

PLATAFORMA DE FINANCIAMENTO COLETIVO ENTRE DESENVOLVEDORES E INVESTIDORES

COLLECTIVE FINANCING PLATFORM BETWEEN DEVELOPERS AND INVESTORS

Mariana Sampaio do Nascimento*
Pedro Henrique Coutinho Martins**
Giuliano Lacerda Dall' Armellina***

RESUMO

Com tecnologias como inteligência artificial, metaverso, computação em nuvem, com novas formas de desenvolver, com os hardwares evoluindo, está começando a ter no mercado muitas empresas grandes e pequenas, e também desenvolvedores solos, criando novos sistemas, novas tecnologias, e além disso existe uma demanda na parte dos consumidores de adquirir todas essas novas tecnologias, novas ideias que estão sendo colocadas em prática. Com isso iremos desenvolver um site para que as pessoas que tenham algum projeto, publiquem ele e visa financiar o projeto tanto monetariamente quanto com desenvolvedores interessados em participar do projeto, semelhante aos sites Catarse e Kickstarter. Para o desenvolvimento do site, buscamos referências em sites que tenha essa premissa de financiamento, e no nosso processo seguimos algumas etapas do processo da engenharia de software, também foram feitos alguns modelos para a elaboração do banco e definimos os requisitos funcionais e não funcionais, tudo isso para facilitar o desenvolvimento do site.

Palavras-chave: Crowdfunding. Website. Desenvolvimento. Investimento.

ABSTRACT

With technologies such as artificial intelligence, the metaverse, and cloud computing, along with new ways of development and evolving hardware, the market is witnessing the emergence of numerous companies, both large and small, as well as solo developers. They are actively creating new systems and technologies. Additionally, there is a growing consumer demand to acquire these new technologies and implement fresh ideas. In response to this trend, we are in the process of developing a website where individuals with projects can publish them. The platform aims to facilitate the funding of projects both financially and by attracting developers interested in participating, similar to platforms like Catarse and Kickstarter. In the development of the website, we have sought inspiration from platforms that share a similar funding premise. Our process follows several stages of the software engineering process. We have also created models for the database structure and defined both functional and non-functional requirements. All of these efforts are aimed at streamlining the website development process.

Keywords: Crowdfunding. Website. Development. Investment. Crowdfunding.

* Faculdade de Tecnologia, Ciência e Educação (FATECE). mariana.sampaionascimento@gmail.com

** Faculdade de Tecnologia, Ciência e Educação (FATECE). pedrohcmartins@hotmail.com

*** Faculdade de Tecnologia, Ciência e Educação (FATECE). giulianold@hotmail.com

Introdução

O cenário da indústria de tecnologia tem testemunhado um crescente interesse na colaboração, inovação e empreendedorismo no mundo do desenvolvimento de software. Nesse contexto dinâmico, a presente pesquisa visa investigar uma abordagem inovadora destinada a fomentar a criatividade e a colaboração entre desenvolvedores de software, investidores e entusiastas.

Esta exploração busca, de maneira central, a concepção de uma plataforma de interação abrangente e singular, onde a conexão entre mentes criativas e recursos valiosos possa prosperar. Inspirada em modelos de crowdfunding e economia colaborativa, esta plataforma proposta visa proporcionar um ambiente inclusivo. Desenvolvedores talentosos podem compartilhar e monetizar suas criações, ao mesmo tempo em que estabelecem conexões com indivíduos interessados em apoiar iniciativas inovadoras.

Um dos elementos que se destaca como diferencial nessa plataforma é a dualidade de oportunidades oferecidas aos desenvolvedores: seja a compra direta de projetos por investidores em potencial ou a possibilidade de crowdfunding, fornecendo uma via direta de apoio financeiro à comunidade interessada em projetos promissores. Essa abordagem visa empoderar desenvolvedores a concretizarem suas visões e projetos, enquanto também permite que os usuários encontrem oportunidades de projetos e sistemas alinhados com suas afinidades, agregando valor à comunidade colaborativa e inovadora.

Portanto, ao longo deste projeto, investigaremos a viabilidade técnica, e os possíveis desdobramentos de uma plataforma capaz de prover a divulgação de ideias e projetos relacionados ao desenvolvimento de software, criando oportunidades de patrocínio e venda, impulsionando o crescimento na indústria de desenvolvimento de software. Por meio de uma análise abrangente e da consideração de múltiplas abordagens, esperamos:

- Compreender as dificuldades que desenvolvedores iniciantes enfrentam;
- Confirmar a existência de nicho de mercado relacionado a uma plataforma para o patrocínio de ideias e projetos;
- Identificar os requisitos de uma plataforma que atenda a demanda de patrocínio de ideias e projetos;
- Prover o projeto e execução de uma plataforma com os requisitos anteriormente levantados;
- Avaliar se a plataforma atende a indústria de software;

- Disponibilizar em site público a plataforma desenvolvida.

1 Referencial Teórico

Neste tópico será abordado um pouco sobre o Crowdfunding (Financiamento Coletivo), sobre a Engenharia de Software e a Análise de Requisitos que foram usados para o desenvolvimento do projeto

1.1 Crowdfunding

A economia criativa abrange uma ampla gama de setores, desde o artesanato tradicional até as indústrias culturais, envolvendo comunicação, telecomunicações e informática. Os empreendedores desempenham um papel fundamental, sendo responsáveis por criar produtos e serviços inovadores e por introduzir novas formas de organização e recursos. Eles são vistos como agentes de "destruição criativa", desafiando a ordem econômica existente.

No âmbito da economia criativa, destaca-se o empreendedor cultural brasileiro, que precisa inovar e usar sua criatividade para produzir e difundir a cultura. Projetos culturais, como música, cinema e vídeo, exemplificam produtos criativos financiados por meio do crowdfunding.

Essa plataforma permite que empreendedores produzam conteúdo, como vídeos, textos e orçamentos, para atrair o apoio financeiro dos prosumidores, que são os produtores-consumidores (AGUIAR, 2016; AVENI; PINTO, 2014; ALMEIDA, 2020).

Os prosumidores são diversos, incluindo amigos, familiares, parceiros de trabalho, fãs, ativistas e curiosos, cada grupo com motivações específicas para apoiar projetos de crowdfunding. A ênfase recai na multidão (crowd) como parte essencial do crowdfunding, e o sucesso de uma campanha depende da capacidade de reunir prosumidores interessados.

A formação de agrupamentos de prosumidores em torno de campanhas de crowdfunding é voluntária, temporária e tática. Esses grupos são flexíveis e duram apenas enquanto o projeto é financiado. Diversos nichos de prosumidores podem se envolver em campanhas de crowdfunding, desde ativistas e fãs até amigos e familiares (AGUIAR, 2016; AVENI; PINTO, 2014; ALMEIDA, 2020).

A afinidade dos prosumidores com o tema do projeto é um fator crucial para o sucesso de uma campanha. Projetos alinhados com questões políticas, sociais ou de interesse geral têm mais probabilidade de atrair prosumidores engajados. Fãs, em particular, são eficazes na mobilização de outros fãs e na divulgação de campanhas em suas redes sociais.

A ArtistShare, fundada em 2003, é a primeira plataforma do mundo a conectar artistas e fãs para compartilhar o processo criativo e financiar projetos artísticos. Atualmente, existem inúmeras plataformas de crowdfunding em todo o mundo, cada uma com seu foco específico, oferecendo serviços como atendimento aos empreendedores e prosumidores, eventos relacionados ao crowdfunding e manuais de desenvolvimento de projetos (AGUIAR, 2016; AVENI; PINTO, 2014; ALMEIDA, 2020).

A modalidade de crowdfunding baseada na recompensa, na qual o empreendedor é responsável por entregar recompensas aos apoiadores, é destacada como a mais pesquisada e a mais relacionada ao mercado cinematográfico. Plataformas como Kickstarter e Catarse se destacam nessa modalidade, com o Kickstarter financiando milhares de projetos em vários setores.

A convergência, cultura participativa, comunidades online, inteligência coletiva e copyleft são conceitos fundamentais na era digital que influenciaram a evolução do crowdsourcing e do crowdfunding. A convergência se refere à fusão de diversas mídias e tecnologias em plataformas únicas, tornando a experiência digital mais integrada. A cultura participativa destaca a mudança na maneira como as pessoas consomem e produzem conteúdo, promovendo a participação ativa.

As comunidades online possibilitaram que grupos com interesses comuns colaborassem e compartilhassem informações, formando nichos para apoiar projetos de crowdsourcing e crowdfunding. A inteligência coletiva demonstra como a colaboração de muitos pode levar a soluções mais inteligentes e criativas. O copyleft é uma abordagem de licenciamento que permite a redistribuição e modificação de conteúdo (AGUIAR, 2016; AVENI; PINTO, 2014; ALMEIDA, 2020).

O crowdfunding é uma aplicação específica do crowdsourcing, onde a multidão é mobilizada para financiar projetos ou empreendimentos por meio de contribuições financeiras em uma plataforma online. Existem diversos modelos de crowdfunding, incluindo recompensa, doação, empréstimo e investimento, cada um desses tipos tem suas próprias características e vantagens, dependendo das necessidades e objetivos do projeto.

A energia criativa da multidão desempenha um papel vital no financiamento coletivo, aproveitando a diversidade de conhecimento e experiência. O sucesso no crowdfunding é influenciado por vários fatores, como o tamanho da rede social do empreendedor, qualidade da campanha, destaque na plataforma e diferenças culturais e regulatórias locais. Os empreendedores podem seguir hipóteses de sucesso, como criar um bom vídeo, atualizar constantemente os apoiadores e conhecer o público-alvo.

No geral, esses conceitos e práticas demonstram como a Revolução Digital e a cultura participativa influenciaram o surgimento e o desenvolvimento do crowdsourcing e crowdfunding. Eles representam formas inovadoras de colaboração e financiamento de projetos na era digital, refletindo a mudança na maneira como as pessoas interagem e compartilham recursos (AGUIAR, 2016; AVENI; PINTO, 2014; ALMEIDA, 2020).

O crowdfunding é, de fato, um modelo de financiamento inovador no contexto empresarial, especialmente para startups e projetos de pequeno porte. Ele se baseia na ideia de angariar fundos de uma grande quantidade de pessoas, geralmente pela internet, para financiar um projeto ou empreendimento. Esse modelo de financiamento aproveita os princípios da Web 2.0, que enfatiza a colaboração, interatividade e participação do público.

No contexto empresarial, o crowdfunding pode ser uma alternativa valiosa às fontes tradicionais de financiamento, como empréstimos bancários ou investidores de capital de risco. Ele permite que empreendedores e startups acessem capital de uma base ampla de apoiadores, muitos dos quais podem estar interessados no produto ou serviço que está sendo desenvolvido. Isso não apenas ajuda a financiar o projeto, mas também envolve potenciais clientes desde o início, criando uma base sólida de apoio.

No Brasil, o Catarse é uma plataforma pioneira e reconhecida no campo do crowdfunding. Ela se destaca como a maior plataforma de financiamento coletivo no país e tem sido utilizada para financiar uma variedade de projetos, incluindo iniciativas culturais, sociais, tecnológicas e empreendimentos criativos.

O Catarse tem desempenhado um papel importante na promoção do crowdfunding no Brasil e na viabilização de projetos que, de outra forma, poderiam não encontrar financiamento facilmente por meio de fontes tradicionais.

A cocriação, conceituada por Prahalad e Ramaswamy, é uma estratégia inovadora que envolve a colaboração entre stakeholders, ou seja, pessoas e organizações que participam no desenvolvimento de um negócio, produto, conteúdo ou marketing.

Essa colaboração busca agregar valor por meio da inovação, e os participantes recebem benefícios em troca de sua contribuição. Esses benefícios podem incluir acesso a produtos customizados, aumento da eficiência global das trocas, redução de custos de negociação e competição, entre (AGUIAR, 2016; AVENI; PINTO, 2014; ALMEIDA, 2020).

Embora a ideia de cocriação não seja uma novidade, a abordagem de Prahalad e Ramaswamy trouxe destaque ao conceito e forneceu um quadro para a compreensão de como a colaboração entre empresas e seus stakeholders pode ser mutuamente benéfica.

A cocriação se tornou especialmente relevante com o advento da Web 2.0, que permitiu maior interação e participação dos usuários na internet. Plataformas como Facebook, Twitter e outras redes sociais possibilitaram a colaboração, a troca de informações e o envolvimento ativo dos usuários. Isso trouxe uma mudança significativa na forma como as empresas se relacionam com seus clientes e outros stakeholders.

O crowdfunding é uma forma de cocriação na qual os usuários investem recursos financeiros para financiar projetos de terceiros. Isso cria uma relação direta entre os financiadores e os empreendedores, permitindo que os projetos sejam viabilizados com base no apoio da comunidade.

No geral, esses conceitos e práticas demonstram como a Revolução Digital e a cultura participativa influenciaram o surgimento e o desenvolvimento do crowdsourcing e crowdfunding. Eles representam formas inovadoras de colaboração e financiamento de projetos na era digital, refletindo a mudança na maneira como as pessoas interagem e compartilham recursos (AGUIAR, 2016; AVENI; PINTO, 2014; ALMEIDA, 2020).

O crowdfunding é, de fato, um modelo de financiamento inovador no contexto empresarial, especialmente para startups e projetos de pequeno porte. Ele se baseia na ideia de angariar fundos de uma grande quantidade de pessoas, geralmente pela internet, para financiar um projeto ou empreendimento. Esse modelo de financiamento aproveita os princípios da Web 2.0, que enfatiza a colaboração, interatividade e participação do público.

1.2 Engenharia de Software e Análise de Requisitos

A engenharia de requisitos, é um processo abrangente que engloba todas as atividades necessárias para criar e manter o documento de requisitos do sistema,

representando a declaração oficial das exigências da equipe de desenvolvimento. A construção do produto de software deve ser alinhada com esses requisitos.

A compreensão insuficiente dos requisitos, como evidenciado por experiências reais e pesquisas frequentemente resulta em inúmeros problemas no desenvolvimento de software. Nesse contexto, um dos principais objetivos da engenharia de requisitos é a elaboração de requisitos facilmente compreensíveis, tanto para os solicitantes quanto para os desenvolvedores. Os requisitos desempenham três papéis fundamentais:

- Permitir que os desenvolvedores expliquem como o cliente deseja que o sistema funcione.
- Informar aos projetistas de software quais funções e características o sistema deve possuir.
- Orientar a equipe de teste na demonstração e convencimento ao cliente de que o sistema entregue atende às suas solicitações.

Requisitos do sistema como descrições de funções e restrições, destacando que a engenharia de requisitos abrange a descoberta, análise, documentação e verificação dessas funções e restrições.

A engenharia de software destaca a importância de identificar inconsistências precocemente para minimizar custos de correção. O custo de encontrar e corrigir um erro na fase de requisitos é significativamente menor (1) em comparação com fases subsequentes (5 na análise/design, 10 na codificação, 20 no teste unitário e mais de 200 após a entrega do sistema) (Sommerville, 2011; Leite; Breitman, 2003; Medeiros Junior, 2006; Boehm; Papaccion, 1988).

Portanto, investir em engenharia de requisitos é uma prática recomendada para prevenir problemas futuros ou alterações durante o desenvolvimento.

A presença onipresente do software é vital para o mundo moderno, onde sistemas computacionais controlam infraestruturas e serviços nacionais. A inclusão de computadores e softwares na maioria dos produtos elétricos é comum. Tanto a manufatura quanto a distribuição industrial, assim como o sistema financeiro, operam de forma totalmente informatizada.

Na indústria do entretenimento, que engloba música, jogos de computador, cinema e televisão, o software desempenha um papel crucial. Dessa forma, a engenharia de software emerge como essencial para o funcionamento eficaz de sociedades nacionais e internacionais.

Os sistemas de software, sendo abstratos e intangíveis, não são limitados por propriedades materiais ou leis físicas, o que simplifica a engenharia de software. No entanto, essa falta de restrições físicas pode levar a sistemas extremamente complexos, difíceis de compreender e dispendiosos de modificar.

A diversidade de sistemas de software, desde simples embutidos até complexos sistemas globais de informações, demanda abordagens específicas. A busca por notações, métodos ou técnicas universais na engenharia de software é inviável, pois diferentes tipos de software requerem enfoques distintos.

O desenvolvimento de um sistema de informações corporativo difere substancialmente da criação de um controlador para um instrumento científico. Apesar dessas variações, todos esses contextos demandam a aplicação de engenharia de software, adaptando as técnicas conforme necessário (Sommerville, 2011; Leite; Breitman, 2003; Medeiros Junior, 2006; Boehm; Papaccion, 1988).

2 Material e Métodos

Nesta seção serão abordados, a metodologia utilizada para o desenvolvimento do sistema, descrição do projeto e as ferramentas e conceitos utilizados para o desenvolvimento do projeto.

Para desenvolvimento do trabalho, foram realizados levantamento e análise bibliográfica de teses e artigos científicos na Scientific Electronic Library Online (SciELO) e Biblioteca Digital de Teses e Dissertações (BDTD), além da busca em blogs que tenha assunto semelhantes.

2.1 Requisitos Funcionais e Não Funcionais

A partir de uma análise foi possível definir os requisitos funcionais e não funcionais para o desenvolvimento da plataforma, que são descritos nas tabelas abaixo. “A busca dos requisitos é um fator crítico para o desenvolvimento de sistemas de software, uma vez que a partir da interpretação desses requisitos por parte dos desenvolvedores é possível atingir as metas e necessidades a serem alcançadas com o software” (Bigonha, 2015).

Requisitos Funcionais	
RF001	Permitir que o usuário se cadastre na plataforma.
RF002	Permitir que o usuário faça o cadastro de projetos na plataforma
RF003	Permitir que o usuário faça login na plataforma
RF004	Permitir que o usuário veja todos os que projetos que foram cadastrados na plataforma.
RF005	Permitir que o usuário veja somente os seus projetos cadastrados na plataforma

Requisitos Não-Funcionais	
RNF001	Garantir que a plataforma seja compatível com diferentes dispositivos moveis e navegadores web.
RNF002	Desenvolver uma interface de usuário amigável e intuitiva para o usuário.
RNF003	Garantir que o sistema não deixe o usuário cadastrar um projeto sem estar logado.
RNF004	Garantir que o usuário não possa fazer alterações do projeto de outra pessoa.

Após a especificação dos requisitos, dá-se início à fase de projeto e desenvolvimento de software. Nesse estágio, ocorre a modelagem do sistema e do banco de dados, além da escolha das ferramentas a serem empregadas no processo de desenvolvimento. A partir dos requisitos delineados, elabora-se um fluxograma que descreve as ações que o sistema executará para cumprir esses requisitos. Para documentar a estrutura do sistema, são empregados diagramas da linguagem UML.

2.2 Diagrama de Casos de Uso

O diagrama de Casos de Uso figura como um dos principais elementos na UML, desempenhando um papel central ao proporcionar uma visão abrangente das funcionalidades que o sistema deve disponibilizar aos usuários.

Modelos de caso de uso (use cases) são modelos passíveis de compreensão tanto por Desenvolvedores – analistas, projetistas, programadores e testadores – como pela comunidade Usuária – clientes e usuários. Como o próprio nome sugere, um caso de uso é uma maneira de Usar o sistema. Usuários interagem com o sistema, interagindo com seus casos de uso. Tomados em conjunto, os casos de uso de um sistema representam a sua funcionalidade. Casos de uso São, portanto, os “itens” que o desenvolvedor negocia com seus clientes (Monalessa, 2018, p. 64).

Abaixo, o Caso de Uso apresentado na Figura 1 tem como objetivo descrever o sequenciamento das principais funcionalidades que serão fornecidas pelo sistema, seguido pela documentação individual de cada caso de uso.

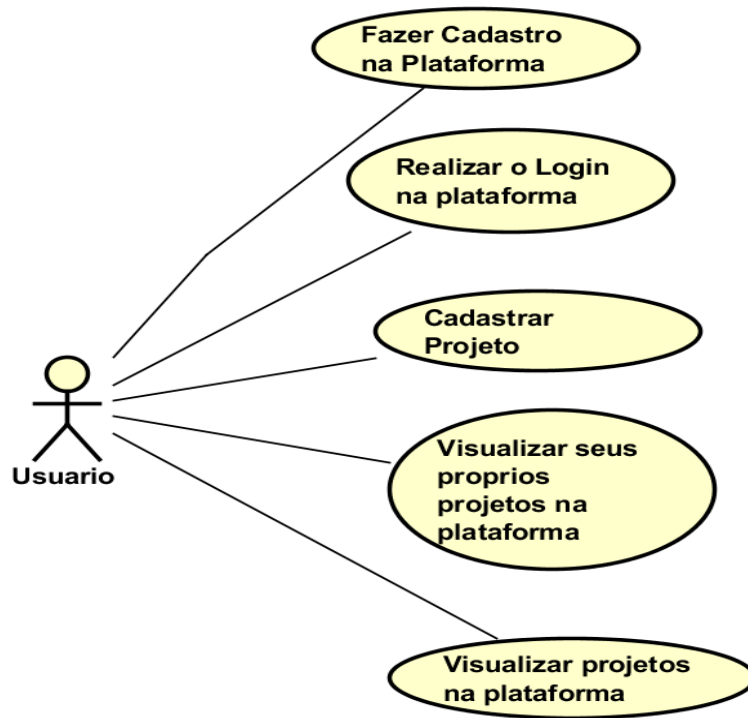


Figura 1 - Diagrama de Casos de Uso

Fazer Cadastro na Plataforma	
Pré-Condições	<ul style="list-style-type: none"> • Não ter cadastro na plataforma.
Atores	Usuário, Sistema
Fluxo Principal	<ol style="list-style-type: none"> 1. O usuário deve informar o nome. 2. O usuário deve informar o nome de usuário. 3. O usuário deve informar o CPF ou CNPJ. 4. O usuário deve informar o Email. 5. O usuário deve informar o Telefone. 6. O sistema verifica se existe alguma conta o e-mail fornecido. 7. Realizar cadastro.
Pós-Condições	<ul style="list-style-type: none"> • O sistema registra as informações fornecidas pelo usuário, criando uma nova conta de usuário na plataforma.

Realizar o Login na Plataforma	
Pré-Condições	<ul style="list-style-type: none"> • O usuário possui uma conta cadastrada na plataforma
Atores	Usuário, Sistema
Fluxo Principal	<ol style="list-style-type: none"> 1. O usuário informa o nome de usuário 2. O usuário informa a senha. 3. O sistema verifica as informações fornecidas 4. Se as informações forem validas realiza o login

Cadastrar Projeto	
Pré-Condições	<ul style="list-style-type: none"> ● O usuário deve estar logado na plataforma
Atores	Usuário, Sistema
Fluxo Principal	<ol style="list-style-type: none"> 5. O usuário informa o nome do projeto. 6. O usuário informa a descrição do projeto 7. O usuário escolhe o tipo de contribuição 8. Realiza cadastro do projeto.
Pós-Condições	<ul style="list-style-type: none"> ● O sistema armazena as informações do projeto na base de dados, associando-as à conta do usuário. O projeto cadastrado agora está disponível na plataforma para visualização e interação por outros usuários.

Visualizar seus próprios projetos na plataforma	
Pré-Condições	<ul style="list-style-type: none"> ● O usuário deve ter feito o cadastro de um ou mais projetos
Atores	Usuário, Sistema
Fluxo Principal	<ol style="list-style-type: none"> 1. O usuário clica na opção "Meus Projetos". 2. O sistema exibe uma lista dos projetos cadastrados pelo usuário.

Visualizar projetos na plataforma	
Pré-Condições	<ul style="list-style-type: none"> ● O usuário estar com a plataforma aberta
Atores	Usuário, Sistema
Fluxo Principal	<ol style="list-style-type: none"> 1. O sistema exibe os projetos de todos os usuários na tela principal

2.3 Diagrama de Classes

O Diagrama de Classes representa um conjunto de classes e seus respectivos relacionamentos. Esses diagramas oferecem uma perspectiva estática da configuração de um sistema e, conseqüentemente, são empregados na modelagem conceitual da sua estrutura (Monalessa, 2018).

O objetivo principal do diagrama é apoiar na delimitação da organização do sistema, por meio da especificação das classes, bem como de seus atributos e métodos. A ilustração a seguir exemplifica a configuração do projeto do sistema utilizando o diagrama de classes.

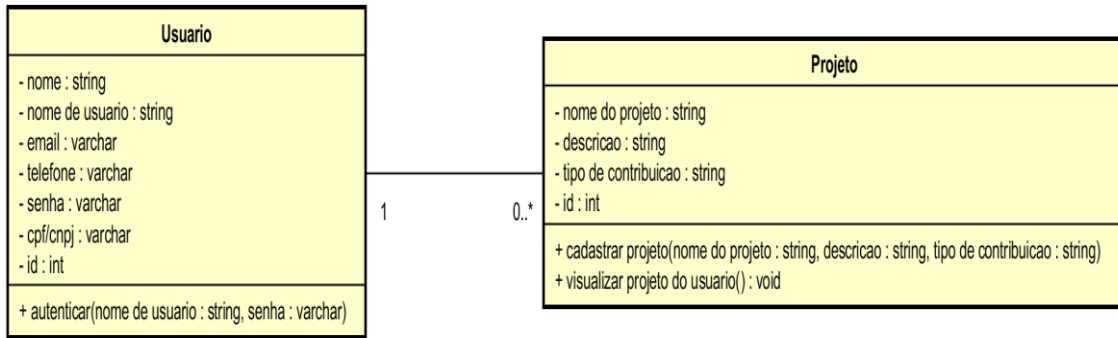


Figura 2 - Diagrama de Classes

2.4 Modelo Entidade Relacionamento (MER)

O Modelo Entidade-Relacionamento (MER) é um modelo conceitual empregado na Engenharia de Software para ilustrar as entidades da aplicação, seus atributos e os vínculos entre elas, com o propósito de documentar a configuração do banco de dados de um sistema.

Para visualizar de forma gráfica a estrutura do banco, o MER faz uso do Diagrama de Entidade-Relacionamento (DER), que incorpora figuras e diretrizes específicas. A representação visual da estrutura do banco de dados do software em desenvolvimento é apresentada na figura abaixo.

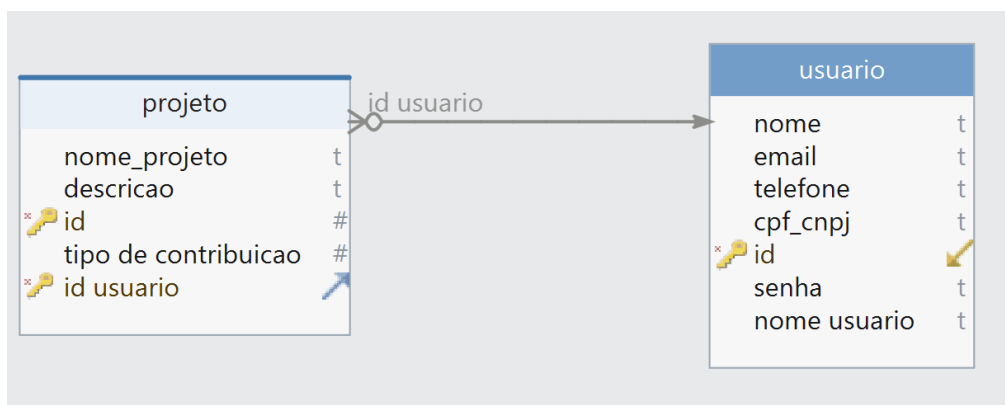


Figura 3 - Diagrama Entidade Relacionamento

2.5 Sistema de Gerenciamento de Banco de Dados (SGBD)

O Sistema de Gerenciamento de Banco de Dados, conforme sugere seu próprio nome, é a aplicação encarregada de supervisionar todas as operações relacionadas à entrada, saída, consulta, controle de acesso e segurança de um banco de dados em um software. A maioria dos SGBDs disponibiliza ferramentas gráficas que simplificam a administração do banco, otimizando o gerenciamento.

Para a criação e administração do banco de dados do sistema financeiro, optaremos pelo MySQL em conjunto com o XAMPP, que incorpora o PHPMyAdmin, oferecendo uma solução integrada e de fácil manipulação.

2.6 PHP

O PHP (Hypertext Preprocessor) é uma linguagem de programação de script amplamente utilizada para o desenvolvimento web. O PHP pode ser empregado para criar páginas dinâmicas, interagir com bancos de dados, processar formulários e executar diversas tarefas do lado do servidor. Ele é frequentemente combinado com HTML para criar aplicativos web dinâmicos e interativos. O PHP é conhecido por sua simplicidade, versatilidade e ampla comunidade de desenvolvedores, tornando-o uma escolha comum para projetos web.

2.7 HTML

HTML (Hypertext Markup Language) é a linguagem padrão para a criação e a estruturação de páginas web. Ela fornece os elementos básicos que compõem uma página, como títulos, parágrafos, listas, links, imagens e outros elementos de marcação. No contexto do seu projeto, você usará HTML para criar a estrutura das páginas web, definindo o conteúdo e a aparência dos diferentes elementos.

O HTML é essencial para construir a interface do usuário e apresentar informações de maneira organizada e compreensível. Ele é frequentemente combinado com outras tecnologias, como CSS (Cascading Style Sheets) para estilização e PHP para tornar as páginas dinâmicas.

2.8 CSS

O CSS é uma linguagem de estilo utilizada para controlar a apresentação visual de documentos HTML. Em projetos web, o CSS desempenha um papel fundamental ao permitir a definição da aparência visual dos elementos HTML.

Com o CSS, é possível estilizar páginas web de maneira abrangente, determinando aspectos como cores, fontes, tamanhos de texto e outros estilos visuais para diversos elementos, incluindo cabeçalhos, parágrafos, listas e links. Além disso, o CSS oferece controle sobre o layout da página, permitindo posicionar elementos, definir margens, preenchimentos e especificar tamanhos.

A responsividade também é uma característica importante do CSS, possibilitando a adaptação da página a diferentes tamanhos de tela e dispositivos. O uso de técnicas como *media queries* permite criar designs responsivos que proporcionam uma experiência consistente aos usuários, independentemente do dispositivo que estão utilizando.

Outro aspecto relevante do CSS é a capacidade de adicionar animações e transições, contribuindo para uma experiência do usuário mais interativa e visualmente atraente. Esses recursos permitem incorporar efeitos visuais, melhorando a dinâmica da página.

Além disso, o CSS desempenha um papel crucial na garantia da compatibilidade entre navegadores, assegurando que a aparência da página seja consistente em diferentes plataformas. Ao trabalhar em conjunto com HTML, o CSS completa a estrutura necessária para criar interfaces de usuário atraentes e funcionais em projetos web.

2.9 BOOTSTRAP

Bootstrap utiliza uma combinação de CSS e JavaScript para estilizar elementos HTML, proporcionando uma gama abrangente de funcionalidades. Vai além da estética, oferecendo interatividade à página por meio de diversos componentes, como menus de navegação, controles de paginação, barras de progresso e mais.

Um dos principais objetivos do Bootstrap é facilitar a construção de sites responsivos para dispositivos móveis. Isso implica que as páginas desenvolvidas com esse framework são adaptáveis a diferentes plataformas, proporcionando uma experiência simplificada e bem organizada em desktops, tablets e smartphones (Souza, 2019).

2.10 XAMPP

O XAMPP é um pacote de software gratuito e de código aberto que fornece um ambiente de servidor web local. A sigla XAMPP refere-se a "X" (indicando que é cross-platform), "Apache" (o servidor web), "MySQL" (o sistema de gerenciamento de banco de dados), "PHP" (linguagem de script) e "Perl" (também uma linguagem de script).

Ao utiliza-lo se tem um servidor web local na máquina, o que facilita o desenvolvimento e teste de aplicações web sem a necessidade de um servidor remoto. Ele inclui os componentes essenciais para o desenvolvimento web, como Apache, MySQL (ou MariaDB), PHP, e Perl.

O XAMPP oferece um ambiente de desenvolvimento completo que permite criar, testar e depurar suas aplicações antes de implantá-las em um ambiente de produção. Isso é especialmente útil para desenvolvedores web, pois agiliza o ciclo de desenvolvimento e proporciona maior controle sobre o ambiente de teste.

3 Resultados e Discussões

Para o desenvolvimento do sistema buscou-se seguir as etapas do processo de software descritas na revisão bibliográfica, como o objetivo do projeto é criar uma plataforma que possa ser criado um projeto, e possa ser feito um financiamento do projeto ou ter desenvolvedores participando do projeto focamos em desenvolver somente algumas partes essenciais para ilustrar como pode ser essa plataforma, sem estar implementando todas as coisas que seriam mais complexas em um curto prazo de desenvolvimento. Após isso foram definidos os requisitos do sistema, seguidos da definição e documentação dos casos de uso, descritos na seção de materiais e métodos.

Seguindo o processo de desenvolvimento, se baseando na documentação dos casos de uso, foi realizada a modelagem do banco de dados, somente após a modelagem do banco, foi criado o diagrama de classes da estrutura do sistema e iniciado o desenvolvimento. Como é mostrado no diagrama entidade relacionamento ou modelagem do banco, o usuário e o projeto estão relacionados, por isso é necessário que o desenvolvimento comece a partir do cadastro e login do usuário.

Na Figura 4 é mostrado a tela inicial, como faremos primeiro o cadastro e login do Usuário somente criamos o front para já ter os botões que precisaremos futuramente, deixando somente necessário programar cada um eventualmente.



Figura 4 - Tela inicial

Na Figura 5 temos o cadastro do usuário, sendo necessário fornecer as informações de Nome, Nome de Usuário, CPF ou CNPJ, e-mail, Telefone e Senha. Também temos o botão home que volta para a tela inicial e o botão login caso o usuário já tenha se cadastrado e queira ir para a tela de login.

Ao realizar o cadastro os dados fornecidos aqui vão para o banco de dados e ficam armazenados para uso posterior.

Figura 5 – Tela cadastro de Usuário

Na Figura 6 temos a tela de login, como já foi feito a tela de cadastro agora é somente necessário inserir os dados e fazer login, aqui temos os campos Email e Senha, caso o usuário não tenha se cadastrado ele pode ir para lá diretamente por essa tela apertando o botão, e novamente temos o botão de voltar para a tela inicial.

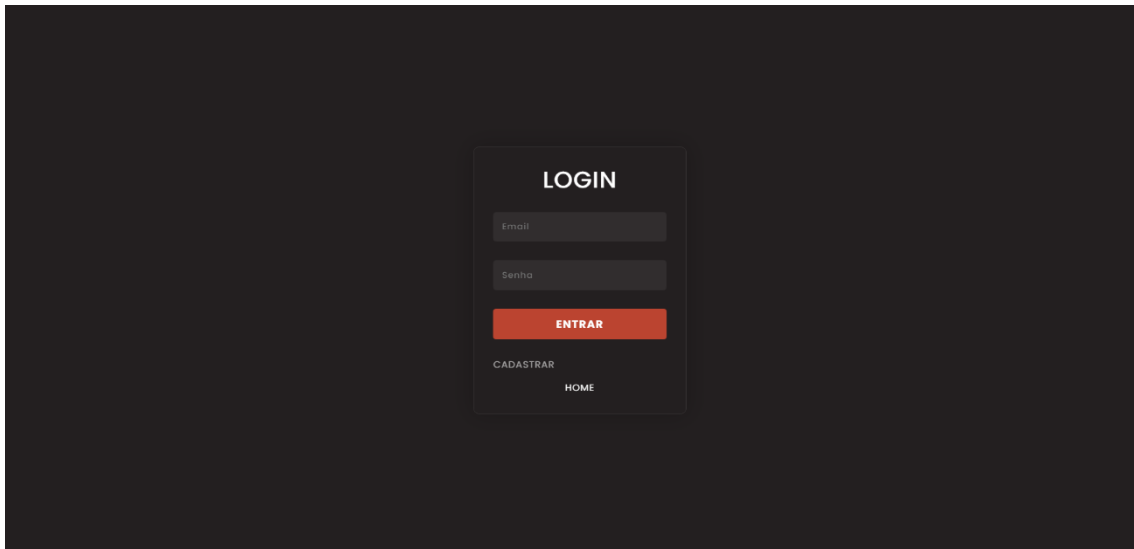


Figura 6 - Tela de login

Na Figura 7 ao apertar o botão entrar e executar o login é feita uma busca no banco de dados para saber se os dados fornecidos estão corretos e caso estejam então o usuário é logado, caso o usuário digite alguma informação errada ou os dados que ele forneceu não estejam na base cadastrados aparecera uma mensagem informando que os dados estão inválidos.

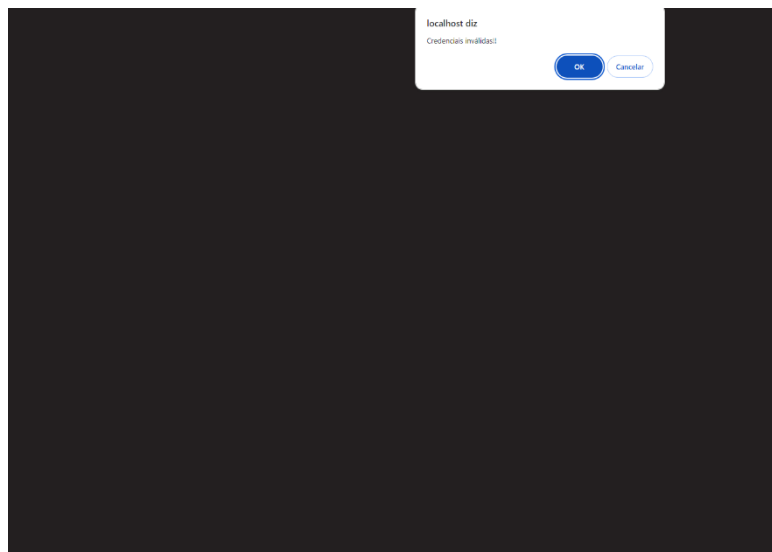


Figura 7 - Tela de erro no login

A Figura 8 mostra onde o usuário vai após o login, queríamos que após o usuário fazer o login, ele fosse direcionado para os seus projetos cadastrados, e como é demonstrado na modelagem de banco de dados e nos requisitos o usuário e o projeto estão relacionados com isso quando fizemos o banco, colocamos o id do usuário como uma FK (Chave Estrangeira) para o projeto, então quando o usuário cadastrar um projeto, será

armazenado junto o id daquele usuário que podemos usar aqui na tela de meus projetos para mostrar somente os projetos o usuário cadastrou.

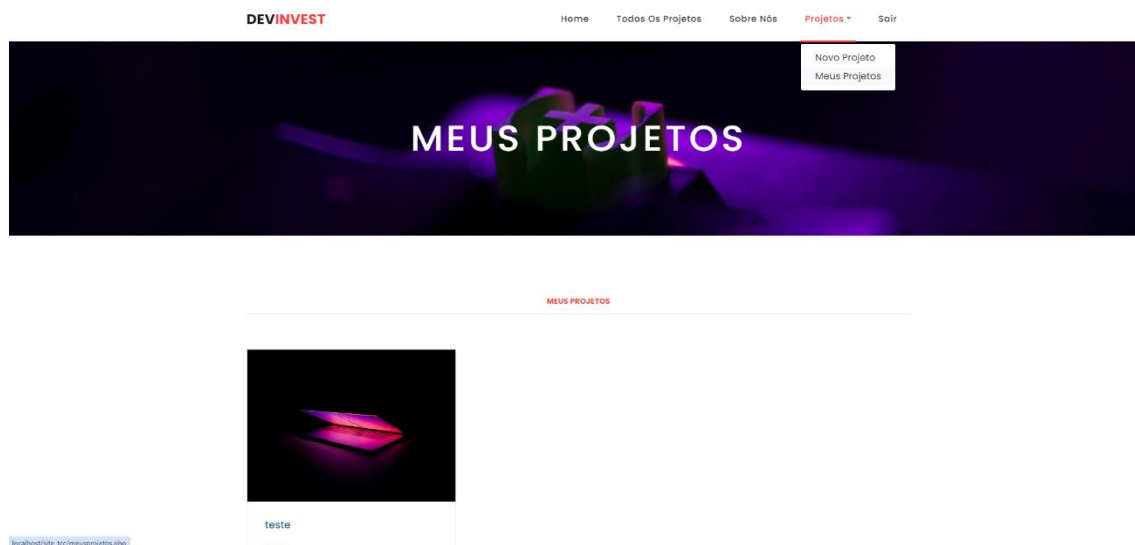


Figura 8 – Tela de projetos do Usuário

Na Figura 9 temos o cadastrar projeto, possuímos o Nome do projeto, o campo de descrição do projeto e também temos para escolher o tipo de contribuição, Desenvolvimento, Investimento e Ambos.

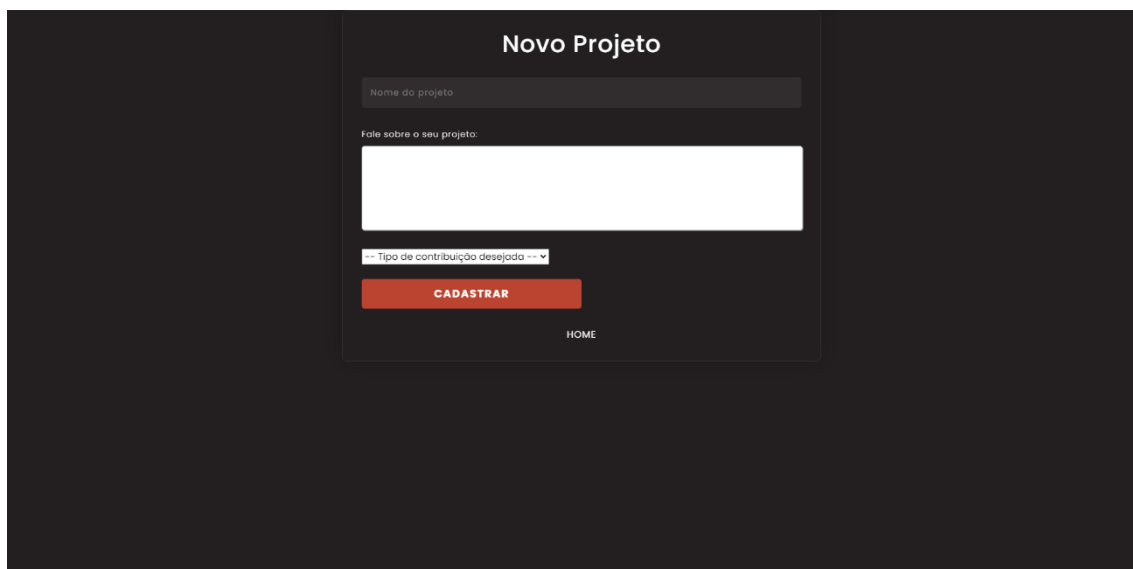


Figura 9 – Tela cadastro de projeto

Na Figura 10 temos a tela Todos os Projetos, como já temos o cadastro do usuário, ele pode fazer o login, pode cadastrar um projeto e todos os projetos que ele cadastrou vão aparecer para ele em uma tela, aqui temos os projetos de todos os usuários, enquanto

na tela de meus projetos aparecia somente os projetos do usuário que vinculamos ao seu id, aqui colocamos para exibir todos os projetos cadastrados na plataforma.

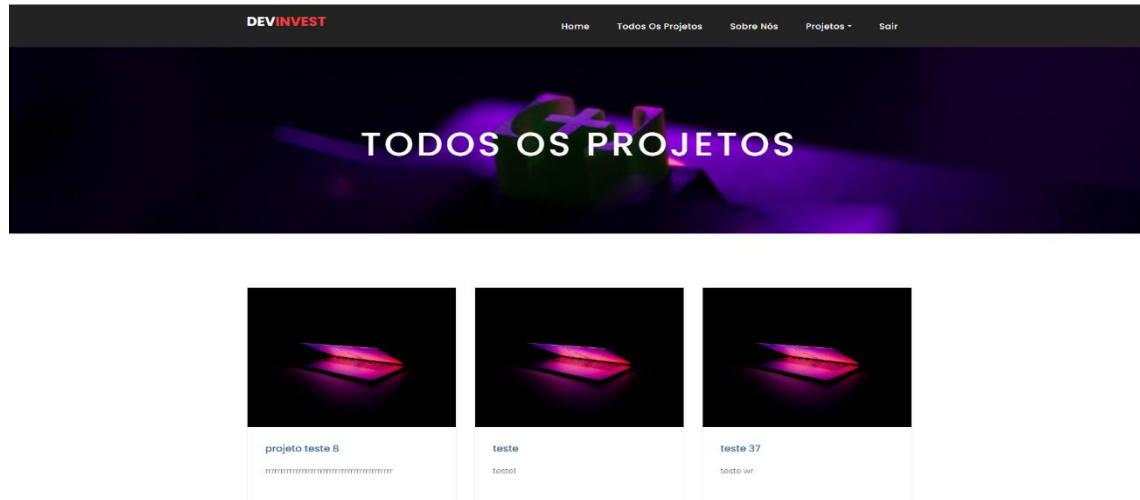


Figura 10 - Tela de todos os projetos

Como esse é um projeto que pode potencialmente ser usado comercialmente depois de futuras implementações, e evolução, decidimos colocar a tela que fala um pouco sobre o site, como mostra na Figura 11 tendo os tópicos, Quem somos, Nossa visão e missão, e Junte-se (sobre os usuários participarem seja investindo nos projetos, como desenvolvedor e também criando um projeto), esse tipo de tela passa confiança, credibilidade, informações sobre qual a linha de pensamentos das pessoas que estão por trás daquela plataforma.

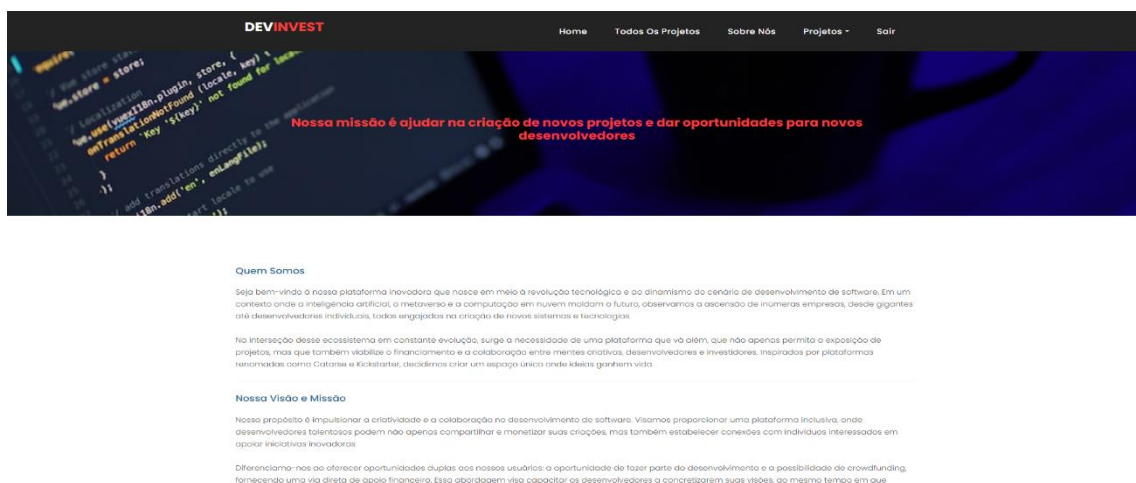


Figura 11 – Tela Sobre a Plataforma

Na figura 12 é mostrado a funcionalidade da plataforma de restringir algumas ações dependendo se o usuário está logado ou não, caso o usuário não esteja logado irá aparecer o botão login e novo usuário, e como queremos que o usuário só faça cadastro

de projeto quando estiver logado e para ver seus projetos na tela Meus Projetos você também precisa estar logado, esses botões só irão aparecer ao fazer o login e junto dele o botão Sair irá substituir o botão login

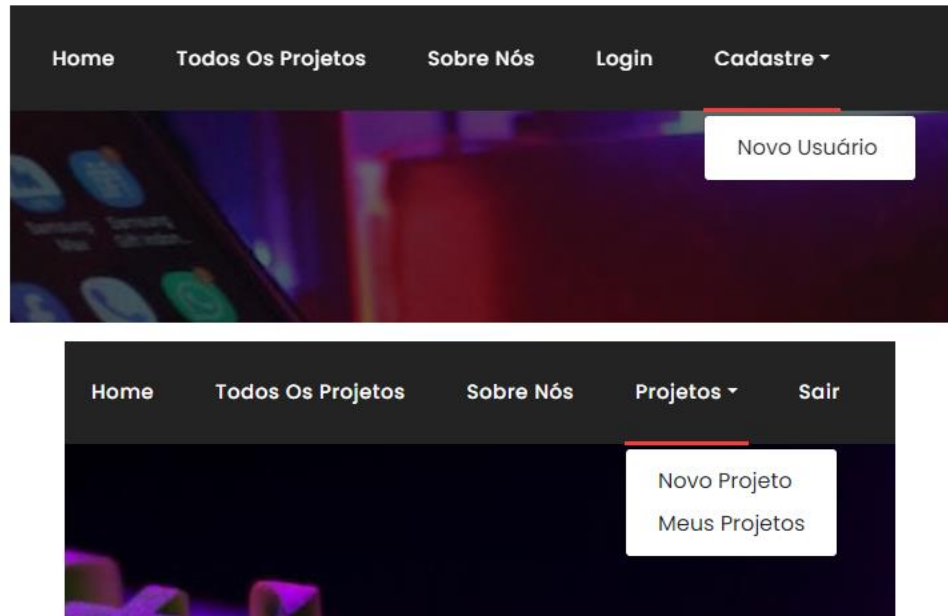


Figura 12 – Print sobre as diferenças de estar ou não logado

Considerações Finais

O objetivo desse projeto foi de desenvolver uma plataforma de financiamento coletivo de projetos e de desenvolvedores interessados em estar participando dos projetos, voltada para projetos da área de software.

Para o desenvolvimento foi utilizado conhecimentos de engenharia de software, banco de dados, ferramentas como bootstrap, estudo de outras plataformas semelhantes que possui o financiamento coletivo.

Acreditamos que em partes o nosso objetivo foi alcançado, desenvolvemos uma plataforma base que possui as principais características para o que propusemos que fosse uma eventual plataforma final, você pode cadastrar um usuário, fazer login, cadastrar um projeto, e visualizar todos os projetos da plataforma e também somente o do usuário logado, também tem algumas funcionalidades básicas para segurança de o cadastro de projeto ser somente para usuários logados.

A partir desse projeto inicial julgamos que é possível chegar no que pretendemos como um projeto final e pronto para uso na rede global, porém como esse é um trabalho mais orientado para a pesquisa e o tempo e recurso em mãos nosso é limitado, acreditamos que conseguimos atingir um bom resultado.

Muitas outras funcionalidades, como a que julgamos ser a mais importante que é o financiamento do projeto, não é possível colocarmos em um projeto inicial como esse, porém é sim possível implementar já que a sempre espaço para ir evoluindo o sistema implementando novas funções.

Para chegar no que pretendemos como um projeto funcional para uso no foco que ele é, futuras implementações seriam, o já falado financiamento do projeto, dar a opção de em vez do Usuário financiar ele estar ingressando no time de desenvolvimento do projeto, editar o projeto, seja adicionando informações ou até excluindo ele, focar na segurança do site, como ele envolve informações pessoas como de pagamentos é de extrema importância, e outra implementação seria ter a meta de financiamento e ela sendo atingida os financiadores ganhariam recompensas.

Referências

AGUIAR, Carlos Eduardo Magalhães Vieira. **Crowdfunding no Brasil: um estudo sobre a plataforma Catarse**. São Carlos: Ufscar, 2016

ALMEIDA, Leonardo Mancini. **Crowdfunding no Brasil: determinantes do apoio em projetos do Catarse**. Rio de Janeiro: Biblioteca Digital de Teses e Dissertações, 2020

AVENI, Alessandro; PINTO, Luiz Fernando Silva. Crowdfunding: o modelo Canvas do site catarse. **Revista Gestão & Saúde**, v. 4, n. 2, p. 2-10, 2014

BARCELLOS, Monalessa Perini. **Engenharia de Software**. 2018. Disponível em: <http://www.inf.ufes.br/~monalessa/ensino/engenharia-de-software-2020/>. Acesso em: 12 dez. 2023.

BIGONHA, Renato Dias. **Requisitos de software: a vagueza e ambiguidade em um requisito de software**. São Paulo: Repositório PUC, 2015.

BOEHM, Barry Willia; PAPACCION, Philip Nunzio. Understanding and Controlling Software Costs. **Revista IEEE Transaction on Software Engineering**, v. 14, n. 10, p. 1462-1477, oct. 1988.

LEITE, Julio Cesar Sampaio; BREITMAN, Karin Koogan. **Ontology as a Requirements Engineering Product**. 2003. Disponível em: <https://ieeexplore.ieee.org/document/1232775>. Acesso em: 12 dez. 2023.

MEDEIROS JÚNIOR, Raul de Abreu. **Uma Ontologia para Engenharia de Requisitos de Software**. 9. ed. Fortaleza: Pearson Education, 2006.

SOMMERVILLE, Ian. **Engenharia de Software**. São Paulo: Pearson Prentice Hall, 2011.

SOUZA, Ivan. **Bootstrap**: saiba neste guia para iniciantes o que é, por que e como usá-lo. 2019. Disponível em: <https://rockcontent.com/br/blog/bootstrap/>. Acesso em: 16 out. 2023.