

SISTEMA *WEB* PARA GERAÇÃO AUTOMÁTICA DO TESTE DE PROGRESSO – FASE III

Eugênio Silva - eugeniosilva@unifeso.edu.br, docente, Ciência da Computação, UNIFESO.

Gabriel Resende Machado - gabrielmachado@unifeso.edu.br, docente, Ciência da Computação, UNIFESO.

Kauã Landi Fernando, discente, Ciência da Computação, UNIFESO.

Rolyson da Rocha Carreiro, discente, Ciência da Computação, UNIFESO.

PICPq 2024-2025.

Área temática: Metodologias e Abordagens de Ensino Aplicadas à Área de Ciência e Tecnologia

RESUMO

Na etapa original deste trabalho, denominada Fase I, foi proposta a implementação de um sistema *web* que proporcionasse uma forma mais amigável, confiável e prática de elaborar e gerar a prova do Teste de Progresso. De todas as funcionalidades previstas para o sistema, algumas foram implementadas na Fase I e outras transferidas para a Fase II. De maneira geral, nessas duas fases foram desenvolvidas as funcionalidades relacionadas aos módulos de cadastro e validação de questões, geração de gráficos de distribuição de questões e de administração. Quanto ao módulo de parametrização e geração da prova, foi possível avançar na modelagem do problema de otimização por trás da geração automática da prova, sendo a implementação transferida para a Fase III. Nesta Fase III, antes do início da implementação das funcionalidades remanescentes, percebeu-se a necessidade de algumas atualizações na arquitetura construída e tecnologias utilizadas até o momento. Dessa forma, este trabalho apresenta as alterações efetuadas nos elementos visuais e também no conjunto de ferramentas de desenvolvimento, a fim de obter um sistema com um visual mais moderno e, principalmente, mais robusto e alinhado às boas práticas de engenharia de software adotadas atualmente.

Palavras-chave: Teste de Progresso; sistema *web*; experiência do usuário.

INTRODUÇÃO

Desde o início de sua implantação, iniciada em 2007 e consolidada em 2009, o Teste de Progresso tem se destacado como parte do processo de avaliação institucional que envolve os cursos de graduação do UNIFESO. Esse instrumento traz indicadores importantes que direcionam o trabalho de gestores e docentes na elaboração de estratégias para aprimorar o processo de ensino-aprendizagem. Além disso, o corpo discente também se vale desse recurso para acompanhar o crescimento acadêmico. (MORGADO, 2014), (MIRANDA; MORAES, 2014).

A construção de um sistema *web* para facilitar a elaboração e automatizar o processo de geração da prova do Teste de Progresso foi apresentada inicialmente na edição de 2020-2021 do Plano de Iniciação Científica e Pesquisa (PICPq). Nesse contexto, foi proposta a construção de um ambiente *web* colaborativo que facilitasse o trabalho dos professores da instituição nas etapas de elaboração de questões e na geração da prova. Essa necessidade se revelou a partir da experiência vivenciada no processo tradicionalmente empregado na construção da prova, que se caracterizava pela morosidade e propensão a erros (SILVA, GEONIZELI, FONTE, MELLO JÚNIOR, 2020).

Diante dessa constatação, desde então vem sendo desenvolvido um sistema *web* para contemplar as seguintes funcionalidades essenciais:

1. **Cadastro de questões:** permitir aos professores cadastrados no sistema a elaboração de questões de maneira a considerar não só seus elementos essenciais como, enunciado, suporte, resposta correta e distratores, mas também outros atributos envolvidos na parametrização e na geração da prova. Entre esses atributos estão o assunto, o tipo de questão objetiva, o grau de dificuldade e a habilidade cognitiva da Taxonomia de Bloom avaliada pela questão.

2. **Validação por pares:** como parte do processo de criação de uma questão, permitir que o elaborador indique o professor responsável pela sua revisão. Essa funcionalidade é de grande relevância, uma vez que contribui para minimizar as chances de inclusão de uma questão com algum erro na prova, evitando assim eventuais anulações.

3. **Exibição de status de questões:** apresentar ao usuário a situação de cada uma das questões existentes em seu perfil. Essas situações estão relacionadas às várias etapas do processo de elaboração de uma questão. Uma questão pode, por exemplo, estar em modo rascunho, ou esperando o parecer do revisor ou ainda pendente de alterações.

4. **Geração de gráficos:** exibir um painel com gráficos com a distribuição das questões cadastradas pelo usuário segundo diferentes atributos. Dentre eles o assunto, o tipo de questão, o grau de dificuldade e a habilidade do domínio cognitivo avaliada pela questão.

5. **Administração:** permitir a inclusão, a exclusão e a configuração de perfis de usuários do sistema. Esses perfis estabelecem os privilégios de visualização das questões cadastradas e também as permissões de execução de ações de elaboração de questões e de geração da prova.

6. **Parametrização da prova:** permitir a configuração das características da prova, indicando percentuais associados aos atributos das questões, a fim de direcionar o processo de escolha daquelas que vão compor a prova.

7. **Geração da prova:** permitir a geração manual e automática do arquivo da prova, de forma a respeitar a parametrização estabelecida previamente para a seleção das questões.

Nas Fases I e II do projeto, desenvolvidas, respectivamente, nas edições 2020-2021 e 2022-2023 do PICPq, foi possível concluir as implementações das cinco primeiras funcionalidades listadas (SILVA, GEONIZELI, FONTE, MELLO JÚNIOR, 2020), (SILVA, GEONIZELI, FONTE, MELLO JÚNIOR, 2021) e (SILVA, GEONIZELI, MACHADO, RUFINO, 2022). A modelagem do problema de otimização subjacente à geração automática da prova também foi concluída na Fase II (SILVA, GEONIZELI, CORRÊA, ROCHA, 2023). No

entanto, as implementações da *parametrização* e da *geração* (manual e automática) da prova foram transferidas para a Fase III.

Antes da implementação das funcionalidades restantes, observou-se a necessidade de melhorias e ajustes no código construído até aqui. Assim, neste trabalho, que representa o início da Fase III do projeto, são apresentadas as modificações efetuadas nos elementos de interface do sistema e também na arquitetura e no conjunto de tecnologias utilizadas em seu desenvolvimento.

JUSTIFICATIVA

Atualmente, o UNIFESO dispõe de um sistema comercial, denominado Qstione¹, que apresenta funcionalidades semelhantes às do sistema proposto. O Qstione vem sendo usado para a construção de bancos de questões, para a elaboração das avaliações das disciplinas dos cursos de graduação e, mais recentemente, para a geração das provas do Teste Progresso.

Apesar dessa aquisição, há diferenciais no sistema proposto perante o Qstione que justificam a continuidade do seu desenvolvimento. Um desses diferenciais é a *validação por pares*. Trata-se de um processo bidirecional que permite sucessivas interações entre o elaborador e o revisor da questão, até que a questão seja validada. Além disso, a escolha do revisor é uma atribuição do elaborador da questão, permitindo que seja selecionado um par capaz de avaliar não só aspectos de forma, mas também de conteúdo da questão.

No Qstione, a validação se caracteriza como um processo unidirecional, em que o parecer emitido pelo revisor não pode ser contestado pelo elaborador. Além disso, o elaborador não tem controle sobre a escolha do revisor da questão. Logo, não é possível garantir a aderência dos conhecimentos do revisor ao tema da questão em análise, perdendo assim a oportunidade de uma avaliação mais abrangente da questão.

O sistema proposto se destaca ainda pela forma como a prova pode ser gerada. Pode-se gerar a prova de forma totalmente manual, em que o usuário escolhe livremente as questões que vão compor a prova, ou de forma automática, em que um modelo de otimização baseado na técnica de Algoritmos Genéticos é responsável pela seleção das questões (MICHALEWICZ, 1996) e (LINDEN, 2008). Em ambas as modalidades, manual e automática, a parametrização prévia da prova atua como um guia da escolha das questões a fim de obter uma prova com as características desejadas.

Além dos diferenciais apresentados, há ainda o interesse institucional pelo desenvolvimento de uma solução personalizada, que possa ser facilmente ajustada às suas particularidades. Somado a isso, há o incentivo ao desenvolvimento de ideias inovadoras por parte dos professores e estudantes.

OBJETIVOS

Objetivo geral

O objetivo geral do projeto ao qual este trabalho está associado, consiste em desenvolver um sistema *web* que permita o trabalho colaborativo entre professores do UNIFESO para que a elaboração e a geração da prova do Teste de Progresso possam ser feitas de forma mais prática, simples e confiável.

Objetivos específicos

Os objetivos específicos desta etapa do projeto consistiram em:

¹ <https://www.qstione.com.br/novo/>

- modernizar aspectos da interface gráfica do sistema a fim de aprimorar a experiência do usuário;
- atualizar o modelo arquitetural e as tecnologias utilizadas no desenvolvimento do sistema para atender às melhores práticas de engenharia de software;
- reparar *bugs* existentes em algumas funcionalidades do sistema.

METODOLOGIA

Arquitetura e Tecnologias de Desenvolvimento

Com a aprovação da Fase III do projeto (edição 2024-2025 do PICPq), uma nova equipe de discentes precisou ser formada. Os novos discentes foram escolhidos principalmente pelas suas respectivas experiências profissionais e proficiência nas principais tecnologias *web* utilizadas atualmente no mercado.

As primeiras atividades dos novos integrantes se iniciaram por meio de tarefas voltadas para o reconhecimento e familiarização da solução construída até o final da Fase II do projeto. Durante esse período, a equipe de discentes também avaliou a arquitetura em relação às tecnologias utilizadas no *backend*² e *frontend*³ da aplicação e elaborou um parecer sobre as funcionalidades implementadas até o momento. O parecer foi discutido entre toda a equipe do projeto, e chegou-se à conclusão de que era preciso atualizar as tecnologias da arquitetura existente a partir da abordagem *greenfield development* (GIARDINO *et al*, 2015). Os motivos que levaram a essa decisão foram três: (i) alinhar as práticas de desenvolvimento ao estado da arte, considerando as atualizações ocorridas ao longo das edições deste projeto; (ii) corrigir *bugs* mais rapidamente em funcionalidades relacionadas a requisitos funcionais e não funcionais a partir da abordagem da refatoração completa proporcionada pelo *greenfield development*; (iii) tornar o produto final mais extensível e documentado para futuras atualizações de negócio.

O novo conjunto de ferramentas definido para o desenvolvimento do projeto, conjunto esse popularmente conhecido como *stack*, passou a ser formado pelas seguintes tecnologias:

- **PostgreSQL4:** sistema de gerenciamento de banco de dados relacional (RDBMS) de código aberto. Ele é conhecido por sua robustez, escalabilidade e extensibilidade, oferecendo suporte a diversos tipos de dados e consultas complexas. Possui grande popularidade devido ao contínuo suporte da comunidade e à vasta gama de ferramentas e extensões disponíveis.
- **Django5:** é um *framework* para desenvolvimento *backend* de alto nível, escrito em linguagem Python, que facilita o desenvolvimento rápido e o *design* limpo de aplicações. Ele segue o padrão MVC (*Model-View-Controller*), oferecendo uma estrutura organizada e modular para o desenvolvimento. O Django inclui diversas ferramentas integradas, como administração automática, autenticação e manipulação de URLs, o que permite que desenvolvedores construam aplicações robustas e escaláveis de forma mais eficiente. Sua popularidade se deve à combinação de flexibilidade, segurança e facilidade de uso, além do suporte de uma ampla comunidade.
- **Angular6:** *framework* de desenvolvimento *frontend* de código aberto, mantido pelo Google, que facilita a criação de aplicações *web* dinâmicas de página única (SPAs). Escrito em *TypeScript*, um superconjunto da linguagem *JavaScript*, ele oferece uma arquitetura baseada em componentes, o que promove a reutilização

2 Parte da aplicação *web* responsável por pelo processamento de dados, lógica de negócios, gerenciamento de banco de dados, autenticação e comunicação com o *frontend*.

3 O *frontend* de uma aplicação *web* é a parte que interage diretamente com o usuário. Ele envolve tudo o que o usuário vê e interage na tela, como *layout*, design, botões, texto, imagens e formulários.

4 Site oficial: <https://www.postgresql.org>. Acesso em 18/08/24.

5 Site oficial: <https://www.djangoproject.com>. Acesso em 18/08/24.

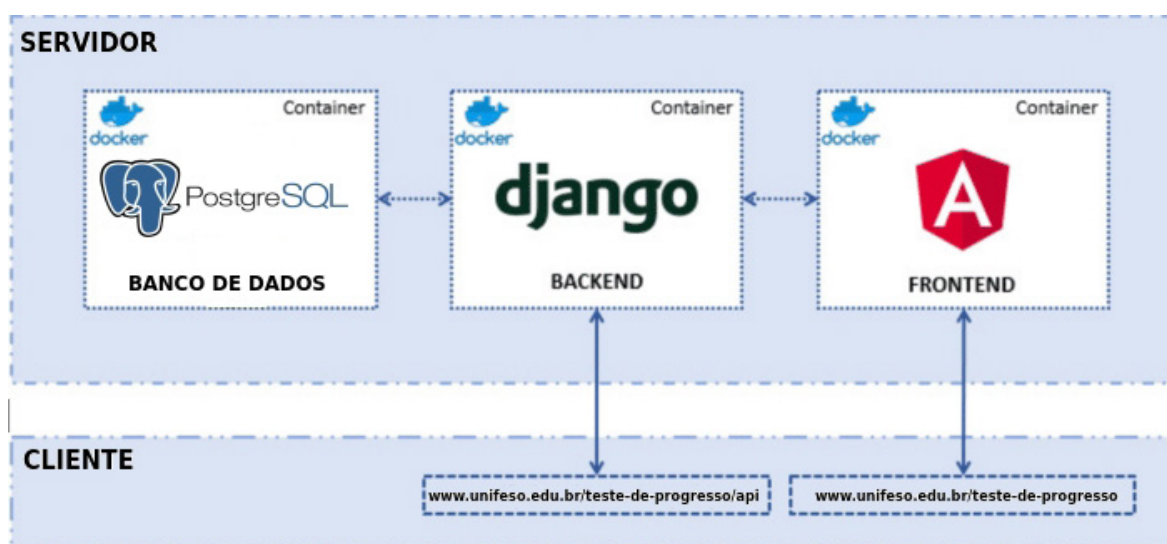
6 Site oficial: <https://angular.dev>. Acesso em 18/08/24.

de código e a modularidade. Angular também inclui recursos avançados, como roteamento, injeção de dependências e testes integrados. É amplamente utilizado devido à sua capacidade de criar interfaces de usuário interativas e escaláveis, além de ser suportado por uma comunidade ativa e extensa.

- **Docker7:** ferramenta de virtualização que permite criar, implantar e executar aplicativos em contêineres. Contêineres são pacotes leves e portáteis que incluem tudo o que um aplicativo precisa para ser executado, como o código, as dependências e o sistema operacional. O Docker garante que um aplicativo funcionará da mesma forma em diferentes ambientes, desde o desenvolvimento até a produção, eliminando problemas de compatibilidade.

A Figura 1 apresenta com mais detalhes o desenho da arquitetura do sistema, instanciada com as tecnologias supracitadas.

Figura 1: Desenho da nova arquitetura do sistema.



Fonte: Adaptado de MOREIRA *et al.* (2021)

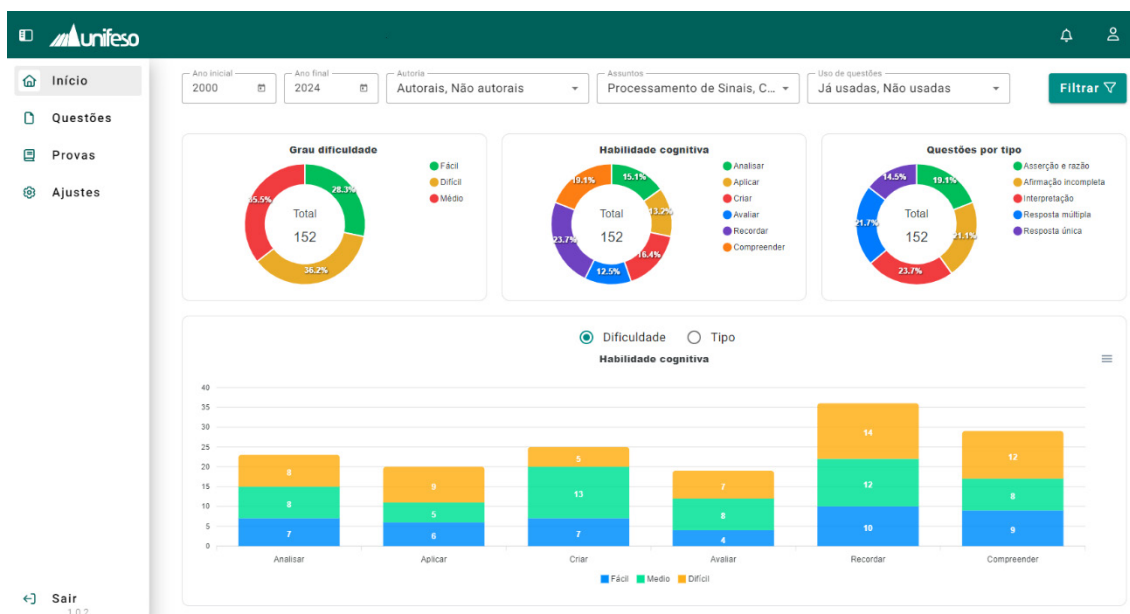
Nova Experiência de Usuário

Após a refatoração do código da aplicação, que envolveu mudanças na arquitetura e nas tecnologias empregadas no desenvolvimento, foram empreendidas modificações na interface. Essas modificações tiveram o propósito de aperfeiçoar a experiência do usuário, oferecendo uma interface mais moderna, simples e prática.

A Figura 2 apresenta o novo *layout* de exibição dos gráficos de distribuição de questões de acordo com os atributos considerados mais relevantes para a análise de diversidade. Os gráficos de rosca exibem as distribuições de questões conforme os atributos *grau de dificuldade*, *habilidade cognitiva* e *tipo*. O gráfico de barras permite a análise da distribuição de questões considerando a combinação do atributo *habilidade cognitiva* com o *grau de dificuldade* ou com o *tipo*. Acima dos gráficos há um conjunto de filtros relacionados a outros atributos das questões que também contribuem para a análise da distribuição. Vale observar ainda que o menu lateral pode ser ocultado para ampliar o espaço a ser ocupado pelos elementos de interface que aparecem à direita.

7 Site oficial: <https://www.docker.com>. Acesso em 18/08/24.

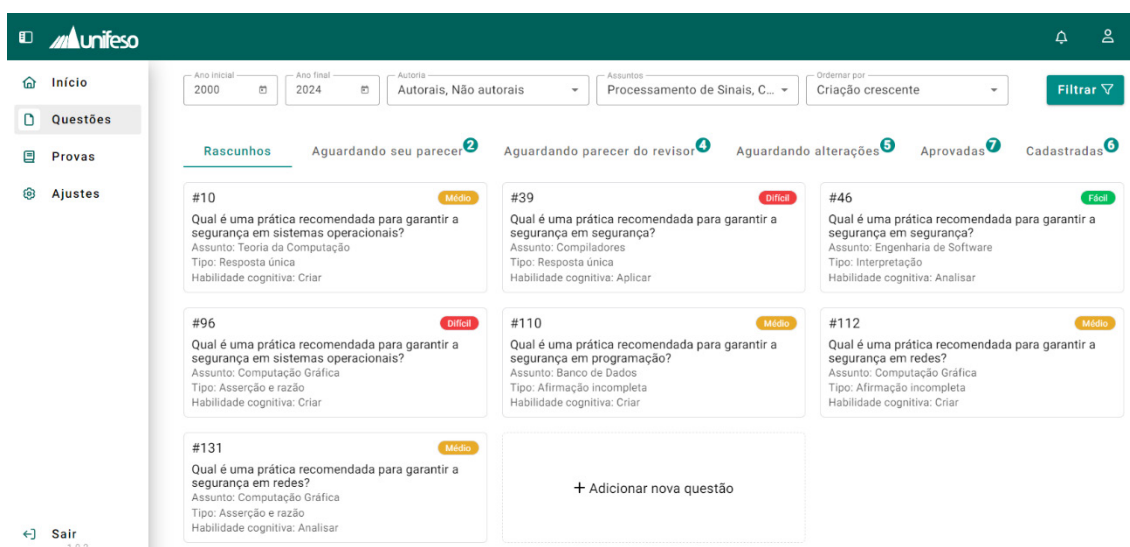
Figura 2: tela de exibição de gráficos de distribuição de questões



Fonte: elaborada pelos autores

A Figura 3 exibe as questões distribuídas em abas, em que cada uma corresponde a um dos *status* possíveis que uma questão pode assumir. Em cada aba há um controle de notificação que, de forma visualmente simples, indica ao usuário se há a necessidade de executar alguma ação relacionada àquelas questões. Na listagem das questões, para que seja possível identificar de forma rápida e fácil as suas principais características, são exibidos os valores associados a alguns dos seus principais atributos (*grau de dificuldade*, *assunto*, *tipo* e *habilidade cognitiva*) e também o trecho inicial do texto do enunciado. Os mesmos filtros presentes na tela de gráficos também estão presentes na tela de questões, permitindo restringir a exibição de questões em todas as abas.

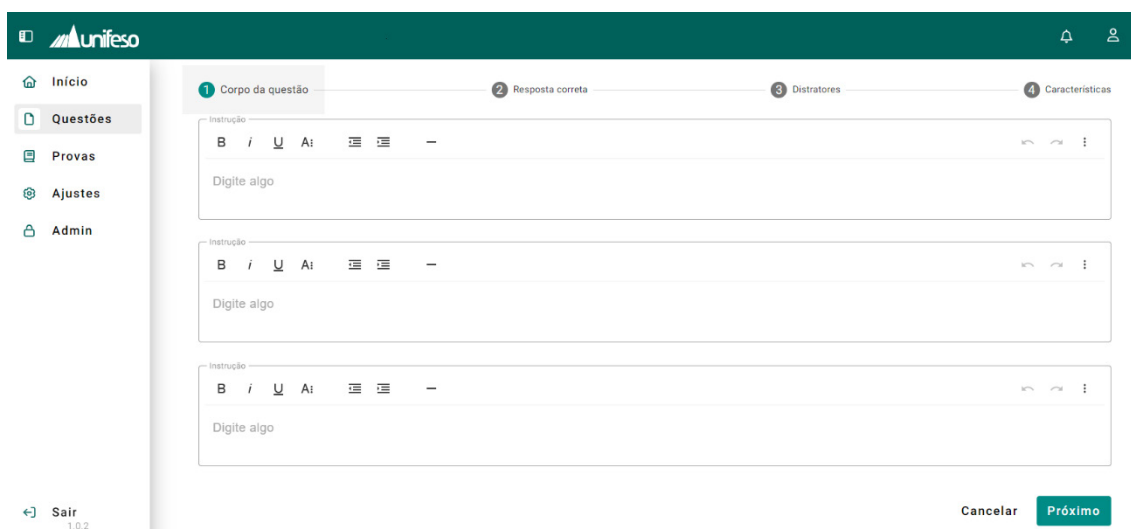
Figura 3: tela de exibição de questões por *status*



Fonte: elaborada pelos autores

Na Figura 4 é apresentada a primeira das quatro telas que compõem os passos de criação de uma questão objetiva. No primeiro passo são preenchidos os elementos correspondentes a *instrução*, *suporte* e *enunciado*. No passo 2 são informadas a *resposta correta*, a *explicação* e a(s) *referência(s)* que ampara(m) a explicação, e no passo 3 são informados os *distratores*.

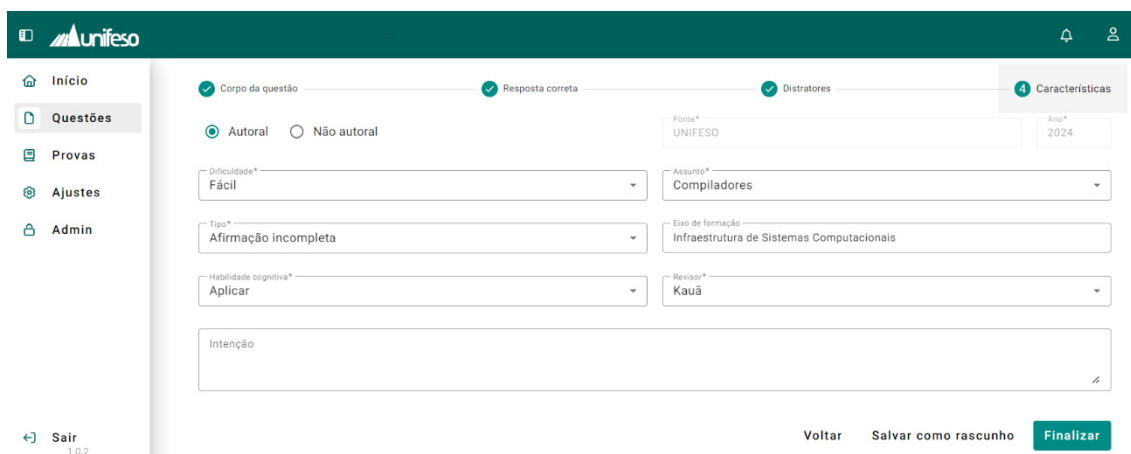
Figura 4: tela do passo 1 de criação de questões



Fonte: elaborada pelos autores

No passo 4 são informados os atributos associados à questão que são úteis não apenas como filtros nas telas anteriores, mas também como parâmetros que vão guiar o processo de geração da prova, seja ele manual ou automático. Na parametrização da prova, o usuário deve selecionar os atributos a serem considerados na seleção de questões e, para cada atributo selecionado, o percentual de questões com aquela característica deve ser indicado. Os atributos informados nessa tela são a *autoria* da questão, o *grau de dificuldade*, o *tipo*, a *habilidade cognitiva*, o *assunto*, o *revisor* e a *intenção* da questão. O *eixo de formação* é preenchido automaticamente quando o *assunto* é selecionado. A tela correspondente a esse passo é exibida na Figura 5.

Figura 5: tela do passo 4 de criação de questões



Fonte: elaborada pelos autores

As telas correspondentes aos itens de menu *Provas* e *Ajustes* ainda estão em fase de construção. A tela correspondente ao item *Provas* deve permitir a parametrização da prova e, no caso da geração manual, oferecer uma forma simples e prática de selecionar as questões que vão compor a prova, especialmente em situações em que o banco de questões é muito numeroso. Durante a seleção de questões, o sistema deve indicar dinamicamente, por meio de um esquema de cores, se uma questão selecionada contribui para a aproximação ou para o distanciamento do objetivo estabelecido na parametrização.

A tela correspondente ao item *Ajustes* deve exibir ao usuário as opções de personalização da interface. Um desses ajustes está relacionado à ordem em que estão dispostas as abas que organizam as questões por *status*.

CONSIDERAÇÕES FINAIS

Na edição 2020-2021 do PICPq foi proposta a construção de um sistema capaz de tornar mais fácil e confiável o trabalho de elaboração da prova do Teste de Progresso. Parte das funcionalidades originalmente previstas para o sistema foram implementadas nesse período, denominado Fase I, e outras foram transferidas para a Fase II, na edição 2022-2023 do PICPq. Nessas duas edições foi possível avançar no desenvolvimento de funcionalidades essenciais ao sistema. Dentre elas está toda a gerência de questões, que envolve o cadastro, a validação e a organização das questões de acordo com o status. Ainda nessas duas fases foram implementados os gráficos para análise da distribuição das questões cadastradas e também a funcionalidade de administração, para a gerência de usuários e seus perfis. Ainda na Fase II foi possível modelar o problema de otimização subjacente à geração automática da prova. Contudo, as implementações do modelo e também das funcionalidades de parametrização e geração manual da prova não foram concluídas e, por isso, foram transferidas para a Fase III.

A Fase III, que se inicia com este trabalho, traz a proposta de concluir a implementação do sistema, ou seja, desenvolver as funcionalidades relacionadas à geração da prova. No entanto, antes de iniciar a implementação desses itens, percebeu-se a necessidade de uma revisão do código construído até o momento. Essa revisão, apresentada aqui, envolveu a correção de algumas falhas de funcionamento do sistema, identificadas nos testes, e também a modernização do sistema segundo aspectos relacionados à sua arquitetura, tecnologias empregadas no desenvolvimento e experiência do usuário.

Acredita-se que o esforço empregado na reestruturação do sistema, desde a interface do usuário até a definição de uma nova *stack* de desenvolvimento, trouxe resultados significativos em um curto espaço de tempo, considerando o início das atividades da edição 2024-2025 do PICPq. Ademais, vale ressaltar que toda a arquitetura pensada para a entrega do produto final do projeto foi desenhada levando em consideração futuras extensões de funcionalidades, entre elas a própria elaboração e geração das avaliações que, no momento, são geridas pelo sistema Qstione. Se concretizado, os resultados beneficiariam duplamente o UNIFESO, atendendo às necessidades de avaliação do seu corpo discente em escopo global.

REFERÊNCIAS

- GIARDINO, C.; UNTERKALMSTEINER, M.; GORSCHKE, T. Software development in startup companies: the greenfield startup model. *IEEE Transactions on Software Engineering*, v. 42, n. 6, p. 585-604, 2015.
- LINDEN, R. *Algoritmos Genéticos: Uma Importante Ferramenta de Inteligência Computacional*. 2ª ed. Rio de Janeiro: Brasport, 2008.
- MICHALIEWICZ, Z. *Genetic Algorithm + Data Structures = Evolution Programs*. Germany: Springer-Verlag, 1996.

MIRANDA, J. F. A.; MORAES, M. B. V. B. PAAI: Programa de Autoavaliação Institucional. In: *Autoavaliação Institucional no UNIFESO – 15 Anos de Avaliação Transformadora*. Vol. 1. Teresópolis: UNIFESO, 2014.

MOREIRA, P.; SEQUEIRA, A.; PEREIRA, S.; RODRIGUES, R.; ROCHA, M.; LOUSA, D. ViralFP: a web application of viral fusion proteins. *Frontiers in Medical Technology*, v. 3, 2021. doi: 10.3389/fmedt.2021.722392.

MORGADO, F. E. F. O Teste de Progresso. In: *Autoavaliação Institucional no UNIFESO – 15 Anos de Avaliação Transformadora*. Vol. 1. Teresópolis: UNIFESO, 2014.

SILVA, E.; GEONIZELI, J. V. S.; FONTE, R. C.; MELLO JÚNIOR, L. C. R. Sistema Web para Geração Automática do Teste de Progresso. In: *Anais do V Congresso Acadêmico Científico do UNIFESO – CONFESO*, Teresópolis: Editora UNIFESO, 2020.

SILVA, E.; GEONIZELI, J. V. S.; FONTE, R. C.; MELLO JÚNIOR, L. C. R. Sistema Web para Geração Automática do Teste de Progresso. In: *Anais do VI Congresso Acadêmico Científico do UNIFESO – CONFESO*, Teresópolis: Editora UNIFESO, 2021.

SILVA, E.; GEONIZELI, J. V. S.; MACHADO, J. V. S.; RUFINO, J. V. Sistema Web para Geração Automática do Teste de Progresso – Fase II. In: *Anais do VII Congresso Acadêmico Científico do UNIFESO – CONFESO*, Teresópolis: Editora UNIFESO, 2022.

SILVA, E.; GEONIZELI, J. V. S.; CORRÊA, F. L. C.; ROCHA, E. F. Sistema Web para Geração Automática do Teste de Progresso – Fase II. In: *Anais do VIII Congresso Acadêmico Científico do UNIFESO – CONFESO*, Teresópolis: Editora UNIFESO, 2023.