa arquitetura beneficia não só a extensibilidade, como também a legibilidade e manutenção da aplicação, já que esta separação torna mais fácil a navegação do código e torna as partes menos acopladas, mais modulares, algo que é encorajado pelo *framework* Django, como é apontado nas suas filosofias de projeto (DJANGO, 2023).

Outra estratégia utilizada para promover a extensibilidade da aplicação foi a definição de uma interface que descreve funcionalidades que visam o agrupamento automático das respostas em grupos de resposta. As implementações desta interface

são chamadas de *Agrupadores* e constituem funções que identificam ou interpretam o conteúdo das imagens das respostas de uma questão, visando classificá-las em grupos usando critérios como semelhança, igualdade ou outros. Estes agrupadores podem ser implementados usando técnicas de Visão Computacional, como a solução proposta em Hsiao (2016) e a implementada na plataforma Gradescope (2022).

O Código 4.4 apresenta um exemplo de agrupador. Neste caso, `classifyMCQ` é um agrupador que visa classificar respostas de uma questão do tipo múltipla escolha. A função recebe uma lista de modelos de resposta de uma mesma questão e, para cada uma, identifica por meio da imagem da resposta qual opção o aluno respondeu. A função, então, insere respostas com a mesma opção selecionada em um mesmo grupo de respostas. A função `identifySelectedOption` retorna a opção marcada na resposta, implementando um modelo de detecção de questões de múltipla escolha a partir da imagem.

Código 4.4 *Exemplo de um agrupador*

```
1 def classifyMCQ(answers):
2
3     groups = {}
4     for answer in answers:
5         try:
6             content = identifySelectedOption( answer.image )
7             if content not in groups.keys():
8                 groups[content] = []
9
10            group[content].append(answer)
11        except:
12            pass
13
14    field = answers[0].field
15    for content, group_answers in groups.items():
16        try:
17            group = AnswerGroup(name=content, field=field)
18            group.save()
19        except:
20            group = field.groups.get(name=content)
21    for answer in group_answers:
22        answer.group = group
23        answer.save()
```

4.4 Avaliação da aplicação

Para verificar se a solução desenvolvida traz benefícios para a correção de atividades, foi realizada uma avaliação baseada em uma pesquisa de experiência de uso. A pesquisa considerou um público formado por 23 docentes do IFG e de outras instituições. Contudo, dentro do tempo disponibilizado para realizar a avaliação, houve resposta de apenas 1 (um). É evidente que este universo de resposta não é suficiente para validar a

aplicação, mas os resultados são apresentados como forma de demonstrar a percepção de uso da aplicação junto ao seu público-alvo.

A aplicação foi implantada em uma máquina virtual na nuvem da *Amazon Web Services* (AWS) e a URL foi disponibilizada ao docente. Além disso, foi fornecido a atividade do Apêndice B, seu gabarito e 7 (sete) entregas fictícias semelhantes à do Apêndice A. O experimento consistiu em solicitar ao docente para realizar a correção das entregas e, em seguida, preencher um formulário aonde era especificado o nível de concordância com afirmações positivas de experiência com o uso da aplicação, majoritariamente em comparação com o processo feito de forma manual. Seguindo o formato da escala Likert (JOSHI *et al.*, 2015), para cada uma das afirmações foram fornecidas opções de concordância ou discordância, total ou parcial, ou indecisão. Também foi disponibilizado um espaço para realizar comentários extras. As afirmações e a respectiva resposta foram o seguinte:

1. “Em comparação com a correção manual, a ferramenta economiza tempo na correção das avaliações”: Concordo parcialmente.
2. A ferramenta consegue lidar com um grande volume de avaliações de forma mais eficiente do que a correção manual: Concordo totalmente.
3. A ferramenta reduz a carga de trabalho dos professores ao assumir tarefas repetitivas e mecânicas: Concordo parcialmente.
4. A ferramenta é capaz de detectar e corrigir erros comuns cometidos pelos alunos de forma mais consistente do que a correção manual: Indeciso.
5. A ferramenta melhora a consistência e a objetividade das correções em comparação com as correções feitas manualmente: Concordo parcialmente.
6. A ferramenta facilita a identificação de padrões de erros e áreas de dificuldade comuns entre os alunos: Concordo totalmente.
7. A ferramenta fornece resultados de correção mais precisos e confiáveis do que a correção manual: Indeciso.
8. A ferramenta permite que os professores se concentrem mais em tarefas de análise e feedback construtivo, em vez de gastar tempo na correção detalhada das avaliações: Concordo parcialmente.
9. A ferramenta permite que os professores personalizem os critérios de correção de acordo com as necessidades e objetivos específicos de cada avaliação: Concordo parcialmente.

As questões buscaram verificar a percepção que o respondente teve a respeito da efetividade da aplicação em auxiliar na correção, em pontos como tempo para realização, alto volume de entrega, e consistência.

A economia do tempo de correção foi um ponto de concordância parcial pelo respondente. O mesmo acrescenta em seu comentário que um tempo extra surge na

correção através da aplicação devido à necessidade de configuração da ferramenta. Em relação ao ganho de eficiência em lidar com um grande volume de avaliações, apesar da concordância total, o respondente levanta a possível dificuldade de navegação pela aplicação que pode surgir neste cenário, devido à interface não separar as listas em páginas.

Com relação à redução da carga de trabalho, o docente concorda com a redução e sugere, como forma de se obter uma maior assistência da aplicação, a implementação de uma nova funcionalidade que realize a correção de forma automática, a partir do uso de detecção de imagem.

O respondente foi indeciso em relação ao resultado de correção ter maior precisão e confiabilidade devido ao auxílio da aplicação. Porém, concordou com a melhoria da consistência na correção e a maior facilidade de identificar padrões de erros entre alunos.

Considerações Finais

A etapa avaliativa por meio de provas no processo de ensino-aprendizagem geralmente é uma tarefa repetitiva para o docente que pode ser bastante cansativa e suscetível a inconsistências que podem acabar por prejudicar a avaliação. Diante disso, o presente trabalho teve o propósito de desenvolver uma ferramenta que auxilia o processo de correção de atividades avaliativas aplicadas em papel, oferecendo maior agilidade e padronização.

Diversos trabalhos desenvolveram soluções digitais com o mesmo intuito de auxiliar este processo. Contudo, a aplicação desenvolvida neste trabalho constitui uma solução de código aberto e extensível, com foco na implementação da lógica da aplicação.

A plataforma foi desenvolvida no formato de aplicação web seguindo a abordagem de SPA. Como destaque, opção de receber as imagens das entregas através do e-mail e enviar a correção pelo mesmo auxilia a obter uma maior efetividade em lidar com um grande volume de entregas.

A interface de avaliação implementada permite que o fluxo de correção das entregas seja realizado percorrendo as respostas de todas as entregas questão por questão, que proporciona o foco em uma questão por vez e uma melhor comparação das respostas entre alunos, possibilitando uma correção mais coerente. Associado à funcionalidade de agrupar respostas, este formato também reduz redundâncias na correção, já que esta é realizada uma única vez para todas as respostas semelhantes.

Todavia, é importante ressaltar que na solução implementada a correção pode ser prejudicada devido as entregas se limitarem à apenas uma imagem, restringindo assim o escopo da atividade. Dessa forma, o tratamento de atividades com múltiplas páginas é tido como um trabalho futuro.

Outra desvantagem evidente é a necessidade de transformar em imagem as entregas realizadas, condicionando ao professor a realização desta tarefa nos casos em que o aluno deva ser privado de qualquer dispositivo no ato da entrega. Isso também pode ocasionar em um tempo extra para configuração da ferramenta.

Os *frameworks* utilizados para o desenvolvimento da aplicação beneficiam a extensibilidade da plataforma, tornando mais fácil futuros incrementos no trabalho. Com

base nesta facilidade, outros pontos de melhoria na aplicação podem ser destacados. Dentre estes, seria de grande adição a implementação de um componente de gerenciamento de usuários que permitiria que formas mais gerenciáveis de recebimento e entrega de correções fossem desenvolvidas. Outro incremento que agregaria bastante seria a implementação de algoritmos de classificação e agrupamento automatizado de respostas, possivelmente empregando técnicas de inteligência artificial, reduzindo o esforço necessário para agrupar respostas. Por fim, a melhoria na interface visando torná-la mais atrativa esteticamente e mais intuitiva constitui outra melhoria imediata.

Referências Bibliográficas

AGOSTI, Maristella. An overview of hypertext. *In: _____. Information Retrieval and Hypertext*. Boston, MA: Springer US, 1996. p. 27–47. ISBN 978-1-4613-1373-1. Disponível em: https://doi.org/10.1007/978-1-4613-1373-1_2. 22

BERNERS-LEE, Tim; CAILLIAU, Robert; LUOTONEN, Ari; NIELSEN, Henrik Frystyk; SECRET, Arthur. The world-wide web. **Commun. ACM**, Association for Computing Machinery, New York, NY, USA, v. 37, n. 8, p. 76–82, aug 1994. ISSN 0001-0782. Disponível em: <https://doi.org/10.1145/179606.179671>. 21

BLOOMFIELD, Aaron. Evolution of a digital paper exam grading system. *In: 2010 IEEE Frontiers in Education Conference (FIE)*. [S.l.: s.n.], 2010. p. T1G–1–T1G–6. 17

CLASSROOM. Google, 2022. Disponível em: <https://edu.google.com/workspace-for-education/classroom/>. Acesso em: 02 jul. 2023. 16

CROWDMARK. Crowdmark, Inc, 2022. Disponível em: <https://crowdmark.com/>. 17

DJANGO. Django Software Foundation, 2023. Disponível em: <https://djangoproject.com>. Acesso em: 02 jul. 2023. 23, 28, 45

EICH, Brendan. **Javascript**. 1995. 24

ENCODE OSS LTD. **Django REST framework**. 2023. Disponível em: <https://www.django-rest-framework.org/>. Acesso em: 02 jul. 2023. 43

EVALBEE. Evalbee Pvt. Ltd, 2022. Disponível em: <https://evalbee.com/>. 16

FIELDING, Roy; GETTYS, Jim; MOGUL, Jeffrey; FRYSTYK, Henrik; MASINTER, Larry; LEACH, Paul; BERNERS-LEE, Tim. **Hypertext transfer protocol – HTTP/1.1**. [S.l.: s.n.], 1999. 22

FIELDING, R.; GETTYS, J.; MOGUL, J.; FRYSTYK, H.; MASINTER, L.; LEACH, P.; BERNERS-LEE, T. **RFC 2616, Hypertext Transfer Protocol – HTTP/1.1**. 1999. Disponível em: <https://www.ietf.org/rfc/rfc2616.txt>. Acesso em: 02 jul. 2023. 22

GRADESCOPE. Turnitin, LLC, 2022. Disponível em: <https://www.gradescope.com/>. 17, 46

HSIAO, I-Han. Mobile grading paper-based programming exams: Automatic semantic partial credit assignment approach. Springer International Publishing, Cham, p. 110–123, 2016. 46

JOSHI, Ankur; KALE, Saket; CHANDEL, Satish; PAL, D Kumar. Likert scale: Explored and explained. **British journal of applied science & technology**, Sciencedomain International, v. 7, n. 4, p. 396, 2015. 47

JOVENSGENIOS. **Qual a diferença entre Avaliação Diagnóstica, Formativa e Somativa?** 2021. Disponível em: <https://blog.jovensgenios.com/avaliacao-diagnostica-formativa-e-somativa/>. Acesso em: 16 jan 2023. 20

KUROSE, J.F.; ROSS, K.W. **Computer Networking: A Top-Down Approach**. Pearson Education, 2012. ISBN 9780133128093. Disponível em: <https://books.google.com.br/books?id=GXIvAAAAQBAJ>. 20, 21

LEFF, Avraham; RAYFIELD, James T. Web-application development using the model/view/controller design pattern. *In: IEEE. Proceedings fifth ieee international enterprise distributed object computing conference*. [S.l.], 2001. p. 118–127. 24

LLAMAS-NISTAL, Martín; FERNÁNDEZ-IGLESIAS, Manuel J.; GONZÁLEZ-TATO, Juan; MIKIC-FONTE, Fernando A. Blended e-assessment: Migrating classical exams to the digital world. **Computers & Education**, v. 62, p. 72–87, 2013. ISSN 0360-1315. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0360131512002497>. 17

MACHADO, Eusébio. A "fabricação" das notas escolares: equidade, sucesso e avaliação. 01 2019. 20

MARKREADER. Pilot Software, Ltd, 2022. Disponível em: <https://www.pilotltd.com/en/products/markreader>. 16

MOODLE. Moodle, 2022. Disponível em: <https://moodle.org/>. Acesso em: 02 jul. 2023. 16

PAREDES, Yancy Vance; HSIAO, I-Han. Webpga: An educational technology that supports learning by reviewing paper-based programming assessments. **Information**, MDPI, v. 12, n. 11, p. 450, 2021. 17

PYTHON. Python Software Foundation, 2023. Disponível em: <https://www.python.org/>. Acesso em: 02 jul. 2023. 23, 28

REACT. Meta Platforms, Inc, 2022. Disponível em: <https://reactjs.org/>. Acesso em: 02 jul. 2023. 24, 28

RUBIO, D. **Beginning Django: Web Application Development and Deployment with Python**. Apress, 2017. ISBN 9781484227879. Disponível em: <https://books.google.com.br/books?id=8sU7DwAAQBAJ>. 23

SAIHOOD, Noor; ELAMEER, Amer. Design and implementation of standard personal website for iraqi academics. 11 2016. 23

SANTILLANA, Fundação. Brasil no pisa 2015: análises e reflexões sobre o desempenho dos estudantes brasileiros. 2016. Disponível em: https://download.inep.gov.br/acoes_internacionais/pisa/resultados/2015/pisa2015_completo_final_baixa.pdf. 16

SHKLAR, Leon; ROSEN, Rich. **Web Application Architecture: Principles, Protocols and Practices**. 2. ed. Chichester, UK: Wiley, 2009. ISBN 978-0-470-51860-1. Disponível em: <https://www.safaribooksonline.com/library/view/web-application-architecture/9780470518601/>. 22

SILVA, Danilo Scherre Garcia da; MATOS, Poliana Michetti de S; ALMEIDA, Daniel Manzoni de. Métodos avaliativos no processo de ensino e aprendizagem: uma revisão. **Cadernos de Educação**, n. 47, p. 73–84, 2014. 19

VIGNONI, David; CALIMO; MGYUGCHA. **Client-Server Model**. 2011. Disponível em: <https://upload.wikimedia.org/wikipedia/commons/c/c9/Client-server-model.svg>. Acesso em: 02 jul. 2023. 21

Apêndice A – Exemplo de entrega de atividade

Matrícula: 20230006

Atividade - Redes de Computadores

Folha de Resposta

Questão 1 - Marque a resposta correta.

a)

☒ b)

c)

d)

e)

Questão 2 - Escreva a resposta.

A confidencialidade é uma medida de segurança que visa proteger informações sensíveis contra divulgação não autorizada a terceiros.

Questão 3 - Marque as afirmativas como verdadeiro (V) ou falso (F)

I - (F)

II - (F)

III - (V)

IV - (F)

Apêndice B – Exemplo de folha de questões de uma atividade

Atividade - Redes de Computadores

Questões

1 - O protocolo de controle de transmissão, ou TCP (Transmission Control Protocol), foi projetado especificamente para oferecer um fluxo de bytes fim a fim confiável em uma rede interligada não confiável. O TCP foi projetado para se adaptar dinamicamente às propriedades da rede interligada e ser robusto diante dos muitos tipos de falhas que podem ocorrer.

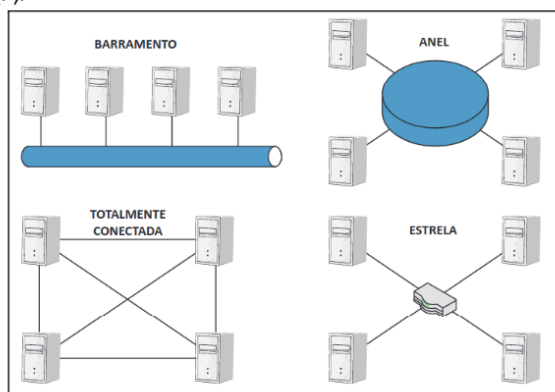
TANENBAUM, A. S.; WETHERALL, D. Redes de computadores, 5. ed. São Paulo: Pearson, 2011 (adaptado).

A respeito das características do protocolo TCP, assinale a opção correta.

- a) Inicia e encerra conexões entre duas entidades por meio do handshake de três vias.
- b) Possui controle de congestionamento, evitando assim a ocorrência de tráfego excessivo na rede.
- c) Permite a comunicação entre entidades por meio de um serviço não orientado a conexões.
- d) Implementa o mecanismo de janela deslizante, permitindo o compartilhamento de uma mesma conexão.
- e) Adapta o tamanho do segmento TCP ao tamanho do MTU da camada de enlace, evitando assim a sua fragmentação ao longo do caminho.

2 - Defina o conceito de confidencialidade que os agentes de tratamento devem considerar nas medidas de segurança para propor e implementar políticas de segurança para redes de computadores.

3 - Considerando as topologias na imagem abaixo, avalie as afirmativas como Verdadeiro (V) ou Falso (F).



I - Em uma rede Totalmente Conectada são utilizados conexões ponto-multiponto para interligar seus nós, permitindo que todos fiquem interconectados.

II - A topologia em Barramento segmenta o domínio de colisão, criando diferentes segmentos de LAN, enquanto mantém todos os dispositivos finais no mesmo segmento broadcast.

III - A topologia em Anel tem como vantagem o retardo de acesso determinístico, pois uma estação sabe qual a melhor e a pior oportunidade para iniciar a transmissão.

IV - Na topologia em Estrela, toda a informação deve passar obrigatoriamente por uma estação central inteligente, que está conectada a cada estação e distribui o tráfego.

Apêndice C – Exemplo de folha de resposta

Matrícula: _____

Atividade - Redes de Computadores

Folha de Resposta

Questão 1 - Marque a resposta correta.

- a)
- b)
- c)
- d)
- e)

Questão 2 - Escreva a resposta.

Questão 3 - Marque as afirmativas como verdadeiro (V) ou falso (F)

I - ()

II - ()

III - ()

IV - ()

Apêndice D – Exemplo de arquivo de correção

Matrícula: 20230007

Atividade - Redes de Computadores

Folha de Resposta

Questão 1 - Marque a resposta correta.

- a)
- b)
- ☒ c)
- d)
- e)

Questão 2 - Escreva a resposta.

A confidencialidade se refere à obrigação de manter informações pessoais privadas e protegidas contra o acesso e uso não autorizado, sendo um dos principais requisitos para garantir a segurança dos dados.

Questão 3 - Marque as afirmativas como verdadeiro (V) ou falso (F)

- I - (F)
- II - (F)
- III - (V)
- IV - (V)

Atividade: Rede de Computadores

Aluno: 20230007

Nota: 5.0 / 10.0

Questão 1

0.0 / 2.0

Resposta correta: b)

Questão 1 - Marque a resposta correta.

a)

b)

c)

d)

e)

Questão 2

3.0 / 6.0

Faltou citar que o conceito de confidencialidade inclui também proteger os dados da divulgação não autorizada

Questão 2 - Escreva a resposta.

A confidencialidade se refere à obrigação de manter informações pessoais privadas e protegidas contra o acesso e uso não autorizado, sendo um dos principais requisitos para garantir a segurança dos dados.

Questão 3 - Marque as afirmativas...

Questão 3

2.0 / 2.0

Correta.

Questão 3 - Marq

I - $(\mathcal{F})$

II - $(\mathcal{F})$

III - $(\mathcal{V})$

IV - $(\mathcal{V})$