

Alumni: Sistema Web para Busca de Cursos e Instituições de Ensino

Gabriel Viaro Leão L. da Silva¹, Michele Cristiani Barion²

Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas
Instituto Federal de Educação, Ciência e Tecnologia de São Paulo – IFSP
CEP 13183-250 – Hortolândia – SP — Brasil

gabriel.leao@aluno.ifsp.edu.br¹, michelefreitas@ifsp.edu.br²

Abstract. *It is important to know what conditions of the academic environment you are seeking to enter are in order to understand whether it is in accordance with your particular tastes and needs, whether in financial, geographic, or cultural terms. Therefore, this article describes the process of developing a web application aimed at people who are looking for a course or academic institution, regardless of organization's educational level. The methodology adopted is based on agile development using Kanban, with iterative testing of user stories to validate the user experience. The application was built with web technologies, allowing the aggregation of information from different institutions in a single location. The information will be provided and reviewed by the platform users themselves, in a collaborative manner. Collaboration will take place freely and optionally, there will be no schedule or need for actions on the part of users, the collaboration must be expressed given the user's interest in the platform. Tests carried out with users found that the platform significantly facilitates the search for academic institutions, thanks to efficient filters and an intuitive interface. The results indicate that the collaborative approach improves the quality of available data, allowing users to make more informed decisions about their educational journey.*

Resumo. *É importante saber quais são as condições do ambiente acadêmico em que se está buscando ingressar para compreender se o mesmo está de acordo com os seus gostos e necessidades particulares, seja na questão financeira, geográfica ou cultural. Dessa forma, este artigo descreve o processo de desenvolvimento de uma aplicação web destinada a pessoas que estão na busca de um curso ou instituição acadêmica, independente do nível de ensino. A metodologia adotada baseou-se no desenvolvimento ágil utilizando Kanban, com testes iterativos de histórias de usuários para validar a experiência do mesmo. A aplicação foi construída com tecnologias web, permitindo a agregação de informações de diferentes instituições em um único local. O fornecimento das informações será dado e revisado através dos próprios usuários da plataforma, de maneira colaborativa. A colaboração acontecerá de forma livre e opcional, não haverá uma agenda ou necessidade de ações por parte dos usuários, esta deve se manifestar dado o interesse do usuário pela plataforma. Os testes realizados com usuários demonstraram que a plataforma facilita significativamente a busca por instituições acadêmicas, pelos filtros eficientes e uma interface intuitiva. Os resultados*

indicam que a abordagem colaborativa melhora a qualidade dos dados disponíveis, permitindo que os usuários tomem decisões mais informadas sobre sua jornada educacional.

1. Introdução

O processo de aprendizagem ocorre de forma única para cada indivíduo, pois a assimilação do conhecimento varia conforme o método de ensino adotado pela instituição acadêmica, o ambiente disponibilizado para esse fim e as interações sociais e intelectuais do aluno com o espaço em que está inserido.

“No que se refere ao cenário escolar, para que a aprendizagem ocorra de forma significativa, é essencial a consideração dos conhecimentos que o aluno traz para a sala de aula, sua forma de compreender, seus interesses. A apresentação de novos conceitos deve estar em sintonia com a realidade do aluno, com o seu contexto de existência, a fim de que ele possa dar sentido e significado aos conteúdos de estudo [...] podendo, assim, tornar-se verdadeiramente o autor de seu processo de aprendizagem”. [NUNES 2011].

Para que o processo de ensino-aprendizagem seja realmente eficaz, é fundamental que o indivíduo esteja inserido em um ambiente que compreenda, no qual possa interagir de forma igualitária e que desperte seu interesse. No entanto, alcançar essas condições exige uma pesquisa criteriosa para identificar instituições alinhadas a esse perfil.

Assim, diante do exposto, este trabalho tem como objetivo principal desenvolver uma solução *web* a fim de centralizar informações relevantes de uma instituição de ensino, como infraestrutura, cultura, valores, horários de funcionamento, endereço e política de cobrança, além de detalhes sobre as atividades extracurriculares, para que futuros estudantes possam consultar seus interesses e suas necessidades. Como objetivos específicos: gerenciar o cadastro de instituições de ensino através do perfil das escolas; criar mecanismos de busca aos usuários que estão fazendo buscas na aplicação; criar *ranking* de instituições em que há mais procura pelos mecanismos de busca.

Este artigo será organizado em seções estruturadas da seguinte forma: a Seção 2 apresentará o referencial teórico; a Seção 3 trará uma análise de trabalhos correlatos; a Seção 4 descreverá a metodologia adotada e os materiais utilizados no desenvolvimento da solução; a Seção 5 abordará o desenvolvimento da solução; as considerações finais serão expostas na Seção 6; e, por fim, as referências serão apresentadas ao final do documento.

2. Referencial Teórico

Esta seção apresenta o referencial teórico utilizado durante o desenvolvimento do trabalho, que será necessário para compreensão de termos, conceitos e nomenclaturas.

2.1. Sistema Web

Os sistemas *web* são aplicações acessíveis via navegador, baseadas em tecnologias, como por exemplo HTML, CSS, JavaScript e PHP, que permitem a comunicação entre cliente (*front-end*) e servidor (*back-end*). Segundo [BENTO 2013], o uso de HTML, CSS e JavaScript é essencial para estruturar e estilizar páginas *web*, o *front-end*. Já para o *back-end*, o PHP é utilizado para gerenciar a lógica do sistema, armazenamento de dados e autenticação.

2.2. Requisitos Funcionais, Diagrama de Caso de Uso e Requisitos Não Funcionais

Um requisito, na engenharia de *software*, pode ser definido como uma condição ou recurso que o sistema deve atender. Conforme [VALENTE 2022], os requisitos determinam o que um sistema deve realizar e sob quais restrições operar.

Os requisitos funcionais referem-se às ações que o sistema deve ser capaz de executar. Eles descrevem comportamentos de entrada e saída ou relações de ação e reação, como, por exemplo, cadastrar uma pessoa na plataforma. Para ilustrar a relação entre um requisito funcional e um usuário, é possível utilizar diagramas de casos de uso, que representam atores (usuários ou entidades do projeto), casos de uso (ações realizadas pelos atores) e os relacionamentos entre eles. O Diagrama de Casos de Uso é uma ferramenta que demonstra de forma clara como as funcionalidades do sistema interagem com seus usuários ou elementos externos, conforme destacado por [BEZERRA 2007].

Já os requisitos não funcionais descrevem atributos ou características de um sistema, como a confiabilidade, que avalia a precisão do sistema na execução de uma determinada função.

2.3. Modelo e Diagrama de Entidade-Relacionamento

O Modelo E-R (MER) é um modelo de dados usado para descrever as informações ou regras de negócio de um projeto ou sistema e seus requisitos, e a parte conceitual da modelagem.

“O MER baseia-se na observação de que o mundo pode ser percebido como um conjunto de objetos, denominados entidades, e pelo conjunto dos relacionamentos entre essas entidades”. [MAIA 2014]

O Diagrama E-R (DER), enquanto o MER é a parte conceitual, é a implementação desse conceito de forma gráfica e documentada, utilizando de ferramentas para abordá-lo.

Essa ideia foi concebida por Peter Chen, divulgado através do artigo *The Entity-Relationship Model-Toward a Unified View of Data* [CHEN 1976]

Na Figura 1 é demonstrado um exemplo básico desse modelo de dados utilizando de um cenário comum no dia-a-dia, onde duas entidades denominadas Cliente - possuidor dos atributos CPF e Nome - e Conta - sem atributos declarados -, respectivamente, se relacionam através do relacionamento “possui”. A cardinalidade desse relacionamento se dá por: um Cliente possui n, um número não específico e limitado de contas, enquanto uma Conta possui apenas um Cliente.

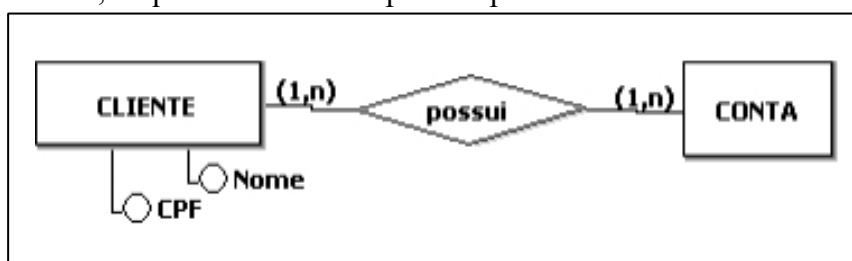


Figura 1. Representação Diagrama E-R (Autor: Teoria da Classificação Facetada e Contribuições para o Modelo Entidade Relacionamento, 2014)

2.4. Banco de Dados

Um banco de dados é um repositório estruturado de informações, organizadas segundo uma lógica definida pelo administrador do banco de dados. Essa organização pode ser realizada utilizando ferramentas de gerenciamento específicas ou por meio de um ambiente de desenvolvimento integrado.

Os bancos de dados são amplamente utilizados em conjunto com aplicações de diferentes tecnologias, funcionando como o equivalente eletrônico de um arquivo físico para armazenamento e recuperação de informações [DATE 2004].

Para este projeto, a fim de solucionar problemas relacionados à responsabilidade, optou-se por utilizar um banco de dados relacional. Esse modelo permite estabelecer relações entre os dados, o que significa que grande parte das operações será gerenciada diretamente pelo banco de dados, reduzindo a complexidade na aplicação.

2.5. Kanban

Kanban é o nome dado a uma metodologia ou ferramenta de gerenciamento de projetos cujo principal objeto é um cartão. Se originou e foi conceituado no Japão, por volta dos anos 70, pelo então vice-presidente da *Toyota Motor Company*, Taiichi Ohno. [GIANESI 1993]

A ferramenta consiste na visualização das tarefas através de cartões, em um quadro ou alguma superfície, seja uma parede ou uma tela, de forma que estejam em colunas apropriadas em relação ao processo da tarefa.

2.6. Framework

Framework, do inglês “estrutura” ou “caixa de ferramentas”, na área de desenvolvimento de software, é um conjunto de códigos abertos que compõem alguma funcionalidade e auxiliam no desenvolvimento do software. O benefício de sua utilização é o aumento de agilidade no desenvolvimento - evitando o retrabalho.

3. Trabalhos Correlatos

Esta Seção apresentará uma análise de soluções já disponíveis no mercado que possuam funcionalidades similares às propostas deste trabalho, como ferramentas que facilitam a busca por instituições de ensino. A pesquisa será realizada utilizando a ferramenta de busca do Google, com palavras-chave como “buscar”, “escola”, “faculdades”, entre outras.

Ela será organizada em subseções, cada uma dedicada à análise de uma solução específica. Ao final, será apresentado um comparativo em forma de tabela, destacando as principais semelhanças e diferenças entre as soluções analisadas.

3.1. Quero Bolsa

A QueroBolsa é o primeiro sistema *web* que aparece como resultado ao combinar algumas palavras-chave e realizar a busca na ferramenta de pesquisa citada anteriormente.

O sistema *web* faz parte de uma iniciativa “Comunidade Quero”, cujo principal ideal é o compartilhamento de informação e conhecimento acerca de cursos e

instituições de ensino superior. É uma ferramenta que não há obrigatoriedade de ter uma conta na plataforma e possui diversas funcionalidades, como a busca por faculdades de graduação e pós-graduação, escolas do nível educacional infantil até o ensino médio, escolas técnicas, escolas de idiomas e cursos livres, como é visto na Figura 2.



Figura 2. Painel de busca de escolas (Autor: QueroBolsa, 2022)

A funcionalidade de busca possui critérios semelhantes entre os níveis educacionais, como a possibilidade de pesquisar pelo nome de um curso ou de uma instituição de ensino, ou também pela localização, valor do orçamento e modalidade - presencial ou a distância. É possível visualizar o resultado de uma busca bem específica, utilizando o nome de uma cidade - Campinas - como parâmetro de busca, somado ao bairro São Bernardo e o ano ou série sendo educação infantil, 5G (cinco anos) na Figura 3.

Figura 3. Resultado de busca na ferramenta (Autor: QueroBolsa, 2022)

Além disso, a QueroBolsa, em sua página de início, sugere e oferece bolsas em instituições próximas ao dispositivo que está acessando, para alguns cursos específicos.

A fim de auxiliar usuários cuja visita ao *site* não tenha tanta objetividade em

buscas específicas, a plataforma também oferece uma ferramenta auxiliar, o teste vocacional gratuito.

Em suma, é uma ferramenta gratuita, completa e que também auxilia seus usuários financeiramente, através das bolsas ofertadas. Possui um sistema de busca e filtros objetivo e claro, além de ter algumas outras funcionalidades auxiliares, como o teste vocacional.

3.2. Educa Mais Brasil

Educa Mais Brasil é uma plataforma que está há mais de 15 anos no mercado, com a principal ideia de auxiliar estudantes impossibilitados de pagar uma mensalidade integral na ingressão a instituições de ensino particulares através de bolsas de estudo parciais.

Para uso, não é obrigatório se cadastrar no portal e possui funcionalidades de busca em instituições de graduação e pós-graduação, escolas do nível educacional infantil até o ensino médio, cursos técnicos, escolas de idiomas, preparatórios para concursos e vestibulares, cursos livres e EJA (educação para jovens e adultos).

Além disso, o portal conta com um assistente virtual cuja principal funcionalidade é auxiliar os visitantes na exploração do site. Ele oferece suporte ao responder dúvidas frequentes e abrir artigos relacionados às questões apresentadas. O conteúdo é organizado por categorias, correspondendo aos diferentes níveis escolares dos cursos ofertados.

A funcionalidade de busca possui critérios semelhantes entre os diferentes níveis escolares, como a possibilidade de pesquisar pelo nome de um curso ou de uma instituição ou até mesmo da série escolar caso seja aplicado. É possível visualizar o resultado de uma busca bem específica, utilizando a localização de cidade - Campinas - como parâmetro de busca, aliado à série escolhida, Infantil G5 - (5 anos), e o bairro sendo o São Bernardo na Figura 4.



The screenshot shows the Educa Mais Brasil website interface. At the top, there are navigation links for 'Como Funciona', 'Login', and 'Mochila'. The main search query is 'Bolsas de Estudo / ESCOLA DO INFANTIL G5 - (5 ANOS) EM CAMPINAS - SP'. The search results are for 'Bolsa para Infantil G5 - (5 anos) - Educa Mais Brasil'. The search filters are set to 'São Bernardo', 'Campinas', and 'Infantil G5 - (5 anos)'. The results list a school named 'Berçário e Brinquedoteca Aconchego' located in São Bernardo, with a monthly fee of R\$ 490.00, a 50% scholarship, and a total cost of R\$ 5,880.00. The page also lists nearby neighborhoods with available schools like Cambuí, Centro, Jardim Chapadão, etc.

Figura 4. Resultado de uma busca por escolas usando filtros (Autor: EducaMais Brasil, 2022)

Outra funcionalidade que pode ser observada em sua página de início é a sugestão de bolsas de estudo, tanto para faculdades quanto cursos.

Por fim, a ferramenta possui cartões de *feedback* também na página inicial, uma forma dos usuários novos terem contato com as opiniões de alguns usuários vigentes.

3.3. Comparação de Funcionalidades

Nesta subseção será comparado algumas funcionalidades encontradas nos trabalhos correlatos analisados anteriormente em relação ao que é proposto neste trabalho.

O critério de seleção das funcionalidades foi baseado na importância em relação ao objetivo deste trabalho. O comparativo entre os trabalhos correlatos e a aplicação proposta neste trabalho pode ser resumido ao que é visto na Figura 5.

	QueroBolsa	Educa+ Brasil	Alumni
Obrigatoriedade de registro para uso da plataforma	✗	✗	✗
Registro de usuários	✓	✓	✓
Painel de login	✓	✓	✓
Busca e aplicação de filtros na busca	✓	✓	✓
Seção de comentários no curso ou instituição	✗	✗	✓
Supoorte de uso (assistente, documentação, etc..)	✓	✓	✗
Gratis	✓	✓	✓
Open-source	✗	✗	✓
Supoorte a todas as instituições de ensino e cursos	✗	✓	✗
Matrícula pela plataforma	✓	✓	✗

Figura 5. Comparativo de funcionalidades presentes em sistemas distintos
(Autor: Autoral)

A maior inovação que essa aplicação busca trazer é o senso de comunidade entre seus usuários, visto que grande parte das informações será introduzida e constantemente atualizada de maneira colaborativa, dado a funcionalidade de Seção de comentários no curso ou Instituição. Além disso, ao passo que outros sistemas semelhantes possuem influência de anúncios e outros meios de marketing, o Alumni busca alavancar as instituições de forma orgânica, com o engajamento de seus próprios usuários.

4. Metodologia e Materiais

A partir do levantamento apresentado na Seção 3, foi feita a análise de requisitos funcionais e não funcionais dos trabalhos correlatos e, a partir dessa análise, foram definidos os requisitos deste projeto. Foi levado em consideração o que poderia ajudar a solucionar o problema dos usuários vigentes somado ao escopo do objetivo proposto, além de eventuais inovações em relação aos concorrentes.

Tendo os requisitos bem definidos, para gerir e planejar a implementação dos mesmos foi utilizado a metodologia ágil Kanban. Em seguida, tem-se o processo de criação de diagramas de caso de uso com o intuito de relacionar os requisitos funcionais com os usuários, isto é, definir o relacionamento entre eles para, então, construir a modelagem de dados utilizando do MER e do DER para implementação no banco de dados.

Com a diagramação de entidade-relacionamento estabelecida, o banco de dados foi estruturado conforme a documentação gerada a partir dos processos anteriores. *Databases*, tabelas e usuários moderadores do banco de dados foram criados.

Com isso, a prototipação do sistema foi desenvolvida para visualização dos requisitos e seus fluxos de tela. Nessa etapa, é esperado que ocorra eventuais

refatorações de telas em razão da melhoria contínua.

O processo de desenvolvimento do *software* foi feito e testado tendo o embasamento de histórias de usuários, para garantir a confiabilidade dos métodos utilizados no programa e segurança em implementações futuras.

Os materiais responsáveis pelo gerenciamento do projeto são: Trello, ferramenta para prática do Kanban; Draw.io como ferramenta de modulação e Wireframe, ferramenta para prototipação de telas. Para o *front-end*, parte visível do *software*, foi utilizado HTML e CSS para a formatação e marcação e JavaScript como linguagem de programação auxiliar. No *back-end*, PHP como linguagem de programação orientada a objetos, phpMyAdmin para o sistema de gerenciamento de banco de dados MySQL e, por fim, o versionamento e distribuição através do GitHUB.

5. Desenvolvimento

Nesta Seção será abordado em subseções as etapas tomadas - definidas na Seção 4 - para o desenvolvimento da aplicação apresentada neste artigo. A aplicação foi dividida em pequenas tarefas, onde o quadro Kanban - com ajuda do Trello - mostrou-se eficaz para gerenciamento dos requisitos já implementados, os que estão sendo implementados e os que serão implementados, em conjunto a informações de comportamentos esperados e suas histórias de usuário.

5.1. Arquitetura do Software

Para definir uma hierarquia organizada dos arquivos do *software*, a arquitetura mostrou-se indispensável, visto que o projeto deverá seguir um padrão: arquivos divididos por responsabilidades no fluxo de obtenção da informação. Tendo como prioridade um baixo acoplamento, onde componentes não possuem uma forte relação de dependência entre si, e uma alta coesão, onde os componentes possuem uma única responsabilidade e, consequentemente, há uma maior facilidade de implementação e manutenção do *software*.

Com isso, a primeira arquitetura pensada para o *software* pode ser visto na Figura 6, que é, essencialmente, dividida em três camadas: o *frontend*, que representa a camada de visualização de dados; o *backend*, que representa a parte lógica do sistema; e a *database*, que representa o arquivamento da informação, o banco de dados.

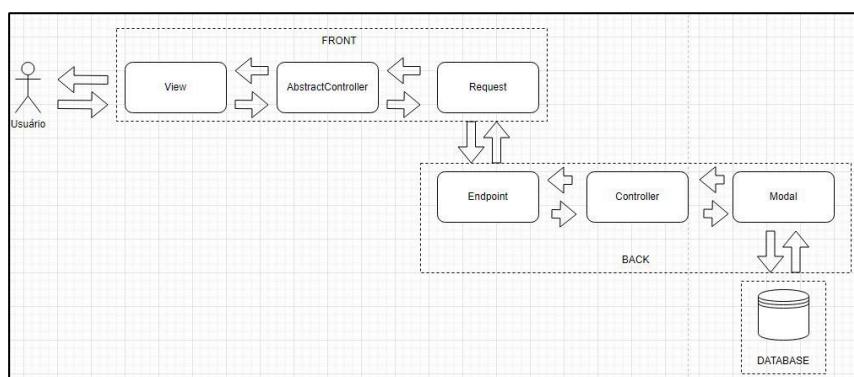


Figura 6. Primeira arquitetura pensada para o software, alto acoplamento (Autor: Autoral)

No entanto, após uma análise de implementação, foi verificado que o acoplamento entre os componentes ficou maior do que o desejado para o *software* e, com isso, a arquitetura foi refatorada e pensada de forma que usasse o modelo *Model*

View Controller (MVC), representado na Figura 7. A arquitetura mantém a divisão entre três camadas, onde a *View* representa a visualização dos dados; o *AbstractController* representa o tratamento lógico, ambas partes da primeira camada, o *front*; na segunda camada, o *back*, tem-se o componente *Modal*, que representa a recuperação dos valores que são buscados na terceira e última camada, a *Database*, e por fim, seguindo o fluxo de dados, os mesmos são resgatados até a *View*, para visualização do usuário.

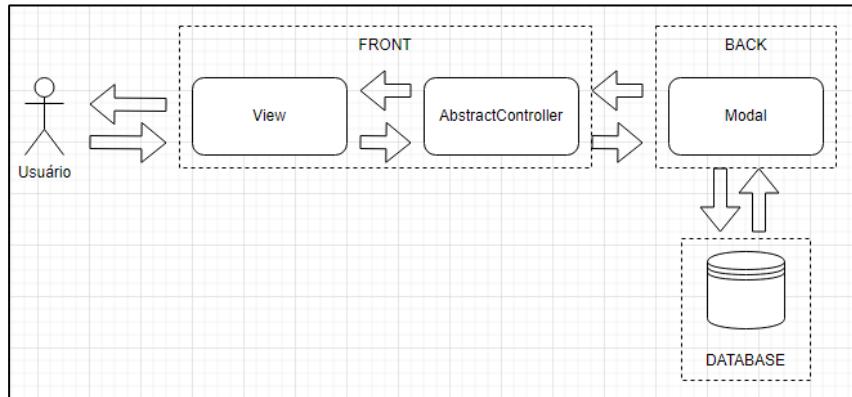


Figura 7. Segunda arquitetura definida para o software, baixo acoplamento
(Autor: Autoral)

5.2. Levantamento de Requisitos

Os requisitos definidos a partir da análise dos trabalhos correlatos somados às ideias do projeto são listados a seguir, em nenhuma ordem em particular, na Tabela 1.

Tabela 1. Tabela de levantamento de requisitos funcionais e não-funcionais

NOME	CLASSIFICAÇÃO	TIPO REQUISITO
Navbar	Interface	Funcional
Armazenamento de localização do usuário	Interface	Funcional
Ícone para cadastro e/ou entrar em uma conta	Interface	Funcional
Lógica de autenticação para uma conta	Segurança	Funcional
Filtros de busca e ordenação para as instituições	Interface	Funcional
Cards de instituições em destaque na página inicial	Interface	Funcional
CRUD (create, update & delete) de instituição	Operacional	Funcional
Feedback (comentários) por instituição	Interface	Funcional
Segurança - criptografia de informações sensíveis	Segurança	Não-Funcional
Performance - processos assíncronos e em background	Performance	Não-Funcional
Portabilidade - disponível em múltiplos dispositivos	Interface	Não-Funcional
Usabilidade - sistema fácil de aprender e usar	Interface	Não-Funcional

Com os requisitos definidos, uma análise mais aprofundada tomará parte utilizando a metodologia ágil Kanban, em conjunto com a ferramenta Trello. Em suma, cada cartão representa um requisito, onde nele apresentará informações pertinentes para seu desenvolvimento.

Estão organizados inicialmente em colunas: Backlog, indicativo de ideias futuras; A fazer, onde estão os cartões cujo processo de desenvolvimento ainda não foi iniciado; Em andamento, são os cartões cujo processo de desenvolvimento já foi iniciado mas ainda não foi finalizado; Feito, são os cartões já finalizados e desenvolvidos, que não apresentam nenhum problema ou dúvida quanto ao seu funcionamento; Bugs, são os cartões que indicam uma funcionalidade que não está apresentando o comportamento esperado. Como indica a Figura 8.

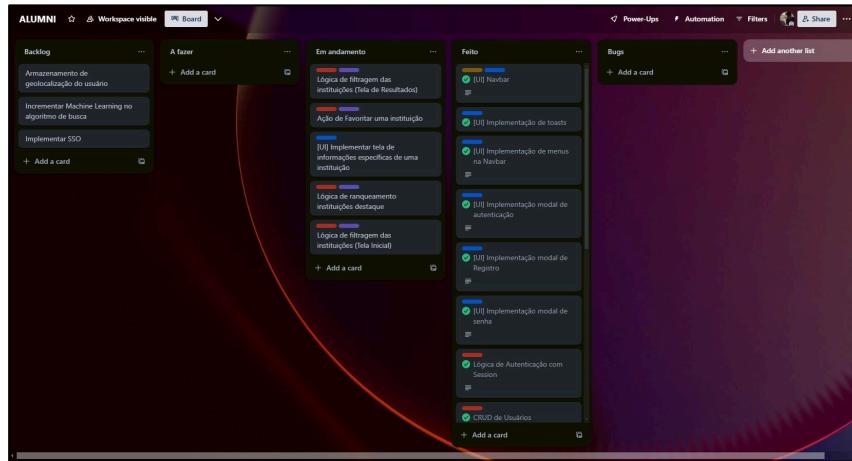


Figura 8. Board Kanban da aplicação Alumni no Trello (Autor: Autoral)

Para identificar rapidamente do que se trata cada cartão, o título pode brevemente dar um indicativo, mas o principal é a organização via etiquetas: UI, em azul, estabelece que é uma tarefa que envolve mudança de interface; BACKEND, em vermelho, estabelece que é uma tarefa com lógica de processamento de informações; e REVISE-ME, em roxo, estabelece que é uma tarefa que necessita de revisão, seja por conta do comportamento ou por estar em conflito com alguma outra regra do sistema.

As histórias de usuário, utilizadas para validar se o requisito cumpre o papel esperado, estão dentro de cada cartão. Na Figura 9 é possível ver as histórias de usuário do requisito "Navbar", que é a barra de navegação superior do sistema.

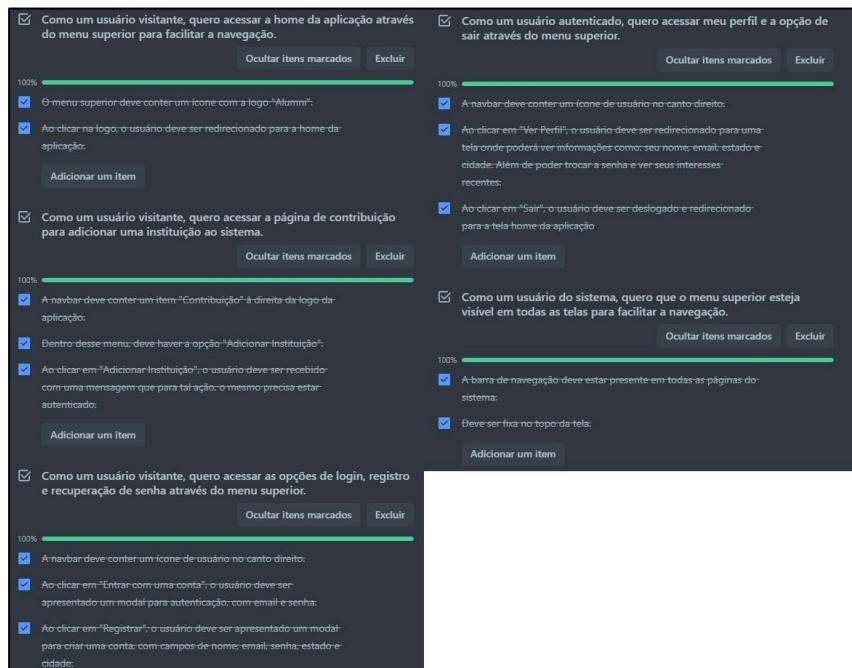


Figura 9. Histórias de usuário do requisito "Navbar"(Autor: Autoral)

5.3. Diagramas de Casos de Uso

Com o objetivo de demonstrar as relações que os usuários (atores) terão com o sistema, embasado pelos requisitos definidos anteriormente, o diagrama de casos de uso é presente na Figura 10, de cima para baixo, cada balão representa uma função do

sistema.

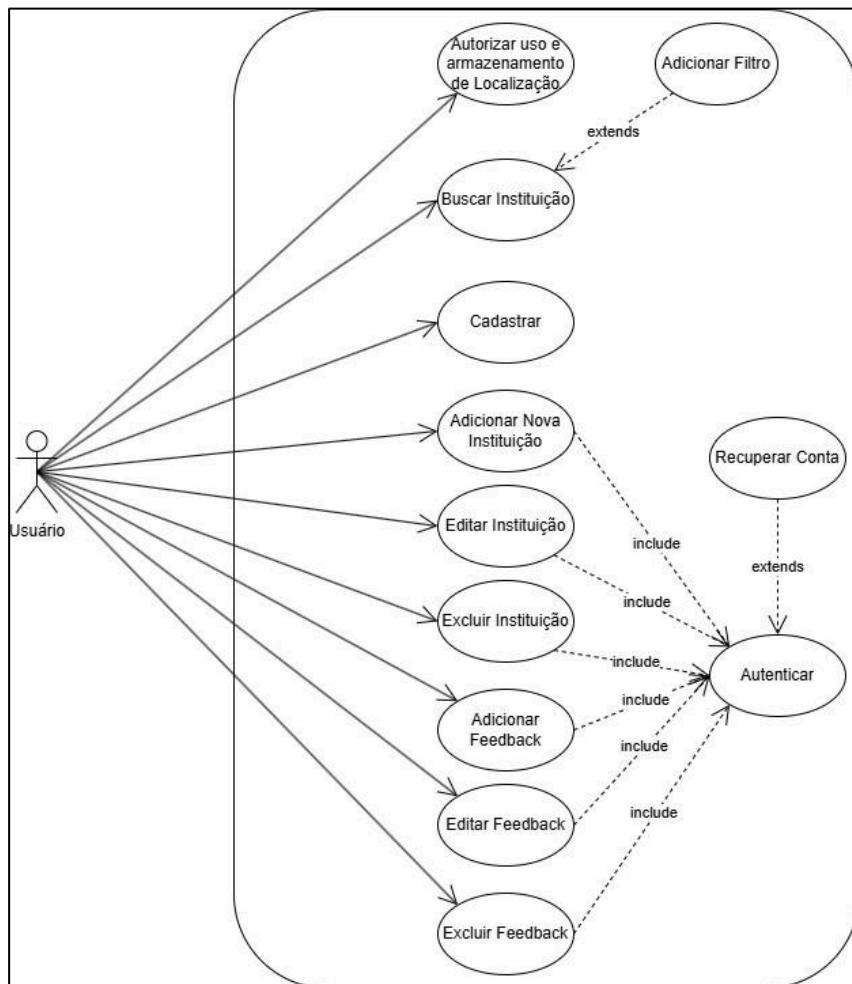


Figura 10. Representação dos relacionamentos através da diagramação (Autor: Autoral)

Autorizar uso e armazenamento de localização - RF01: A autorização de uso e armazenamento de localização é uma projeção futura, que não será incorporada durante o desenvolvimento do *software* no escopo deste artigo; seu objetivo é melhorar a experiência do usuário armazenando, com a aprovação do mesmo, a localização geográfica do cliente através do uso de sessão no PHP. Com isso, os resultados serão filtrados por proximidade.

Buscar Instituição - RF02: A busca de uma instituição é composta por filtros, mas não há a obrigatoriedade do uso de tal. As opções de filtragem são: nome, localização (estado ou cidade), valor, modalidade (presencial ou remoto) e tipo de instituição (escola, faculdade, curso de idiomas, etc...). Caso não sejam definidos filtros, a busca trará todas as instituições cadastradas na plataforma.

Cadastrar - RF03: O cadastro é tomado como medida opcional, o sistema exige nome, email, estado e cidade para a criação de uma conta.

Autenticar - RF10: A autenticação do usuário não é obrigatória. No entanto, algumas funções do sistema, como Adicionar Nova Instituição - RF04, Editar Instituição - RF05, Excluir Instituição - RF06, Adicionar Feedback - RF07, Editar Feedback - RF08 e Excluir Feedback - RF09 só estão disponíveis a usuários autenticados. Além disso, tem-se a possibilidade de recuperar a conta caso o usuário

tenha esquecido a senha, através de um envio de email - Recuperar Conta - RF11.

5.4. DER

A modelação de dados foi feita a partir do conceito de Modelo Entidade-Relacionamento, que consiste na representação dos objetos do mundo real: usuários, instituições, comentários e avaliações, no contexto deste projeto; através de entidades que se relacionam: como o usuário interage com a instituição e de que modo se dá esse relacionamento.

O uso do Modelo ER serve de parâmetro, abstrato de certa forma, para auxílio do desenvolvimento das classes no código e também qual será a estrutura da regra de negócios: o que uma entidade pode fazer com outra, sob determinadas condições, por exemplo. É importante notar, no entanto, que trata-se de uma modelagem, isto é, uma representação alto-nível dos relacionamentos que estarão presentes no *software*, cujas entidades receberão uma camada de polimento maior, como atributos auxiliares, métodos e funções.

Nesta fase da proposta, será apresentado o Diagrama Entidade-Relacionamento, considerando a aplicação das etapas da normalização diante do MER adotado no projeto. Essa representação, utilizada na implementação física do banco de dados, utilizando o phpMyAdmin, é o visto na Figura 11.

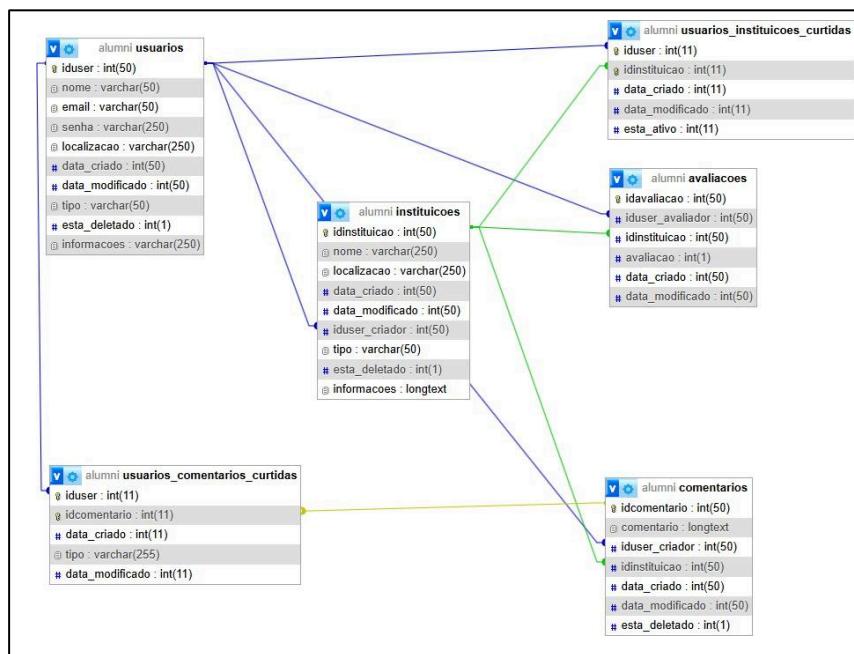


Figura 11. Representação dos relacionamentos através da diagramação (Autor: Autoral)

5.5. Prototipação de Telas

Com o intuito de melhor visualizar e organizar as funções do sistema, um protótipo da tela inicial foi desenvolvido utilizando a ferramenta *Wireframe*, cujo resultado pode ser visto na Figura 12.

De cima para baixo, os elementos que compõem a tela são: A *navbar* - barra de navegação - presente em toda e qualquer tela do sistema. Seu objetivo é facilitar a navegação entre o mesmo e oferecer alguns serviços, como a logo do projeto à esquerda, cujo clique redireciona o usuário à tela inicial; a mostra e coleta da localização do

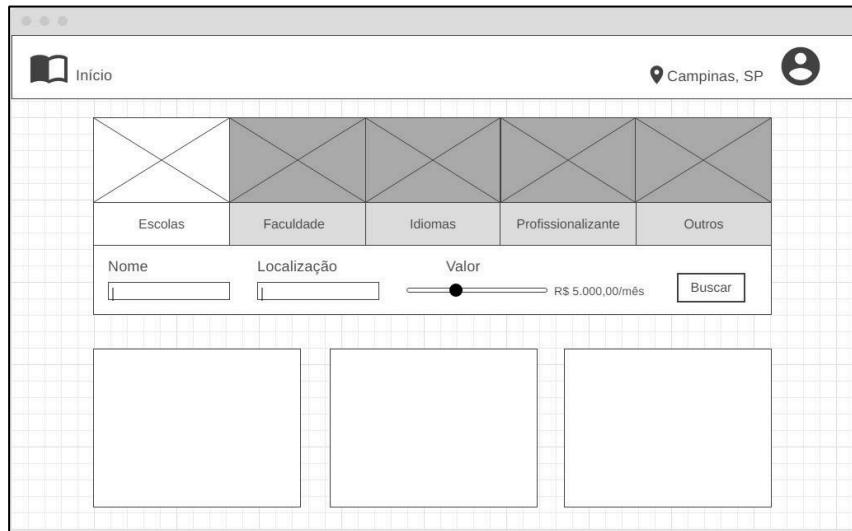


Figura 12. Protótipo da tela inicial (Autor: Autoral)

usuário, com o consentimento do mesmo e o ícone, ou foto de perfil, caso o usuário tenha adicionado uma.

Além disso, ao clicar no ícone do usuário, algumas opções serão ofertadas em formato de um seletor *dropdown*, como por exemplo: entrar e cadastrar (para usuários visitantes, não cadastrados) ou editar perfil, ver informações (para usuários cadastrados).

Logo abaixo é possível ver os elementos da tela inicial, que são: um retângulo acima de três quadrados. O retângulo é, na tela, o responsável por organizar a filtragem das instituições de ensino que, após aplicados (ou não) e dado o clique o botão "Buscar", redirecionará o usuário a uma outra tela, contendo os resultados da busca.

Os três quadrados representam três instituições de ensino não necessariamente relacionados aos filtros inseridos na parte superior da tela. Essas instituições estão à mostra ao usuário por ordem de relevância - quantas interações, negativas ou positivas, a mesma recebeu nos últimos sete dias (uma semana). Caso não tenha instituições que tenham cumprido essa regra, nenhuma será mostrada.

As cores predominantes da plataforma serão azul, branco e cinza, defendendo a ideia de inteligência e sua conexão com ambientes acadêmicos. [HELLER 2021].

Adiante, tem-se as implementações finais das telas da aplicação. Na Figura 13 e 14, alguns elementos foram ajustados em relação à prototipação anterior, dentre eles: a remoção da localização da barra de navegação (navbar), elemento superior da tela; a adição de filtros por modalidade no retângulo superior e a adição do botão "Ver todas", que refere a ver todas as instituições cadastradas na plataforma. Na Figura 13 tem-se os mecanismos de busca na página inicial, destacando os filtros de busca que os usuários irão utilizar. Na Figura 14, é possível ver as instituições em destaque - as que tiveram mais curtidas - aparecerão, concluindo o objetivo de ranqueá-las de acordo com a procura.

O menu do usuário, elemento ao superior direito da Figura 15, possui as opções de "Entrar com uma conta", para autenticá-lo e que possa usufruir de algumas funções do sistema; "Cadastrar", para criar uma conta e "Esqueceu sua senha?", caso o visitante possua uma conta mas não se lembre da senha de autenticação. Essas opções são resultados implementais dos requisitos levantados na Tabela 1: Navbar e Ícone para cadastrar e/ou entrar em uma conta. É importante destacar que essas opções servem

conforme o nível de autenticação do usuário que, neste exemplo, é visitante.

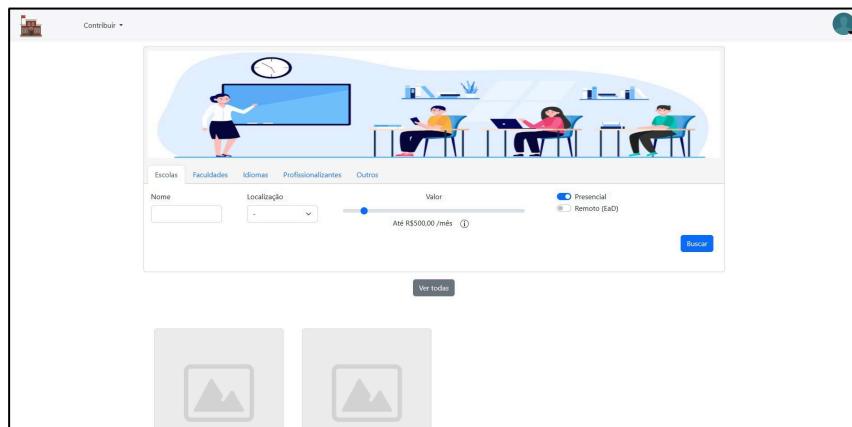


Figura 13. Implementação da tela inicial (Autor: Autoral)

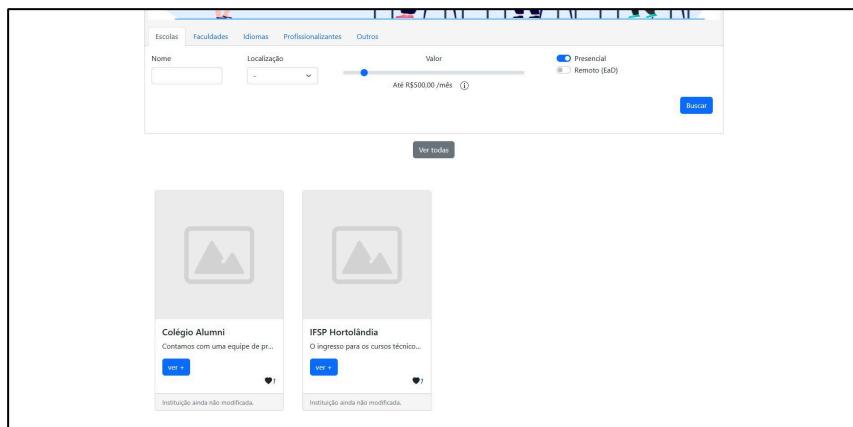


Figura 14. Implementação da tela inicial (Autor: Autoral)

Ao clicar em cada um dos elementos da Figura 15, respectivamente, tem-se o resultado nas Figuras 16, 17 e 18. Com isso, além dos elementos de interface, também foram implementados os requisitos funcionais de autenticação de usuário, cadastro de usuário e recuperação de senha.

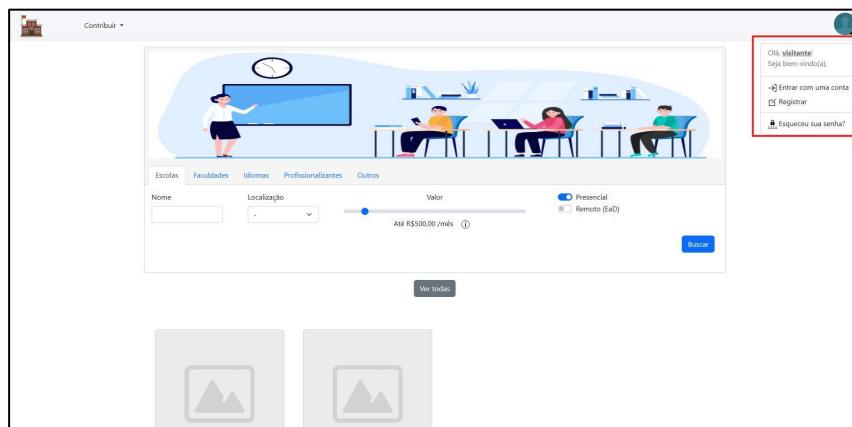


Figura 15. Tela inicial com o menu do visitante aberto (Autor: Autoral)

Quando a autenticação já foi realizada através da tela da Figura 16, as opções "Ver perfil" e "Sair", presentes na Figura 19 podem ser vistas ao clicar no ícone do usuário no canto superior direito da tela. Ao clicar em "Ver perfil", o usuário é levado para uma página onde está concentrada todas as suas informações relevantes. Além disso, no lado direito da página, consta também os interesses do usuário, divididos em duas sessões: Favoritados - são as instituições em que o usuário curtiu - e Interesses - levantamento dos tipos de instituições que o usuário favoritou.

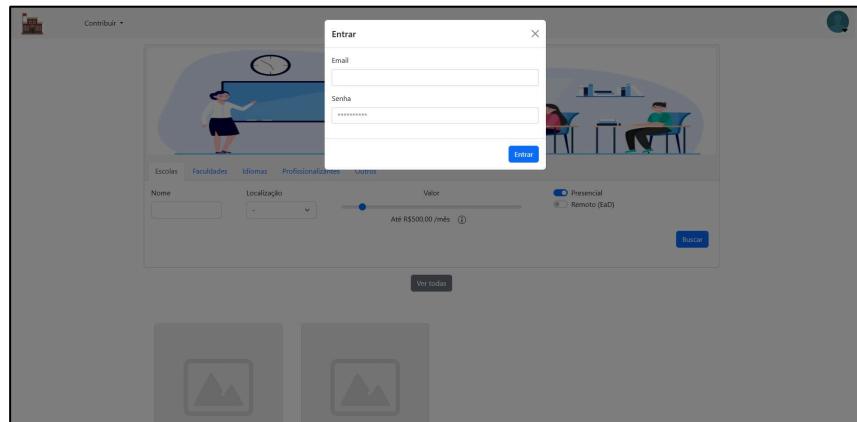


Figura 16. Implementação da tela para autenticação de usuário (Autor: Autoral)

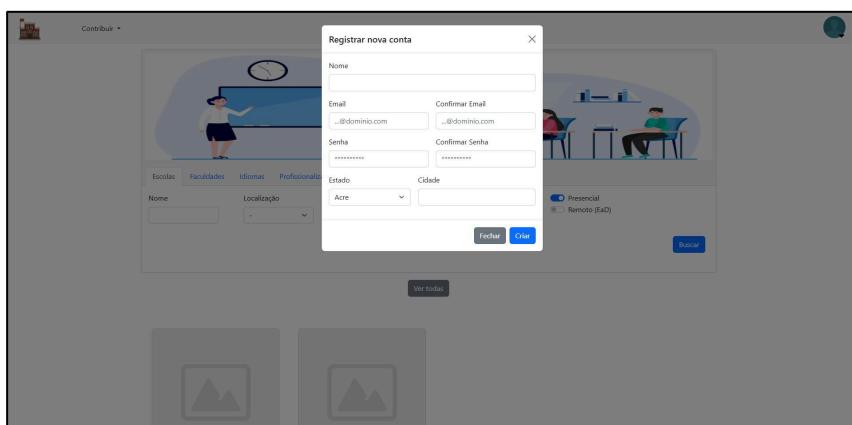


Figura 17. Implementação da tela para cadastro de usuário (Autor: Autoral)

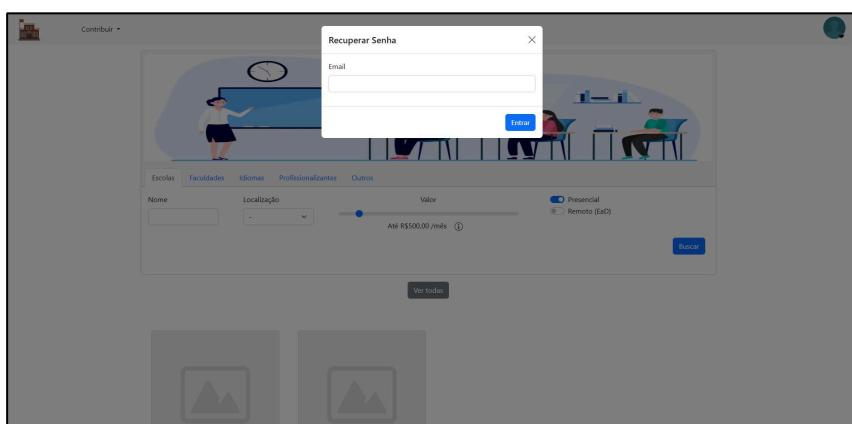


Figura 18. Implementação da tela para recuperar senha de usuário (Autor: Autoral)

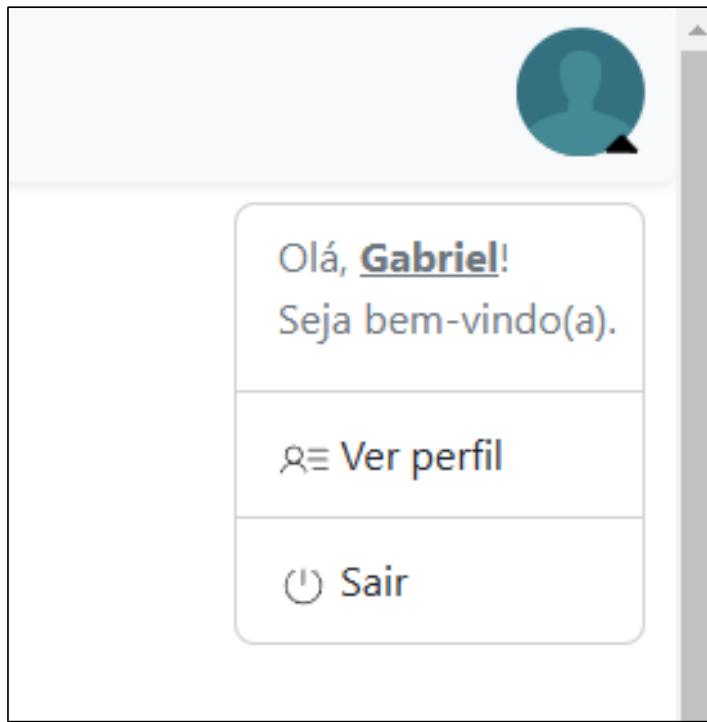


Figura 19. Zoom da tela inicial com o menu do usuário autenticado aberto (Autor: Autoral)

Na Figura 20, tem-se em uso o principal requisito da aplicação - a busca por instituições de ensino dado determinado filtro. A filtragem ocorre de forma que o resultado deve atender a pelo menos um dos critérios de cada modalidade: Tipo, Modalidade, Localização, Valor por mês e Instituição Inclusiva, caso sejam preenchidos. A Figura 21 é invocada ao clicar no botão "ver +", visível nos cartões de cada instituição em diferentes partes da aplicação. Dentro dessa tela destaca-se todas as informações da instituição e também todos os comentários referentes à mesma. Também é possível, nesta tela, editar as informações que o usuário julgar estarem distantes da realidade ou deixar um comentário próprio, como visto em alguns exemplos na figura.

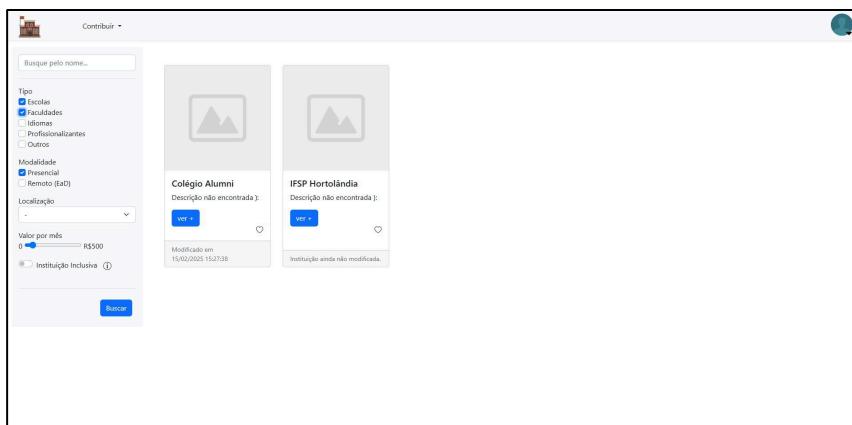


Figura 20. Implementação da tela de resultados de busca (Autor: Autoral)

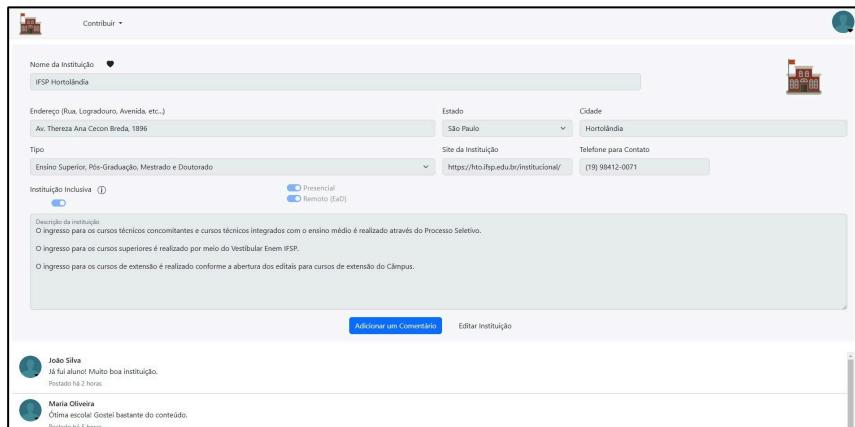


Figura 21. Implementação da tela visualizar instituição (Autor: Autoral)

6. Conclusão e Trabalhos Futuros

A aplicação Alumni foi desenvolvida para centralizar informações relevantes sobre instituições de ensino, proporcionando aos usuários uma busca mais eficiente por instituições de ensino que atendam às suas necessidades individuais, sejam elas quais forem. O desenvolvimento seguiu uma abordagem ágil, utilizando Kanban para o gerenciamento de tarefas, que garantiu uma implementação estruturada e iterativa.

A inovação, quando confrontado com outras soluções semelhantes, é a abordagem colaborativa adotada na plataforma, onde os usuários não possuem uma obrigatoriedade. Essa abordagem mostrou-se eficaz para manter os dados atualizados e relevantes, aumentando a confiabilidade das informações disponibilizadas. Além disso, foi capaz de utilizar esses parâmetros para organicamente ordenar as instituições.

Com relação às dificuldades para o desenvolvimento da aplicação e do artigo, o levantamento e gerenciamento dos requisitos funcionais foi a etapa mais difícil. A necessidade de documentar detalhadamente cada funcionalidade antes da implementação exigiu um grande esforço, mas resultou em um desenvolvimento mais organizado e eficiente.

Para desenvolvimento deste trabalho e até mesmo destacando as competências e habilidades adquiridas no decorrer do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas, houve aplicação de conceitos associados às disciplinas de Algoritmo e Lógica de Programação, Desenvolvimento de Sistemas Web, Banco de Dados, Engenharia de Software, Programação Orientada a Objetos, Metodologia de Pesquisa e Projeto de Sistemas para planejamento e desenvolvimento teórico-técnico.

Para trabalhos futuros, propõe-se a adição de um sistema de geolocalização, permitindo que a busca de instituições seja feita automaticamente com base na localização do usuário. Além disso, recomenda-se o desenvolvimento de uma versão otimizada para dispositivos móveis, garantindo uma experiência de navegação mais fluida e acessível. Outra melhoria possível seria a implementação de um algoritmo de recomendação, utilizando *machine learning* para sugerir instituições de acordo com o perfil e interesses do usuário. Ademais, implementar um sistema de fila para que cada edição e adição de instituições por parte dos usuários seja analisada e aprovada por um usuário mediador. Por fim, implementar autenticação *Single Sign-On* com os principais provedores de identidade, como Google e Microsoft, possibilitando o login unificado e facilitando o acesso do usuário.

Com essas melhorias, espera-se que o Alumni se torne uma ferramenta ainda

mais eficiente e completa para a escolha de instituições de ensino, impactando positivamente a jornada acadêmica de seus usuários.

Referências

- BENTO, E. (2013). *Desenvolvimento Web com PHP e MySQL*. Casa do Código, 1 edition.
- BEZERRA, E. (2007). *Princípios de análise e Projeto de sistemas com UML*. Elsevier, 2 edition.
- CHEN, P. (1976). The entity-relationship model-toward a unified view of data. *ACM Transactions on Database Systems*, Vol. 1, No. 1. Disponível em: https://edisciplinas.usp.br/pluginfile.php/1769466/mod_resource/content/1/p9_chen.pdf. Acesso em : 02 jun.2022.
- DATE, C.; LIFSCHITZ, S. (2004). *Introdução a sistemas de bancos de dados*. Campus, 8 edition.
- GIANESI, I.; CORREA, H. (1993). *Just in Time MRP II e OPT. Um Enfoque Estratégico*. Atlas, 2 edition.
- HELLER, E. (2021). *A psicologia das cores*. Olhares, 1 edition.
- MAIA, R.; ALVARENGA, L. (2014). Teoria da classificação facetada e contribuições para o modelo entidade relacionamento. *Prisma.com*. Disponível em: <https://ojs.letras.up.pt/index.php/prismacom/article/view/1870/1703>. Acesso em: 02 jun. 2022.
- NUNES, A.; SILVEIRA, R. (2011). *Psicologia da Aprendizagem*. EdUECE, 3 edition.
- VALENTE, M. (2022). *Engenharia de Software Moderna*. Independente, 1 edition.