

SISTEMA *WEB* PARA GERAÇÃO AUTOMÁTICA DO TESTE DE PROGRESSO - FASE II

WEB SYSTEM FOR AUTOMATIC GENERATION OF PROGRESS TEST - PHASE II

Eugênio Silva; João Victor da Silva Machado; Eric Falcão Rocha

RESUMO

Desde a sua implantação no UNIFESO, o Teste de Progresso tem se caracterizado como um importante instrumento de auto avaliação institucional. Por meio dele, tanto professores quanto alunos têm tido a oportunidade de identificar virtudes e fragilidades do processo de ensino-aprendizagem e, em caso de necessidade, adotar as devidas medidas corretivas. Atualmente, o UNIFESO dispõe de um sistema que tem sido empregado tanto na elaboração de avaliações das avaliações das disciplinas dos cursos de graduação da instituição, quanto na elaboração do Teste de Progresso. No entanto, há alguns requisitos não atendidos pela solução em uso que justificam a proposta de um novo sistema. Assim, o objetivo do trabalho é o desenvolvimento de um sistema *web* para automatizar o processo de geração da prova do Teste de Progresso e, com isso, facilitar e aumentar a confiabilidade do trabalho dos atores envolvidos no processo de elaboração e geração da prova. A primeira etapa deste trabalho, denominada Fase I, se encarregou do desenvolvimento das funcionalidades de *cadastro de questões e validação por pares e geração de gráficos* do sistema. A Fase II, apresentada aqui, se encarregou do desenvolvimento e da modelagem, respectivamente, das funcionalidades de *administração e de parametrização e geração de prova*.

Palavras-chave: Teste de Progresso; sistema *web*; trabalho colaborativo.

ABSTRACT

Since its implementation at UNIFESO, the Progress Test has been characterized as an important institutional self-assessment instrument. Through it, both teachers and students have had the opportunity to identify strengths and weaknesses in the teaching-learning process and, if necessary, adopt appropriate corrective measures. Currently, UNIFESO has a system that has been used both in the preparation of tests to the institution's undergraduate courses, and in the preparation of the Progress Test. However, there are some requirements not met by the solution in use that justify the proposal of a new system. Thus, the objective of the work is the development of a web system to automate the process of generating the Progress Test and, therefore, facilitate and increase the reliability of the work of the actors involved in the process of preparing and generating the test. The first stage of this work, called Phase I, was responsible for developing the question registration and peer validation and graphics generation functionalities of the system. Phase II, presented here, was responsible for the development and modeling, respectively, of administration and parameterization and test generation functionalities.

Keywords: Progress Test; web system; collaborative work.

INTRODUÇÃO

De acordo com Miranda e Crisostomo (2014):

A autoavaliação numa Instituição de Ensino Superior (IES) consiste numa oportunidade importante de realizar, por meio de pesquisa acurada, um levantamento diagnóstico e a consequente reflexão sobre a realidade de um contexto acadêmico em vistas a ir além do autoconhecimento e, assim, promover a superação das fragilidades e a otimização das potencialidades identificadas.

Ciente da importância da autoavaliação institucional como um instrumento para se buscar a excelência acadêmica, a gestão do UNIFESO iniciou em 2008 a implantação do Programa de Autoavaliação Institucional (PAAI). Trata-se de um conjunto de mecanismos de autoavaliação que compreende oito projetos, voltados aos três segmentos da comunidade acadêmica: docentes, estudantes e técnicos-administrativos (MIRANDA; MORAES, 2014). Um desses projetos é o Teste de Progresso, uma avaliação anual aplicada a todos os cursos de graduação da instituição para medir o desenvolvimento cognitivo dos alunos.

Conforme Miranda e Moraes (2014), o Teste de Progresso:

é uma avaliação formativa cujo conteúdo tem por objetivo avaliar o crescimento cognitivo do estudante, aplicado longitudinalmente em todos os períodos, sem caráter de premiação, punição ou promoção, traduzindo na prática a política de avaliação formativa preconizada no Projeto Político-Pedagógico (PPPI) do UNIFESO.

O Teste de Progresso começou a ser implantado em alguns cursos em 2007 e a partir de 2009 passou a ser aplicado a todos os cursos do UNIFESO (MORGADO, 2014). A partir daí tem produzido informações relevantes que contribuem para o aprimoramento do processo de ensino-aprendizagem, tanto por parte da gestão acadêmica quanto do corpo docente. Paralelamente, os estudantes também têm tido a oportunidade de acompanhar quantitativamente o seu desempenho acadêmico (MIRANDA; MORAES, 2014).

OBJETIVOS

O objetivo geral deste trabalho consiste em desenvolver um sistema *web*, que permita a automati-

zação de todo o processo de elaboração da prova do Teste de Progresso dos cursos de graduação do UNIFESO, oferecendo assim um meio simples, amigável e confiável para execução dessa tarefa. Parte do sistema proposto já foi desenvolvida na Fase I do projeto no qual se insere e envolve, especificamente, as funcionalidades de *cadastro e validação de questões e geração de gráficos*. Essas funcionalidades estão descritas em detalhes em Silva, Geonizeli, Fonte, Mello Júnior (2021). Neste trabalho, denominado Fase II, as atenções estiveram voltadas para o desenvolvimento das funcionalidades de *administração e de parametrização e geração de prova*.

JUSTIFICATIVA

Por muito tempo, o processo de construção da prova do Teste de Progresso foi executado de forma totalmente manual. Esse processo compreendia a elaboração das questões, bem como a definição das características da prova, como distribuição das questões entre as disciplinas de um curso e o grau de dificuldade, e a montagem do arquivo (formato .DOCX) com as questões, considerando que esse arquivo devia seguir um formato estabelecido previamente. Em linhas gerais, conforme descrito em Silva, Geonizeli, Fonte, Mello Júnior (2021), esse processo era guiado por passos altamente propensos a falhas que, em casos extremos, podia provocar a anulação de questões e, com isso, comprometer o processo de avaliação.

Atualmente, o UNIFESO dispõe do sistema Qstione¹, que foi adquirido com o objetivo inicial de elaboração de bancos de questões objetivas e discursivas e também para a criação de avaliações para as disciplinas de todos os cursos de graduação da instituição. Posteriormente, o seu uso foi estendido para a geração do Teste de Progresso. Trata-se de um sistema semelhante ao proposto aqui, mas que apresenta alguns diferenciais importantes em relação ao Qstione, que justificam os esforços empenhados em seu desenvolvimento.

A *validação por pares* é um diferencial de destaque uma vez que, diferentemente do Qstione, propõe um processo de validação de questões que seja bidirecional. O Qstione adota um processo unidirecional,

¹ <https://www.qstione.com.br/novo/>

pois, não existe a possibilidade de o elaborador da questão contestar um parecer emitido pelo avaliador. O sistema proposto prevê a possibilidade de sucessivas interações entre o revisor e o elaborador até que se chegue a um consenso e a questão possa ser aprovada, caracterizando assim um processo bidirecional. Outro diferencial importante do sistema proposto em relação ao Qstione, diz respeito à escolha do revisor. No Qstione, a escolha do revisor não está sob o controle do elaborador e, com isso, pode ser designado um professor que não tenha conhecimento sobre o tema da questão sob sua avaliação. Conseqüentemente, a avaliação efetuada por esse professor fica limitada a aspectos relacionados à forma da questão, perdendo-se a oportunidade de uma avaliação mais profunda, que considere também aspectos relacionados ao conteúdo. No sistema proposto, como parte do processo de elaboração de uma questão, indica-se obrigatoriamente o revisor. Assim, há a possibilidade de selecionar um professor que esteja capacitado a avaliar a questão sob todos os aspectos.

Ademais, como o sistema proposto está direcionado especificamente à elaboração do Teste de Progresso, apresenta uma interface mais amigável que a do Qstione. Isso exige um esforço de aprendizagem menor de seus usuários, em especial daqueles com pouca familiaridade com tecnologia.

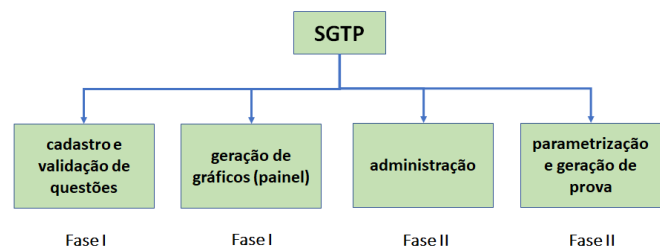
METODOLOGIA

A Internet e as modernas ferramentas de desenvolvimento de aplicações proporcionaram a evolução da *web* de um ambiente de apresentação de conteúdo para outro capaz de hospedar aplicações interativas com diversas finalidades. Aplicações *web*, desconsiderando eventuais restrições, são de acesso global e estão disponíveis a qualquer momento e em qualquer lugar e, para acessá-las, basta um dispositivo computacional conectado à Internet, como um *desktop*, *laptop*, *tablet* ou *smartphone*. Isso faz do desenvolvimento *web* uma excelente opção para construção de aplicações dedicadas ao trabalho colaborativo, onde têm-se vários atores executando uma tarefa em comum, mas que podem não estar no mesmo ambiente físico (ALVES, 2015; QUEIRÓS, PORTELA, 2018).

O sistema proposto na Fase I deste trabalho se caracteriza bem como um sistema *web*, pois prevê o

trabalho colaborativo de várias pessoas para alcançar um objetivo comum que é a elaboração do Teste de Progresso. As funcionalidades essenciais previstas para o sistema, denominado SGTP (Sistema para a Geração do Teste de Progresso) e as fases em que foram desenvolvidas, são apresentadas na Figura 1:

Figura 1: funcionalidades do SGTP e fases em que foram implementadas



Fonte: elaborada pelos autores

A seguir são apresentadas, em linhas gerais, as descrições de cada funcionalidade:

Cadastro de questões e validação por pares: permite que professores cadastrados no sistema possam elaborar as questões, provendo os elementos tradicionais como enunciado, resposta e distratores, e outros atributos importantes para a parametrização e geração da prova. Dentre esses atributos estão o nível de dificuldade, a modalidade, o assunto, o tipo (no caso de questões objetivas) e a habilidade cognitiva da Taxonomia de Bloom. Durante o cadastro da questão, o professor elaborador indica outro professor para validá-la. Essa é uma medida importante para evitar que questões que apresentem eventuais falhas em sua formulação possam ser incluídas no Teste de Progresso e precisem ser anuladas posteriormente.

Geração de gráficos (painel): permite que o usuário visualize gráficos que, de forma amigável, apresentam a distribuição das questões de sua autoria segundo diversas possibilidades de agrupamento. As questões podem, por exemplo, ser agrupadas de acordo com a combinação de atributos como: grau de dificuldade, tipo de questão (no caso de questões objetivas), habilidade do domínio cognitivo, etc. Com isso, o usuário tem um instrumento que permite verificar com facilidade se, por exemplo, existe um equilíbrio entre as quantidades de questões considerando um determinado atributo. Existe ainda a possibilidade de o usuário

acompanhar os status das questões com as quais está diretamente envolvido, seja na elaboração, seja na revisão. Com isso, é possível verificar facilmente as questões que, por exemplo, estão aguardando a avaliação do revisor, as que ainda estão em processo de elaboração ou aquelas que estão aguardando a avaliação do usuário corrente.

Administração: permite a inclusão e a exclusão de usuários e também a configuração de perfis desses usuários para que sejam estabelecidos os privilégios de visualização das questões cadastradas. Além disso, permite também alterações nas listas de valores dos atributos *assunto*, *eixo de formação* e *categoria*, que são selecionados durante a criação de questões.

Parametrização e geração da prova: permite definir as características da prova, levando em consideração a distribuição de questões de acordo com a configuração de percentuais associados aos diversos atributos associados às questões. Uma vez concluída a parametrização, permite a geração da prova de forma automática, onde um algoritmo de otimização se responsabiliza por selecionar as questões de forma a atender ao máximo a parametrização estabelecida pelo usuário. O sistema permite ainda a geração manual da prova, onde o próprio usuário seleciona as questões que irão compor a prova, sendo essa seleção facilitada por filtros associados aos atributos das questões. Em ambos os casos o arquivo final da prova é gerado automaticamente, contendo todas as questões escolhidas dentro da formatação esperada.

Como apontado na Figura 1, na Fase I as atenções estiveram voltadas para o desenvolvimento das funcionalidades de *cadastro de questões e validação por pares* e de *geração de gráficos (painel)*, cujos detalhes estão descritos em Silva, Geonizeli, Fonte e Mello Júnior (2021). Na Fase II, o desenvolvimento esteve voltado para as funcionalidades de *administração*, descrita em Silva, Geonizeli, Machado e Rufino (2022) e de *parametrização e geração da prova*.

RESULTADOS OBTIDOS

Aprimoramentos Técnicos do Sistema

No decorrer do desenvolvimento das funcionalidades previstas na Fase II, houve a necessidade de

efetuar alguns aprimoramentos na codificação das funcionalidades implementadas na Fase I. Esses aprimoramentos impactaram toda a sua arquitetura e desencadearam modificações tanto no *back-end* quanto no *front-end* da aplicação.

Inicialmente, os projetos de *back-end* e *front-end* da aplicação estavam separados, o que tornava o processo de configuração do projeto mais trabalhoso, tanto em um servidor para consumo, quanto no ambiente de desenvolvimento local para os desenvolvedores. Para simplificar esse processo de configuração, optou-se pela reestruturação do código da aplicação de tal maneira que o *back-end* e o *front-end* fossem reunidos em um mesmo projeto. Através dessa união e da exploração do conceito de “containerização” usando Docker², o *setup* do projeto passou a demandar apenas a execução de um *script*, tornando desnecessárias as instalações do gerenciador de banco de dados e dos interpretadores de linguagens, tanto no servidor quanto na estação de trabalho do usuário. Com essa simplificação, passou a ser possível executar o projeto até mesmo em um *tablet* usando o navegador.

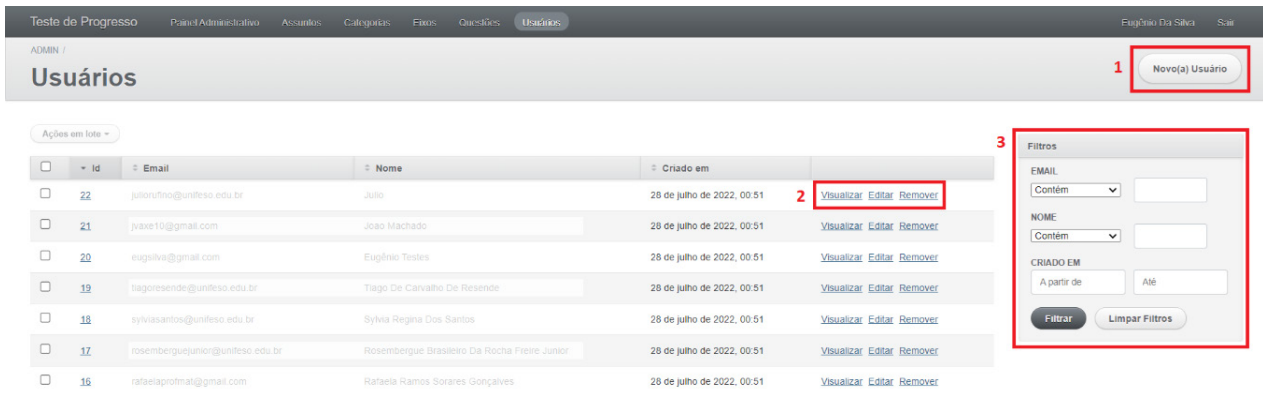
Administração

Antes da implementação da funcionalidade de *administração*, qualquer modificação que envolvesse, por exemplo, a inclusão ou exclusão de um usuário ou a configuração de seu perfil, só podia ser efetuada pelo desenvolvedor por meio alterações no código da aplicação. Obviamente que essa limitação cria uma dependência indesejada em relação ao desenvolvedor e pode comprometer o uso da aplicação, sobretudo em situações em que alterações de usuários são frequentes.

Com a nova funcionalidade, um usuário com perfil de administrador passou a ter acesso a uma interface especial em que configurações que antes dependiam de alterações no código do sistema, passaram a ser feitas de uma forma bastante amigável. A Figura 2 a seguir apresenta a tela com a listagem de usuários em que aparece no destaque 1, um botão que permite a inclusão de um novo usuário. No destaque 2 estão as opções de “Visualizar”, “Editar” e “Remover” um registro de usuário. No destaque 3 estão alguns filtros que facilitam a navegação pelos registros de usuários.

² <https://hub.docker.com/>

Figura 2: tela de listagem de usuários



Fonte: Elaborada pelos autores

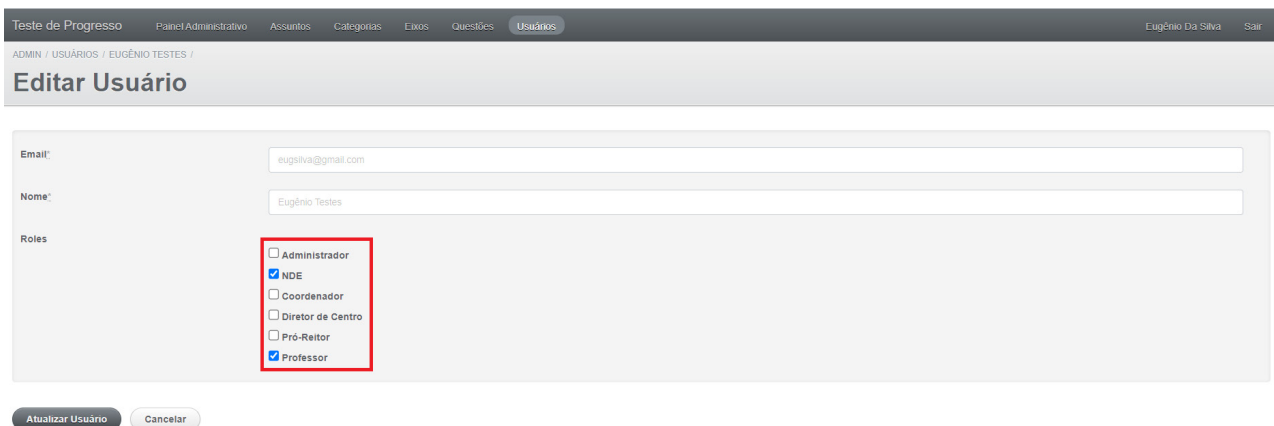
As opções de inclusão e de edição, uma vez acionadas, levam à tela apresentada na Figura 3. Nela, além do nome e do endereço eletrônico do usuário, observa-se no destaque que é possível definir também seu(s) perfil(is).

Esses perfis estabelecem as permissões de visualização de questões cadastradas de acordo com a seguinte descrição:

Administrador: permite visualizar, editar e excluir qualquer tipo de cadastro e também de parametrizar e gerar a prova do Teste de Progresso.

Professor: permite criar questões, editar suas próprias questões que ainda não tenham sido registradas, validar questões e visualizar o painel com estatísticas sobre as questões de sua autoria.

Figura 3: tela de edição de usuário



Fonte: Elaborada pelos autores

NDE: permite editar quaisquer questões, desde que do curso ao qual o usuário está associado e que ainda não tenham sido registradas, validar questões, visualizar o painel com estatísticas sobre todas as questões de todos os temas do curso ao qual o usuário está associado e parametrizar e gerar a prova do Teste de Progresso.

Coordenador: permite visualizar o painel com as estatísticas sobre todas as questões de todos os temas do curso ao qual o usuário está associado e parametrizar e gerar a prova do Teste de Progresso.

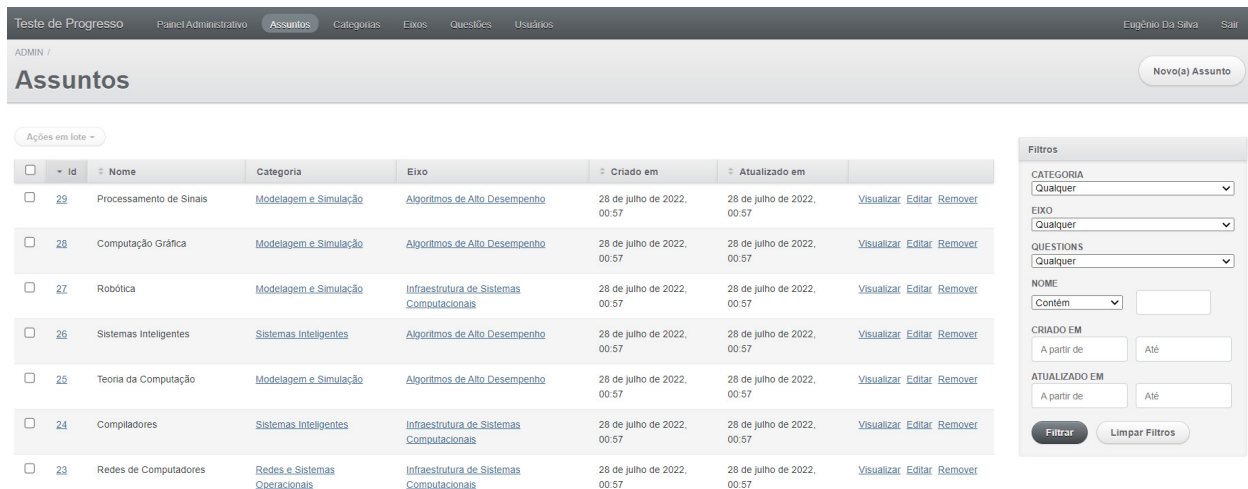
Diretor de Centro: permite visualizar o painel com as estatísticas sobre todas as questões de todos os temas e de todos os cursos do centro ao qual o usuário está associado.

Pró-Reitor: permite visualizar o painel com as estatísticas sobre todas as questões de todos os temas e de todos os cursos da instituição.

Durante a elaboração de uma questão, um dos atributos que deve ser configurado é o assunto ao qual a questão está associada. Uma vez selecionado o assunto, automaticamente são estabelecidas as correspondências entre aquele assunto e o eixo de formação e a categoria. Antes da implementação da funcionalidade de administração, quaisquer modificações nas listagens de opções, tanto de assuntos quanto de eixos de formação e categorias, só podiam ser feitas diretamente no código. Com essa nova funcionalidade, a interface passou a permitir alterações nas listagens de valores desses atributos de forma bem mais amigável. A Figura 4 mostra a tela de listagem de assuntos, que é muito semelhante à tela de listagem de usuários (Figura 2). O mesmo vale para as telas de listagem de eixos de formação e de categorias.

Outra facilidade importante trazida pelo mecanismo de administração está na possibilidade de o administrador alterar o status de uma questão já registrada. A última etapa do processo de criação de uma questão consiste no seu registro. Uma vez registrada, não existe mais a possibilidade de aquela questão ser editada ou excluída. No entanto, em casos excepcionais, pode haver a necessidade de editar ou excluir uma questão já registrada por diferentes razões. Uma delas seria a percepção tardia de algum erro que tenha passado despercebido tanto pelo elaborador quanto pelo revisor da questão.

Figura 4: tela de listagem de assuntos

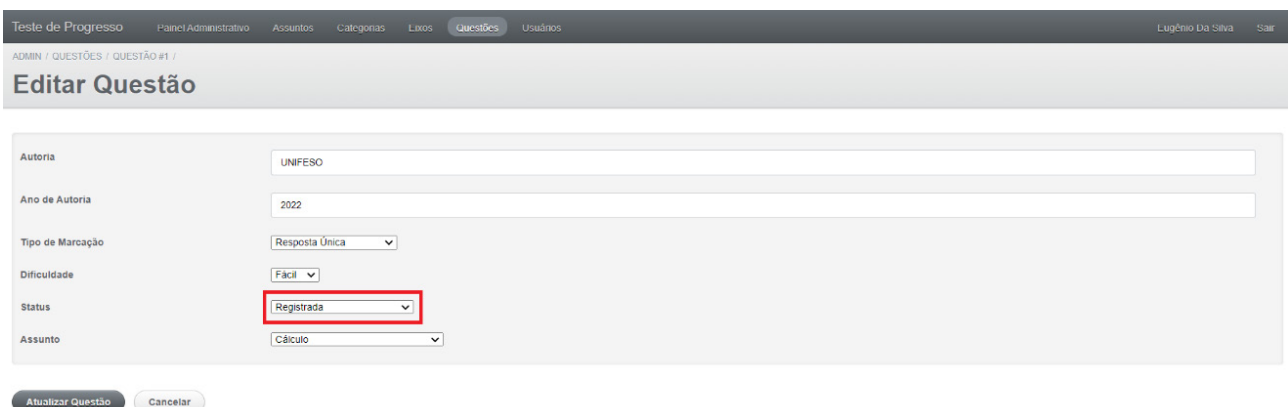


Id	Nome	Categoria	Eixo	Criado em	Atualizado em	Ações
29	Processamento de Sinais	Modelagem e Simulação	Algoritmos de Alto Desempenho	28 de julho de 2022, 00:57	28 de julho de 2022, 00:57	Visualizar Editar Remover
28	Computação Gráfica	Modelagem e Simulação	Algoritmos de Alto Desempenho	28 de julho de 2022, 00:57	28 de julho de 2022, 00:57	Visualizar Editar Remover
27	Robótica	Modelagem e Simulação	Infraestrutura de Sistemas Computacionais	28 de julho de 2022, 00:57	28 de julho de 2022, 00:57	Visualizar Editar Remover
26	Sistemas Inteligentes	Sistemas Inteligentes	Algoritmos de Alto Desempenho	28 de julho de 2022, 00:57	28 de julho de 2022, 00:57	Visualizar Editar Remover
25	Teoria da Computação	Modelagem e Simulação	Algoritmos de Alto Desempenho	28 de julho de 2022, 00:57	28 de julho de 2022, 00:57	Visualizar Editar Remover
24	Compiladores	Sistemas Inteligentes	Infraestrutura de Sistemas Computacionais	28 de julho de 2022, 00:57	28 de julho de 2022, 00:57	Visualizar Editar Remover
23	Redes de Computadores	Redes e Sistemas Operacionais	Infraestrutura de Sistemas Computacionais	28 de julho de 2022, 00:57	28 de julho de 2022, 00:57	Visualizar Editar Remover

Fonte: Elaborada pelos autores

A tela de listagem de questões também segue o mesmo modelo das telas apresentadas nas Figuras 2 e 4. Quando a opção de edição é selecionada, tem-se a tela apresentada na Figura 5. Nela, observa-se a possibilidade de alterar vários atributos da questão, dentre eles o status, que aparece em destaque.

Figura 5: tela de edição de questão



Fonte: Elaborada pelos autores

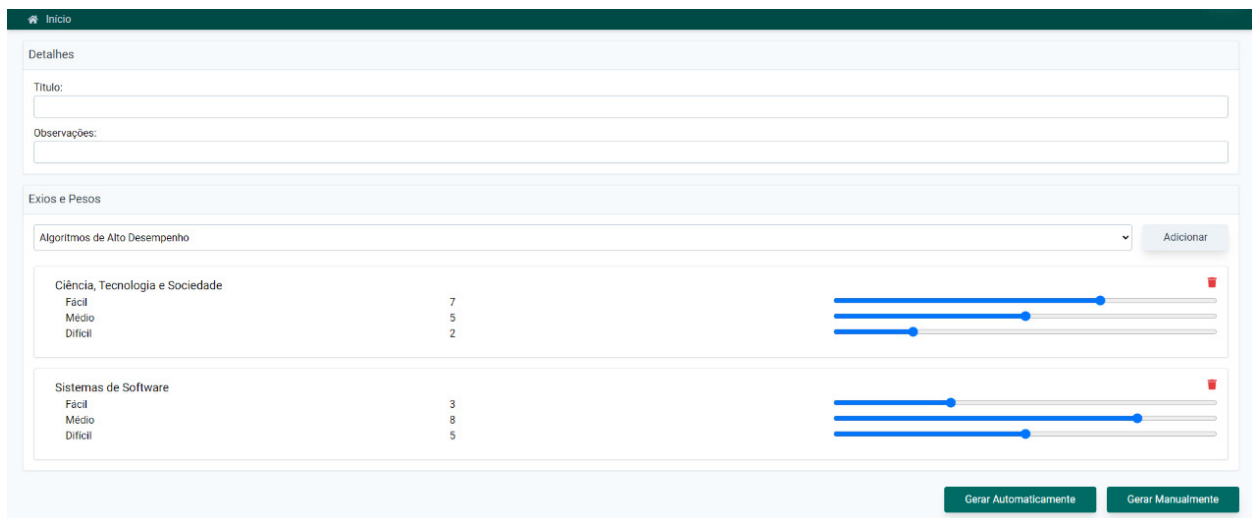
Ao alterar o status de uma questão de “Registrada” para, por exemplo, “Rascunho”, essa questão volta a aparecer para o seu criador como rascunho e, com isso, pode ser livremente editada e novamente submetida ao validador. O papel da funcionalidade de validação por pares é justamente evitar a necessidade desse tipo de intervenção. Contudo, ainda que em casos raros, essa funcionalidade pode ser de grande utilidade.

Parametrização e Geração de Prova

A proposta original para a funcionalidade de *geração da prova* visava a formulação da avaliação de forma totalmente automática, baseada em determinada parametrização. Com o decorrer do desenvolvimento, foi identificada a necessidade de uma opção adicional que permitisse a elaboração da prova de forma manual. Além disso, a possibilidade de editar a avaliação obtida pela geração automática, antes que seja convertida para o arquivo final, também foi identificada como uma funcionalidade importante.

Para atingir os novos objetivos, uma tela destinada à criação de uma nova avaliação foi desenvolvida e está exibida na Figura 6. Nessa tela, o título da prova, bem como observações relacionadas a ela, podem ser inseridos. Em seguida, eixos – áreas do conhecimento – podem ser adicionados à prova e, para cada um deles, pesos ligados às dificuldades das questões podem ser selecionados. Os pesos servem como a base para a geração automática da prova, que irá buscar um equilíbrio entre as questões de determinado nível de dificuldade e de determinada área segundo os valores estabelecidos pelos pesos. Esta etapa constitui a funcionalidade de *parametrização da prova*. Após a seleção dos parâmetros, o usuário pode optar pela geração automática, que fará uso dos eixos e pesos, ou manual da prova.

Figura 6: Tela de criação de prova



Área do Conhecimento	Nível de Dificuldade	Peso
Ciência, Tecnologia e Sociedade	Fácil	7
	Médio	5
	Difícil	2
Sistemas de Software	Fácil	3
	Médio	8
	Difícil	5

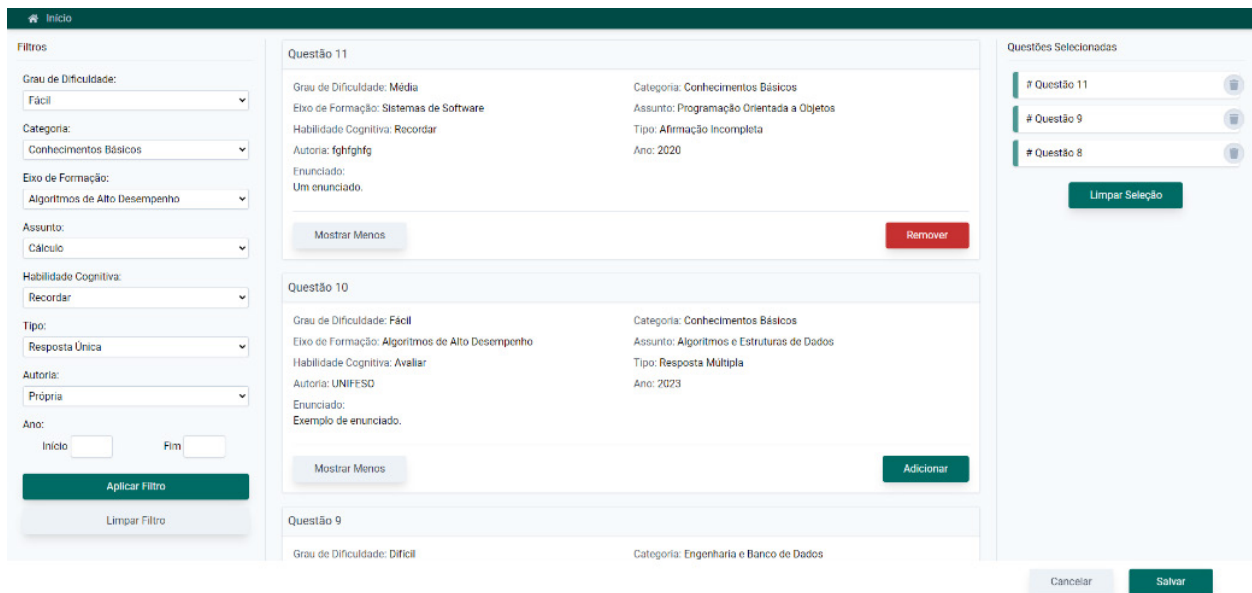
Fonte: Elaborada pelos autores

Ao selecionar a geração manual, o usuário é redirecionado para uma nova página, apresentada na Figura 7, onde as questões podem ser escolhidas individualmente. O painel central mostra as questões, exibindo suas características e enunciado. Para melhorar a visualização, cada questão pode ter seu conteúdo expandido (Figura 7a) ou ocultado (Figura 7b) ao clicar no botão no seu canto inferior esquerdo. O botão localizado na direção oposta permite a adição ou remoção da questão na lista de questões selecionadas, encontrada no painel à direita. O ícone de lixeira à direita dos itens da lista também é responsável pela remoção de determinada questão, enquanto a opção *limpar seleção* realiza a remoção de todas as questões selecionadas. O painel à esquerda, por sua vez, apresenta uma série de filtros que facilitam a localização de questões específicas. Ao concluir a seleção, o usuário poderá salvá-la e seguir para a próxima etapa ou cancelar a operação de geração manual da prova.

No estado no qual a aplicação se encontra, o sistema de filtros ainda está inoperante e as questões carregadas na página de geração manual não são necessariamente aquelas aprovadas durante a etapa de *validação por pares*. A funcionalidade de salvamento e a página referente à próxima etapa da geração manual também precisam de implementação.

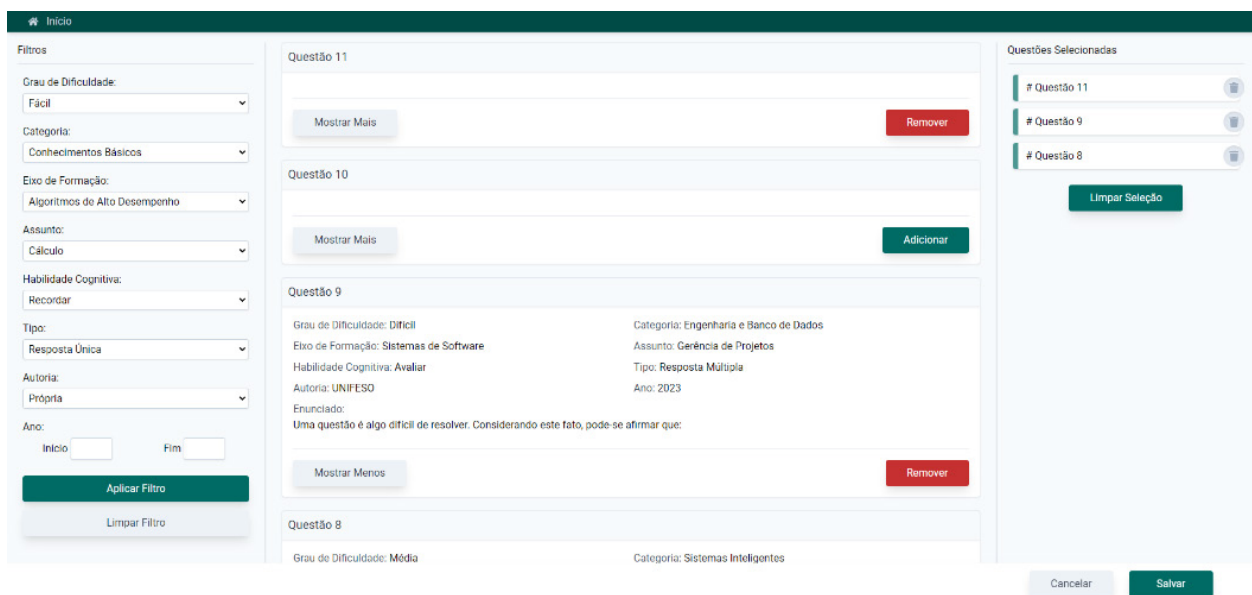
Quanto à geração automática, nenhum progresso foi feito com relação a sua parte visual, ainda que a página de seleção de questões da geração manual esteja prevista para ser utilizada no momento de edição da prova obtida pela geração automática. No entanto, a solução por trás deste mecanismo já foi totalmente modelada.

Figura 7: Tela de geração manual da prova



The screenshot shows a web interface for manual question generation. On the left, there is a 'Filtros' (Filters) panel with dropdown menus for 'Grau de Dificuldade' (Easy), 'Categoria' (Conhecimentos Básicos), 'Eixo de Formação' (Algoritmos de Alto Desempenho), 'Assunto' (Cálculo), 'Habilidade Cognitiva' (Recordar), 'Tipo' (Resposta Única), and 'Autoria' (Própria). Below these are 'Aplicar Filtro' and 'Limpar Filtro' buttons. The main area displays three questions (Questão 11, 10, and 9) with their respective details like difficulty, category, and author. Each question has a 'Mostrar Menos' button and a 'Remover' button. On the right, a 'Questões Selecionadas' panel lists the selected questions (11, 9, 8) with a 'Limpar Seleção' button. At the bottom right, there are 'Cancelar' and 'Salvar' buttons.

(a) Questões expandidas



This screenshot is similar to (a) but shows a different set of questions. The 'Filtros' panel is identical. The main area displays three questions (Questão 11, 10, and 9) with their details. Questão 11 has a 'Mostrar Mais' button and a 'Remover' button. Questão 10 has a 'Mostrar Mais' button and an 'Adicionar' button. Questão 9 has a 'Mostrar Menos' button and a 'Remover' button. The 'Questões Selecionadas' panel on the right lists the selected questions (11, 9, 8) with a 'Limpar Seleção' button. 'Cancelar' and 'Salvar' buttons are at the bottom right.

(b) Questões ocultadas

Fonte: Elaboradas pelos autores

Considerando que a geração automática da prova do Teste de Progresso pode ser interpretada como um problema de otimização, que envolve a busca pela solução mais adequada dentre um universo de possíveis soluções, uma técnica muito utilizada para a solução de tais problemas foi escolhida para servir como a base para a implementação dessa funcionalidade: os algoritmos genéticos, originalmente propostos em Holland (1975). Essa técnica, inspirada no *Princípio da Seleção Natural*, proposto por Charles Darwin em sua Teoria da Evolução das Espécies, baseia-se em conjuntos de possíveis soluções, denominadas *populações*, onde cada solução, chamada *indivíduo*, passa por um processo de evolução, no qual suas propriedades são modificadas por meio de mutações ou trocadas por propriedades de outras soluções por meio de cruzamentos. Ao final do processo de otimização, espera-se encontrar uma população com os indivíduos mais aptos, ou seja, as melhores soluções para um determinado problema de otimização. Para que tal objetivo possa ser alcançado, é essencial que o problema seja modelado da melhor forma possível, evitando ou excluindo completamente soluções inválidas.

Nos algoritmos genéticos, cada indivíduo é representado por uma estrutura denominada *cromossomo*, podendo ser um número binário ou composto por um conjunto de números inteiros, por exemplo. Para o problema de otimização por trás da geração da prova do Teste de Progresso, a representação a ser adotada consiste em um vetor com n números inteiros, onde n representa a quantidade de questões a serem inseridas na prova e cada número inteiro representa o identificador de alguma questão que esteja no banco de questões. Assim, o indivíduo representado pelo cromossomo consiste em uma possível configuração de questões para a prova. Um exemplo de cromossomo é apresentado na Figura 8:

Figura 8: Exemplo de cromossomo que representa uma prova

45	32	17	28	95	74	23	15	94	67
----	----	----	----	----	----	----	----	----	----

Fonte: Elaborada pelos autores

No exemplo apresentado tem-se um vetor de 10 posições, representando uma prova com 10 questões. Em cada posição do vetor tem-se um valor correspondente ao identificador de uma questão que esteja armazenada no banco de questões do sistema.

Para a representação cromossômica estabelecida, a operação de cruzamento encontrada no modelo clássico do algoritmo genético pode ser utilizada sem qualquer alteração. No caso do operador de mutação, contudo, é necessário um pequeno ajuste na implementação. Enquanto na representação binária a mutação consiste em inverter alguns valores do cromossomo (trocar 0 por 1 ou trocar 1 por 0), na representação adotada para a geração da prova, a mutação consiste em substituir alguns valores do cromossomo por outros correspondentes a outras questões disponíveis no banco de questões do sistema.

Apesar de os operadores descritos serem considerados adequados à nova forma de representação cromossômica, não se pode garantir que os indivíduos gerados após a aplicação desses operadores sejam válidos. Isso porque as modificações empreendidas podem gerar cromossomos em que o mesmo código de questão apareça mais de uma vez no mesmo cromossomo. Na prática, isso significa ter uma prova com questões repetidas, o que não é permitido. Na falta de operadores mais eficientes, que evitem a geração de indivíduos inválidos, a alternativa a ser adotada consiste em penalizar os eventuais indivíduos inválidos gerados durante a evolução. Mais informações sobre aprimoramentos, formas de representação e operadores genéticos podem ser obtidas em Michalewicz (1996) e Linden (2008).

Por fim, a função de aptidão, responsável por medir o quão aptos são os indivíduos, deve ser definida. Considerando o presente problema de otimização como um problema de maximização, os indivíduos mais aptos são aqueles que assumem os maiores valores de aptidão. Para a construção da função são levados em consideração todos os parâmetros que compõem as características da prova a ser gerada. Esses parâmetros envolvem: intervalo para o ano de criação da questão; o fato de a questão ter sido usada ou não em outras provas; o tipo de questão objetiva; o grau de dificuldade; a habilidade cognitiva a ser avaliada; e o eixo de formação ao qual a questão pertence. Para a geração da prova, o usuário deve ter a liberdade de selecionar um ou mais desses parâmetros para compor a função de aptidão. Dessa forma, a função de aptidão deve ser uma expressão que retorne valores mais altos para configurações de prova que respeitem ao máximo os valores estabelecidos para os parâmetros

escolhidos para compor a função de avaliação. Vale destacar que também deve ser incorporada à função de avaliação um fator que represente a penalização a ser aplicada a indivíduos inválidos.

A tarefa de elaboração da função de aptidão, levando em conta todas as características apresentadas, ainda não foi concluída. Com isso, não foi possível proceder com a implementação da funcionalidade de geração automática da prova. A geração do arquivo de prova é outra funcionalidade que ainda não foi possível implementar.

CONSIDERAÇÕES FINAIS

Por muito tempo o processo de elaboração da prova do Teste de Progresso foi executado de forma totalmente manual, tornando o processo penoso e sujeito a diversos tipos de falhas. As mais comuns envolvem a formatação de texto, gráficos e tabelas, e qualidade gráfica de figuras, ou ainda falhas mais graves relacionadas a erros de formulação. Todas elas devem ser evitadas ao máximo, uma vez que, em geral, resultam em anulação de questões, o que traz prejuízos ao processo avaliativo. A aquisição, por parte do UNIFESO, do sistema Qstione, minimizou alguns desses riscos, mas ainda assim, considerou-se a existência de lacunas que poderiam ser preenchidas pelo sistema proposto na Fase I deste trabalho. Como exemplo, destacam-se a validação de questões por pares com interação bidirecional e a geração automática de prova.

Na Fase I, foram contempladas as funcionalidades de cadastro de questões e validação por pares e geração de gráficos (painel), enquanto a Fase II propôs a implementação das funcionalidades restantes, a saber: administração, parametrização e geração da prova. A administração foi totalmente concluída e descrita em detalhes aqui e em Geonizeli, Machado e Rufino (2022). Quanto à parametrização e geração da prova, a elaboração da tela de criação de avaliação representou um avanço considerável para o projeto, permitindo a implementação da parametrização e da tela de geração manual da prova, ainda que o algoritmo por trás da geração do arquivo ainda não tenha sido desenvolvido. Além disso, foram executados os primeiros passos para a geração automática da prova, em que foi dada uma atenção especial à modelagem do problema para que, futuramente, possa ser resol-

vido por meio de um algoritmo genético. Além da conclusão da implementação, o sistema ainda precisa ser submetido a testes e, se necessário, correções e ajustes para que possa ser validado e, finalmente, disponibilizado para uso aos professores da instituição.

REFERÊNCIAS

- Alves, WP. *Projetos de Sistemas Web – conceitos, estruturas, criação de banco de dados e ferramentas de desenvolvimento*. ed. 1. São Paulo: Érica; 2015.
- Holland, JH, *Adaptation in Natural and Artificial Systems*. Ann Arbor: University of Michigan, 1975.
- Linden, R, *Algoritmos Genéticos: Uma Importante Ferramenta de Inteligência Computacional*, 2ª ed., Rio de Janeiro: Brasport, 2008.
- Michalewicz, Z, *Genetic Algorithm + Data Structures = Evolution Programs*. Germany: Springer-Verlag, 1996.
- Miranda, JFA, Crisostomo, RPG. *Autoavaliação Institucional no UNIFESO. Autoavaliação Institucional no UNIFESO – 15 Anos de Avaliação Transformadora*. Vol. 1. Teresópolis: UNIFESO, 2014.
- Miranda, JFA, Moraes, MBVB. *PAAI: Programa de Autoavaliação Institucional. Autoavaliação Institucional no UNIFESO – 15 Anos de Avaliação Transformadora*. Vol. 1. Teresópolis: UNIFESO; 2014.
- Morgado. FEF. *O Teste de Progresso. Autoavaliação Institucional no UNIFESO – 15 Anos de Avaliação Transformadora*. Vol. 1. Teresópolis: UNIFESO; 2014.
- Queirós, R, Portela, F. *Introdução ao Desenvolvimento Moderno para a Web: do front-end ao back-end, uma visão global!* ed. 1. Lisboa: FCA; 2018.
- Silva, E, Geonizeli, JVS, Fonte, RC, Mello Júnior, LCR. *Sistema Web para Geração Automática do Teste de Progresso. Anais do VI Congresso Acadêmico Científico do UNIFESO – CONFESO*, Teresópolis: Editora UNIFESO, 2021.