INCUBEPRO: AMBIENTE INTEGRADO PARA APOIO, GESTÃO E FINANCIAMENTO DE PROJETOS DE SOFTWARE

Ygor Gonzaga Josué; Mateus de Sousa Valente[n]
Universidade de Uberaba
ygorjosue7@gmail.com; mateus.sousa.valente@gmail.com

Resumo

A IncubePro é uma plataforma desenvolvida para impulsionar o financiamento e a evolução de projetos de software, com foco no fortalecimento da inovação tecnológica no Brasil. Seu principal objetivo é conectar desenvolvedores, investidores e mentores em um ambiente digital unificado, facilitando a publicação, gestão e apoio a projetos promissores. A ideia surge diante de um cenário em que muitas startups brasileiras não recebem apoio financeiro, e ideias inovadoras são frequentemente abandonadas por falta de recursos. Nesse contexto, a IncubePro apresenta-se como uma solução estratégica, utilizando o crowdfunding como ferramenta central. Essa abordagem permite não apenas a captação de recursos, mas também a validação pública das ideias, fortalecendo sua credibilidade. Ao facilitar o encontro centralizado com possíveis financiadores, a plataforma visa tornar os projetos mais atrativos, transparentes e com maior potencial de sucesso. Essa conexão direta amplia as oportunidades de investimento, estimula a colaboração e aumenta a confiança dos apoiadores. Com essa proposta, a IncubePro contribui para a retenção de talentos e o fortalecimento do ecossistema de inovação, promovendo impactos positivos no desenvolvimento econômico e social do país. Trata-se de uma iniciativa que busca transformar boas ideias em soluções reais por meio do apoio coletivo.

Palavras-chave: *inovação tecnológica; financiamento coletivo, startups, desenvolvimento de software*

1 Introdução

A Tecnologia da Informação (TI) tem desempenhado um papel central na transformação digital de empresas, governos e da sociedade como um todo. Soluções tecnológicas baseadas em inteligência artificial, computação em nuvem, Internet das Coisas (IoT) e sistemas distribuídos têm proporcionado ganhos expressivos em produtividade, escalabilidade e inovação em setores como saúde, educação, logística e finanças. Contudo, apesar do mencionado avanço global, o cenário brasileiro ainda enfrenta desafios significativos no que se refere ao incentivo à inovação e suporte à criação de novas propostas digitais atuantes na resolução de problemas na sociedade.

Diversos estudos apontam que a escassez de investimento e a falta de apoio técnico qualificado comprometem o desenvolvimento de startups e projetos de software no país. Segundo dados da Abstartups (2024), mais de 60% das startups brasileiras nunca receberam qualquer tipo de investimento externo, o que compromete não apenas o crescimento dessas empresas, mas também a capacidade do país de gerar soluções tecnológicas inovadoras. Como reforçam Alves (2013) e Cardoso & Zilber (2017), startups

são iniciativas empreendedoras baseadas em tecnologia e diferenciação, porém exigem capital significativo e apoio técnico desde os estágios iniciais para que possam se consolidar em um mercado altamente competitivo.

Nesse contexto, o financiamento coletivo (crowdfunding) emerge como uma alternativa viável e estratégica. Trata-se de um modelo de captação de recursos via internet, no qual pessoas físicas ou jurídicas contribuem financeiramente para a execução de projetos em que acreditam (AMEDOMAR, 2015). Além do aspecto financeiro, Bastos (2021) destaca que o crowdfunding também contribui para a validação pública das ideias, criando um ambiente de engajamento que favorece a confiança de investidores e parceiros. Contudo, a maioria das plataformas atuais ainda carece de recursos que ofereçam suporte técnico estruturado, o que limita sua efetividade no desenvolvimento de soluções mais robustas.

Diante desse panorama, a proposta deste trabalho se justifica por seu potencial de impactar positivamente o ecossistema tecnológico brasileiro, ao integrar, em uma única solução, mecanismos de financiamento coletivo com funcionalidades voltadas ao suporte técnico e mentoria especializada. A criação de um ambiente digital colaborativo não apenas viabiliza o desenvolvimento de projetos de software, mas também contribui para a retenção de talentos no país e a democratização do acesso a recursos de inovação. Conforme relatado pelo portal TI Inside (2023), 80% das empresas brasileiras planejam ampliar seus investimentos em TI nos próximos anos, o que evidencia uma janela de oportunidade para soluções que fortaleçam a conexão entre empreendedores, investidores e especialistas.

Por fim, este trabalho tem como objetivo o desenvolvimento da IncubePro, uma solução que conecta desenvolvedores de software, investidores e técnicos que poderão guiar as ideais. A proposta busca integrar recursos de financiamento coletivo com suporte especializado, proporcionando um ambiente que estimula a criação, validação e execução de projetos inovadores. Ao promover o engajamento entre diferentes atores do ecossistema tecnológico, a plataforma pretende reduzir barreiras técnicas e financeiras, fomentar a retenção de talentos e contribuir para a democratização do acesso à transformação digital no Brasil.

2 Materiais e Métodos

2.1 Estrutura Geral e Ambiente de Desenvolvimento

A aplicação desenvolvida consiste em um sistema web composto por uma API RESTful implementada com a linguagem JavaScript, utilizando o ambiente de execução Node.js e o framework Express.js, em conjunto com o banco de dados não relacional MongoDB. O backend da aplicação utiliza o Node.js devido ao seu modelo de I/O assíncrono e não bloqueante, o que o torna ideal para aplicações escaláveis e orientadas a eventos (NODE.JS FOUNDATION, 2025, tradução do autor). A arquitetura adotada seguiu o padrão MVC (Model-View-Controller), promovendo a separação de responsabilidades e a organização lógica do sistema em camadas.
O ambiente de desenvolvimento foi estabelecido no editor Visual Studio Code, com controle de versão via Git. A gestão de pacotes e dependências foi realizada com o Node Package Manager (npm). O projeto contou com a utilização de variáveis de ambiente, armazenadas no arquivo .env, para garantir segurança e flexibilidade na configuração do sistema.

2.2 Inicialização e Organização do Projeto

A estrutura inicial foi criada com o comando npm init -y, que gerou o arquivo package.json. Em seguida, foram instaladas as dependências principais do backend, como express, mongoose, dotenv, cookie-parser, cors, bcryptjs, jsonwebtoken, express-validator e ejs. Para fins de desenvolvimento, a ferramenta nodemon foi utilizada como dependência auxiliar.

A pasta backend/ foi organizada com os seguintes diretórios:

- config/: arquivos de configuração, como conexão com o banco de dados;
- controllers/: lógica de controle de requisições e respostas;
- middleware/: funções intermediárias de autenticação e validação;
- models/: definição dos esquemas de dados com Mongoose;
- routes/: roteamento das requisições para os respectivos controladores;
- utils/: funções auxiliares padronizadas;
- views/ e public/: templates EJS e arquivos estáticos, respectivamente.

2.3 Configuração do Servidor e Banco de Dados

O arquivo principal server.js configurou o servidor Express, integrando middlewares como express.json(), express.urlencoded(), cors e cookie-parser. O framework Express foi adotado para o gerenciamento de rotas e middlewares, por sua flexibilidade e conjunto robusto de funcionalidades para desenvolvimento web (OPENJS FOUNDATION, 2025, tradução do autor). A conexão com o MongoDB foi estabelecida por meio do arquivo config/db.js, utilizando a biblioteca Mongoose e a variável de ambiente MONGO_URI.

2.4 Modelagem dos Dados

Dois modelos principais foram definidos:

- Usuário (User.js) sendo os campos: nome, email, senha, role (desenvoldor, investidor), data de criação, bio, skills, avatar, token e confirmed. Incluiu middleware pre('save') para hashing de senhas com bcryptjs e método de instância para comparação segura de senhas.

- Projeto (Project.js) sendo os campos: nome, descrição, status (Conceito, em andamento, beta, etc), categoria (Web, Mobile, Jogo, etc), meta de financiamento, criador, recompensas, data de criação, protótipo, com referências ao modelo de usuário e controle de criação e modificação via timestamps.

2.5 Lógica de Negócio e Controladores

As funcionalidades da aplicação foram implementadas nos controladores localizados em controllers/, onde cada arquivo trata de um domínio específico. As operações foram desenvolvidas de forma assíncrona, com tratamento de exceções e padronização de respostas por meio de utilitários próprios.

Os principais controladores implementados foram:

- authController: autenticação, registro, confirmação de conta, redefinição de senha e logout;

- userController: manipulação de dados do perfil do usuário;
- projectController: criação, leitura, atualização e remoção de projetos, além da gestão de colaboradores;
- sessionController: verificação de sessão ativa.

2.6 Middlewares e Validações

Dois tipos principais de middlewares foram empregados:

- Autenticação (auth.js): validação de tokens JWT presentes em cookies HTTPOnly, garantindo acesso apenas a usuários autenticados;
- Validação de dados (validation.js): uso da biblioteca express-validator para verificar campos de entrada, com validações específicas para cada operação (ex.: validateLogin, validateProject, validatePasswordUpdate).

Esses middlewares foram aplicados em rotas estratégicas para garantir a integridade e segurança dos dados.

2.7 Rotas da API

As rotas foram distribuídas conforme a responsabilidade funcional:

- /api/auth: operações de autenticação e recuperação de senha;
- /api/users: gerenciamento de perfil e senha do usuário autenticado;
- /api/projects: operações CRUD de projetos e gerenciamento de colaboradores;
- /api/sessions: validação de sessão do usuário logado;
- /: renderização de páginas com templates EJS, por meio das rotas definidas em pages.js.

A associação entre rotas, middlewares e controladores foi feita por meio do express.Router().

2.8 Desenvolvimento da Interface (Frontend)

A camada de apresentação da aplicação foi desenvolvida com foco em simplicidade, baixo acoplamento e rapidez de implementação, características essenciais para protótipos funcionais. A escolha da stack evitou frameworks de SPA (Single Page Application) mais robustos, como React ou Vue.js, em favor de ferramentas que favorecem o desenvolvimento ágil, com menor complexidade e curva de aprendizado reduzida.

Tecnologias Utilizadas:

- A renderização das páginas foi feita com EJS (Embedded JavaScript Templates), uma linguagem de *template* que permite gerar HTML dinâmico com JavaScript puro (EJS, 2025, tradução do autor). A linguagem permite injeção de dados no HTML sem a necessidade de lógica client-side complexa;
- Para estilização da interface, utilizou-se o Tailwind CSS, por seu modelo utilitário que permite construir interfaces personalizadas com agilidade (TAILWIND LABS, 2025, tradução do autor). O framework possibilita estilização direta nas classes HTML, com foco em responsividade, consistência e produtividade;

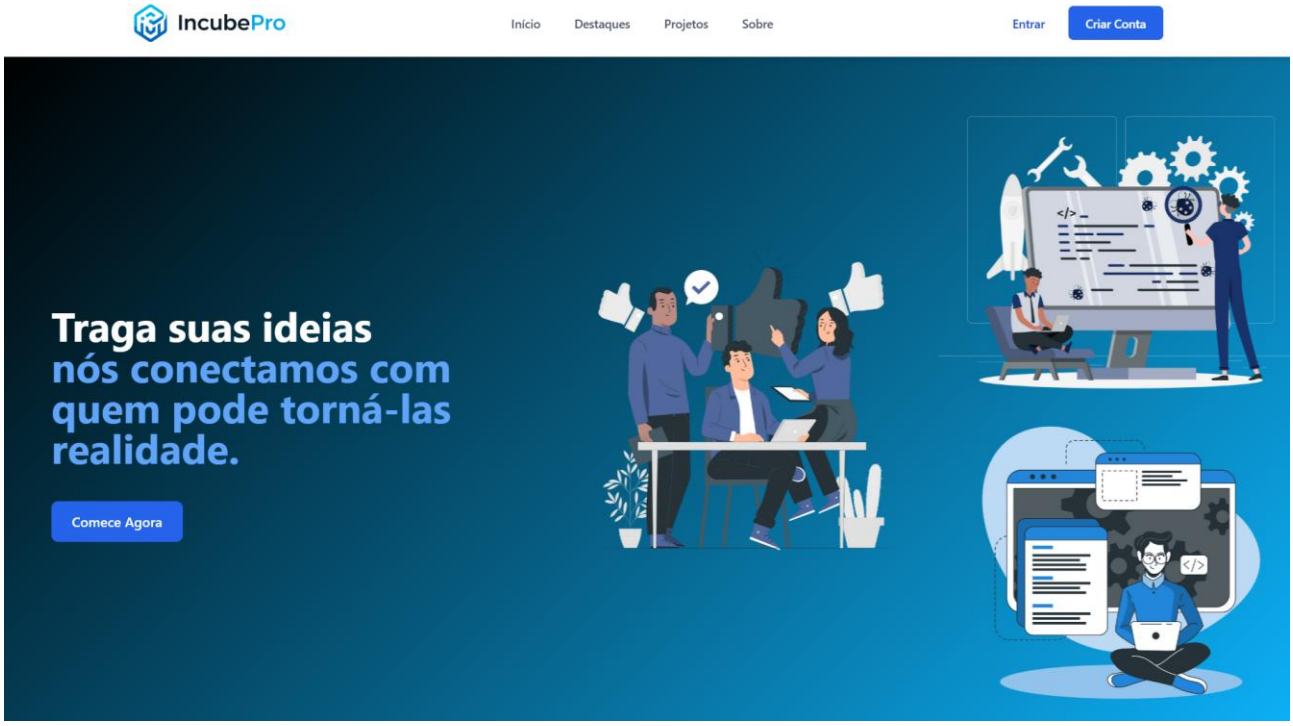
- As interações dinâmicas da interface foram realizadas com Alpine.js, uma biblioteca leve e reativa voltada à composição de comportamento JavaScript diretamente no HTML (CROSSLIGHT, 2025, tradução do autor). A biblioteca também é definida como declarativa e é utilizada para manipulações no DOM, como alternância de visibilidade, alertas e interações de formulários.
- Material Design: padrão visual adotado como referência estética, priorizando acessibilidade, hierarquia de informação e experiência do usuário.

A combinação das ferramentas acima permitiu:

- Rápida prototipação de interfaces dinâmicas com aparência profissional;
- Integração direta com o backend sem a necessidade de APIs assíncronas complexas;
- Navegação com redirecionamentos nativos e persistência de sessão via cookies;
- Baixo custo de configuração e alta produtividade para projetos acadêmicos ou MVPs.

A Figura 1 demonstra a primeira tela do site, ou seja, a página inicial. Nela o usuário pode visualizar informações sobre a ferramenta em si, navegar em outras páginas pelo menu, como destaques e projetos, realizar o seu login ou cadastrar uma nova conta.

Figura 1 – Página inicial da plataforma IncubePro.



Fonte: Elaborada pelo autor (2025).

As Figuras 2, 3, 4, 5 e 6 demonstram mais algumas informações presentes na página inicial. Essas informações dizem respeito ao projeto em si, explicando sobre a ferramenta e trazendo *insights* e dados.

Figura 2 – Informações e insights sobre a plataforma IncubePro na página inicial

Fonte: Elaborada pelo autor (2025).

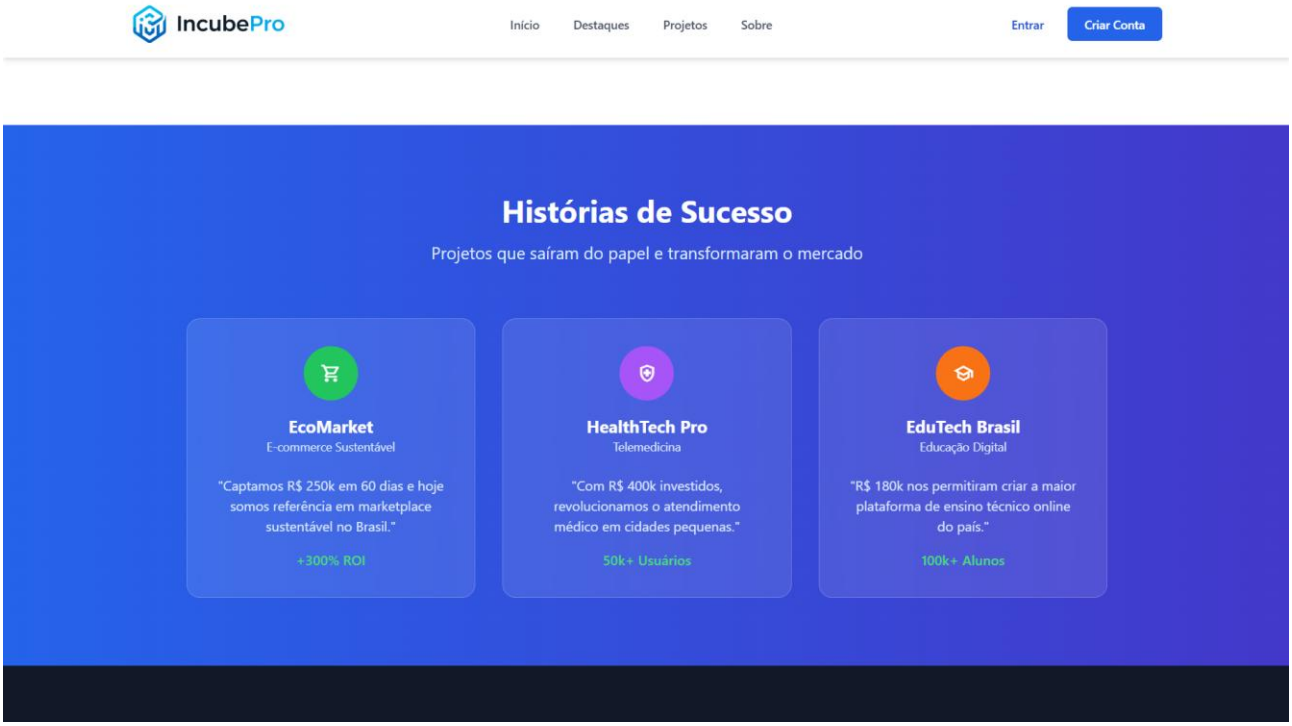
Figura 3 – Informações de funcionamento da plataforma IncubePro

Fonte: Elaborada pelo autor (2025).

Figura 4 – Informações sobre os recursos da plataforma IncubePro

Fonte: Elaborada pelo autor (2025).

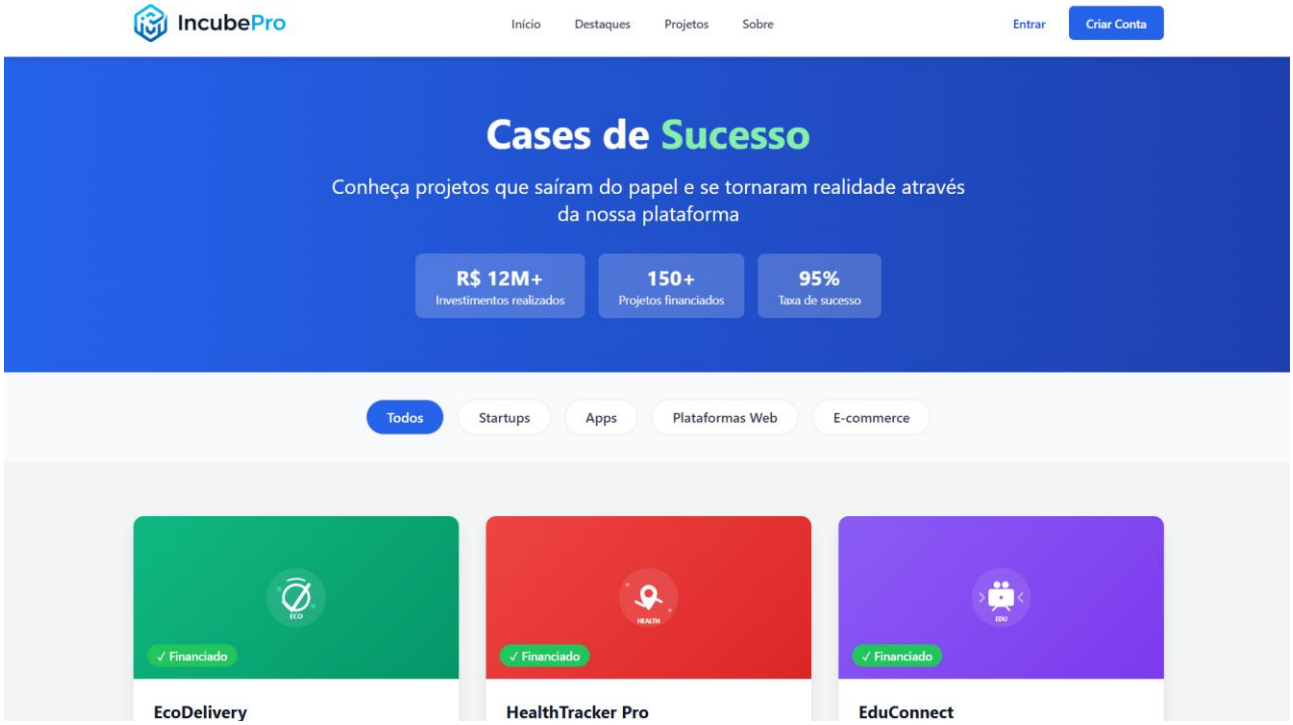
Figura 5 – Cases de sucesso fictícios da plataforma IncubePro



Fonte: Elaborada pelo autor (2025).

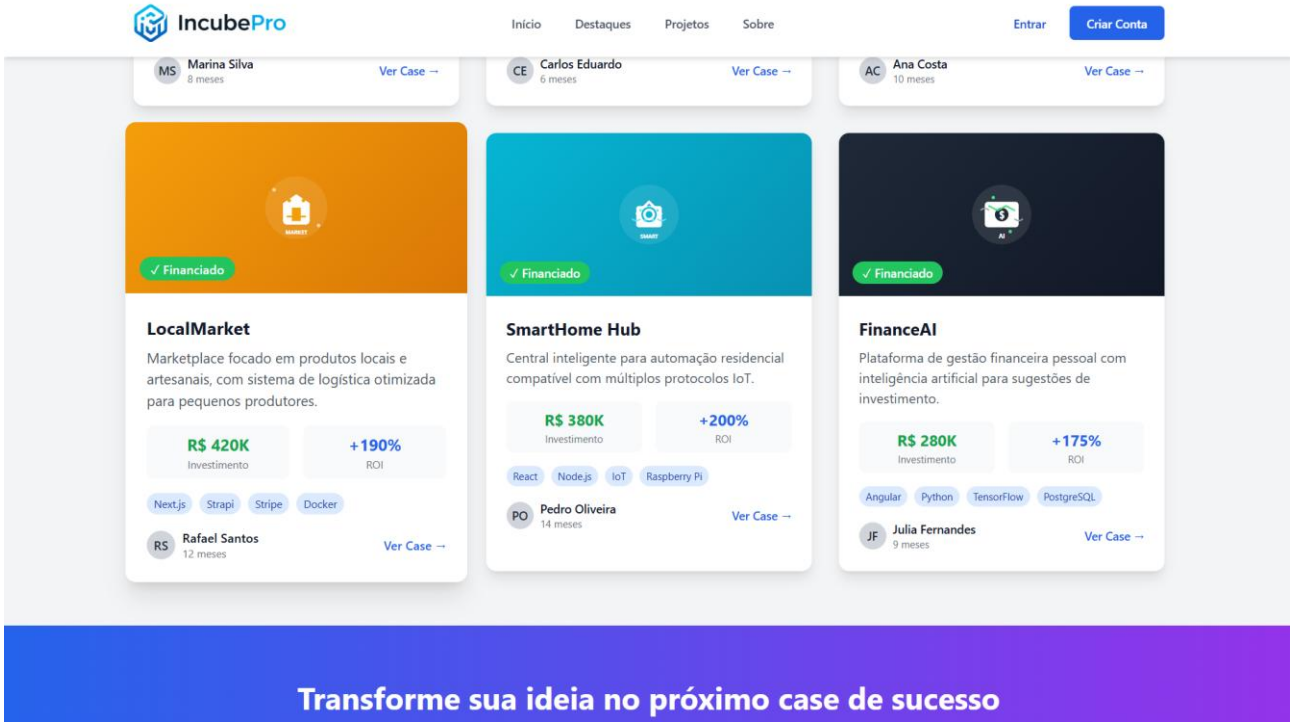
As Figuras 6 e 7 demonstram a página de destaques, onde estão localizados os casos de sucesso da plataforma, com o intuito de atrair o desenvolvedor e demonstrar que o serviço obteve sucesso naquilo que se propõe.

Figura 6 – Início da página de destaques da IncubePro



Fonte: Elaborada pelo autor (2025).

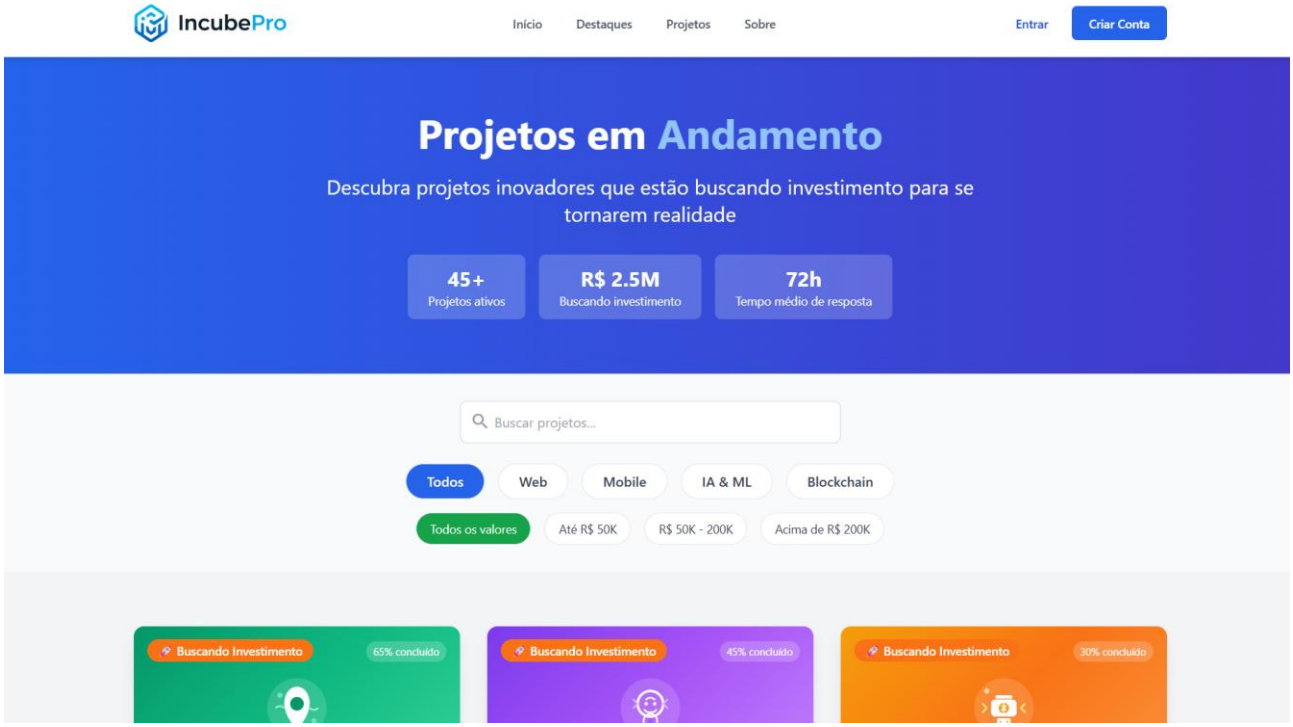
Figura 7 – Projetos na página de destaques da IncubePro



Fonte: Elaborada pelo autor (2025).

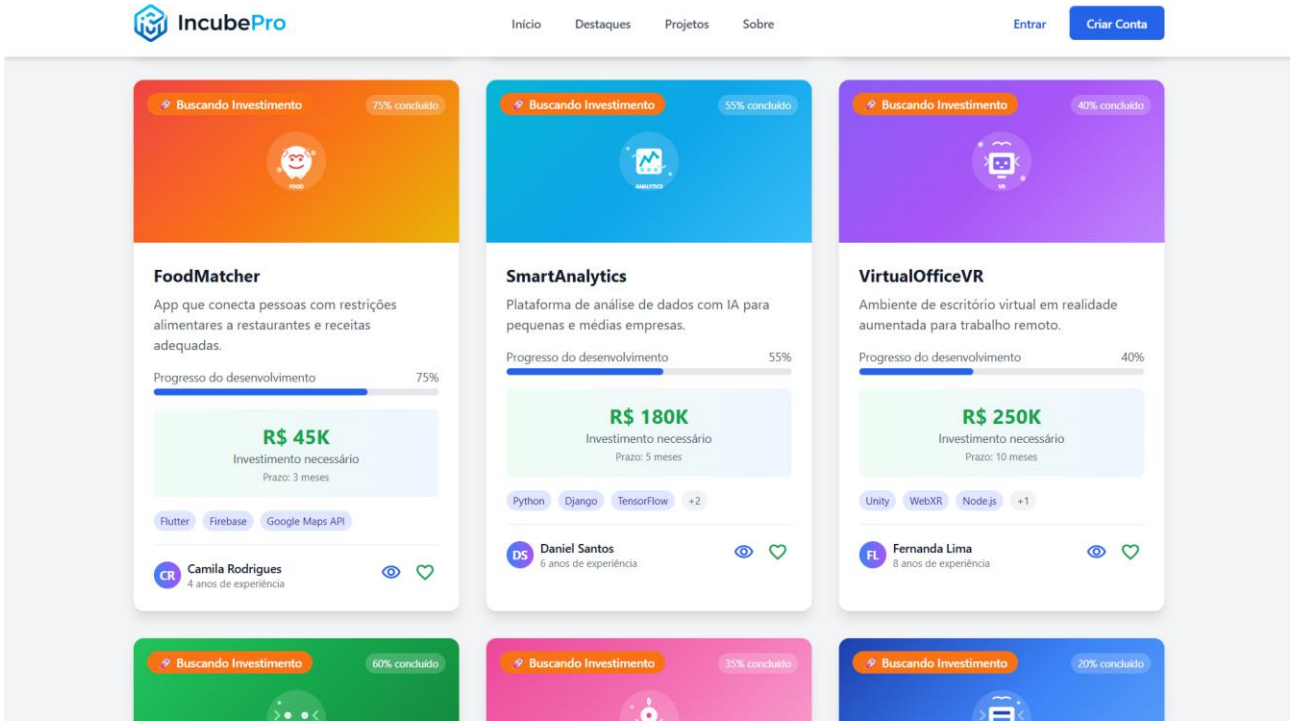
As Figuras 8 e 9 demonstram a página de projetos em andamento da IncubePro. Essa página tem o intuito de trazer informações para os investidores, mostrando os projetos que estão buscando financiamento e demais informações sobre as ideias.

Figura 8 – Início da página de projetos da IncubePro



Fonte: Elaborada pelo autor (2025).

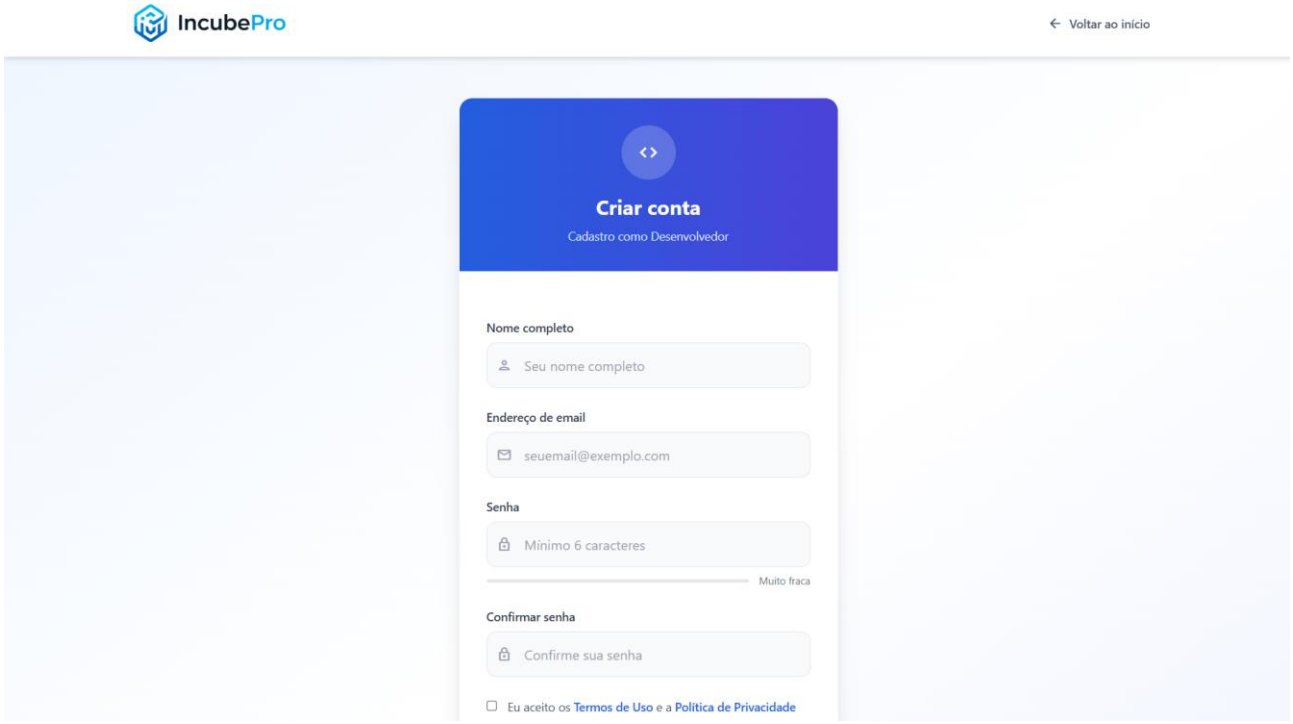
Figura 9 – Projetos em andamento dentro da plataforma IncubePro



Fonte: Elaborada pelo autor (2025).

A Figura 10 demonstra a página de cadastro de uma nova conta dentro da plataforma. Essa página pode ser acessada clicando em “Criar Conta” presente no menu ao lado direito.

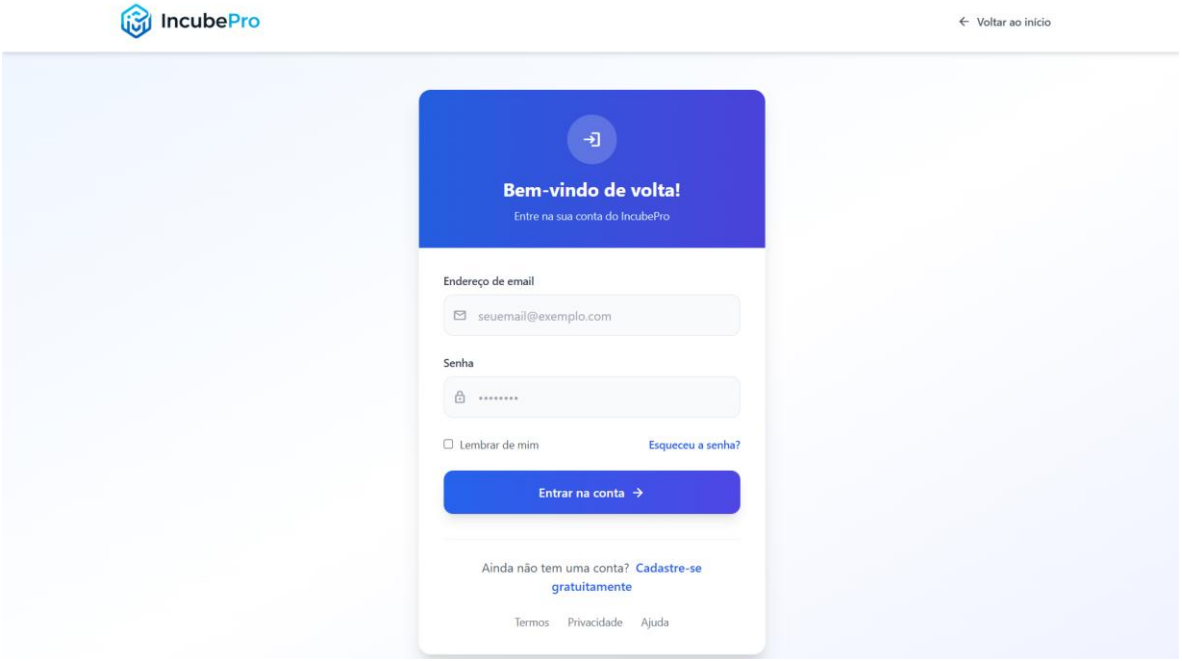
Figura 10 – Página da cadastro de uma nova conta na plataforma IncubePro



Fonte: Elaborada pelo autor (2025).

A figura 11 demonstra a página de login, onde o usuário pode entrar com sua conta já cadastrada na plataforma. Essa página pode ser acessada clicando em “Entrar” presente no menu do site.

Figura 11 – Página de *login* da plataforma IncubePro



Fonte: Elaborada pelo autor (2025).

Os *templates* EJS foram organizados na pasta views/, com divisão entre páginas e componentes reutilizáveis na pasta partials/. A pasta public/ concentrou os arquivos estáticos como imagens, folhas de estilo customizadas e scripts adicionais. Interações com Alpine.js foram definidas diretamente nos atributos HTML, eliminando a necessidade de estruturas de script externas.

A *stack* utilizada possibilitou a construção de uma interface responsiva, clara e funcional, adequada ao objetivo de demonstrar os principais fluxos da aplicação. A utilização de Tailwind e Material Design trouxe consistência visual e foco na experiência do usuário, enquanto Alpine.js agregou dinamismo de forma leve e eficiente.

2.9 Integração e Testes

A integração entre front-end e back-end foi feita de forma acoplada, com as páginas renderizadas diretamente pelo back-end. Isso facilitou a manutenção da sessão via cookies e a execução de redirecionamentos após ações como login e logout.

Foram realizados testes manuais para avaliar:

- Fluxos principais de uso: registro, autenticação, criação e edição de projetos, gerenciamento;
- Validação de regras de negócio: permissões de acesso, validação de formulários, autenticação via token;
- Usabilidade e responsividade da interface, testando diferentes dispositivos e resoluções.

2.10 Forma de Análise dos Resultados

A análise dos resultados consistiu na verificação do comportamento esperado da aplicação em cenários comuns e críticos. Foi avaliada a consistência das respostas HTTP, o armazenamento correto das informações no banco de dados e a estabilidade das sessões autenticadas.

No frontend, a usabilidade e clareza da interface foram examinadas a partir dos princípios de design adotados. As interações realizadas com Alpine.js demonstraram ser suficientes para garantir uma boa experiência, sem sobrecarga computacional ou dependência de frameworks complexos.

A arquitetura adotada, combinando Node.js, MongoDB, EJS, Tailwind CSS e Alpine.js, mostrou-se eficiente para o desenvolvimento de um protótipo funcional, com potencial de expansão e refatoração futura para aplicações mais robustas melhorando ainda mais a experiência do usuário.

3 Resultados

A construção da plataforma IncubePro resultou na implementação bem-sucedida de um protótipo funcional que cumpre os principais requisitos estabelecidos na metodologia. O sistema apresenta uma arquitetura moderna, baseada em tecnologias de código aberto como Node.js, Express.js, MongoDB e EJS, integradas sob o padrão de projeto MVC, o

que proporcionou organização, escalabilidade e clareza na separação de responsabilidades.

O sistema contempla as funcionalidades essenciais para um ambiente colaborativo de financiamento de projetos de software, como:

- Cadastro e autenticação de usuários, com distinção de perfis (desenvolvedores e investidores);
- Criação, edição e listagem de projetos, com campos personalizados como meta financeira, descrição, categoria e recompensas;
- Navegação dinâmica e responsiva por meio de templates EJS, utilizando Tailwind CSS para estilização e Alpine.js para interações no frontend;
- Segurança por meio de autenticação via JWT, criptografia de senhas com bcrypt e middleware de proteção para rotas sensíveis.

Os testes manuais executados abrangeram fluxos críticos do sistema, como registro, login, criação de projetos e acesso ao perfil do usuário. Esses testes permitiram validar tanto a robustez da lógica de negócio quanto a experiência do usuário (UX). A usabilidade foi considerada satisfatória, com boas taxas de resposta, navegação fluida e ausência de falhas críticas regularmente observadas.

O Quadro 1 evidencia as principais funcionalidades implementadas nessa versão da aplicação:

Quadro 1 – Funcionalidades implementadas no protótipo da plataforma IncubePro.

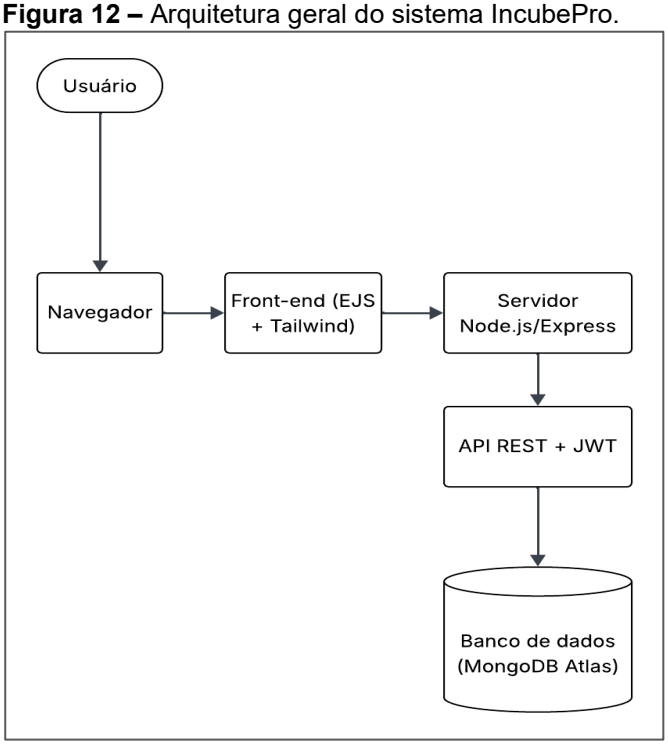
Módulo	Funcionalidades principais
Autenticação	Registros de usuários, login/logout, geração de tokens JWT, persistência de sessão via cookies
Gestão de usuários	Visualização e edição de perfil, proteção de dados sensíveis, controle de permissões
Gestão de projetos	Criação, edição, visualização e categorização de projetos com metas de financiamento
Interface e front-end	Templates dinâmicos com EJS, estilização com Tailwind CSS e Material Design, interações com Alpine.js
Segurança	Criptografia de senhas, verificação de sessão, middleware de proteção e validação de entrada

Fonte: Elaborada pelo autor (2025).

Durante os testes, observou-se que o sistema é capaz de armazenar corretamente os dados no banco MongoDB Atlas, realizar buscas filtradas por categoria e autenticar usuários com baixa latência, mesmo em ambiente local. Além disso, a renderização do conteúdo com EJS se mostrou eficiente, reduzindo o tempo de resposta entre as requisições e proporcionando uma experiência de navegação intuitiva.

A seguir, a Figura 12 ilustra a arquitetura geral da plataforma, evidenciando a integração entre camadas e componentes:

Figura 12 – Arquitetura geral do sistema IncubePro.



Fonte: Elaborada pelo autor (2025).

A aplicação validou as hipóteses principais, demonstrando que é possível construir um ambiente virtual que, mesmo em fase prototípica, já favorece a colaboração entre desenvolvedores e investidores, com mecanismos seguros e funcionais. Além disso, a separação clara entre os papéis de usuário, o uso de rotas protegidas e o armazenamento seguro dos dados mostram que o sistema, a princípio, está alinhado com as boas práticas de desenvolvimento web moderno.

Por fim, os resultados obtidos até esta etapa do projeto confirmam que a plataforma é tecnicamente viável e oferece uma base sólida para futuras expansões, como integração de sistemas de pagamento, gamificação, curadoria de projetos, dashboards de métricas para investidores e até expansões como a idealização de startups e empresas. Isso reforça o potencial da IncubePro como ferramenta de estímulo à inovação tecnológica e à democratização do acesso ao financiamento de projetos de software no Brasil.

4 Discussão

O desenvolvimento da plataforma IncubePro permitiu inferir que é tecnicamente viável implementar um sistema web capaz de conectar desenvolvedores de software e

investidores em um ambiente colaborativo, seguro e funcional. A organização do backend em uma arquitetura baseada no padrão MVC, aliada à integração com tecnologias modernas e de código aberto, demonstrou-se eficiente para garantir a separação lógica de responsabilidades, promover escalabilidade e facilitar futuras expansões do sistema.

Essas constatações apontam para uma contribuição relevante da plataforma: a possibilidade de unir em um único ambiente digital o financiamento coletivo com funcionalidades voltadas ao suporte técnico e à gestão de projetos. Essa integração amplia o papel tradicional das plataformas de *crowdfunding*, oferecendo recursos que favorecem a validação de ideias, a continuidade de projetos e a interação entre diferentes perfis de usuários. A presença de controles de acesso, autenticação segura via JWT e mecanismos de proteção de dados reforça a maturidade técnica do sistema mesmo em sua fase prototípica.

A estruturação da camada de apresentação com EJS, Tailwind CSS e Alpine.js, apesar de simples, se mostrou eficaz para garantir uma boa experiência do usuário. A decisão de utilizar ferramentas leves e com baixa curva de aprendizado contribuiu para a rápida prototipação de uma interface funcional, responsiva e acessível, sem comprometer a clareza ou a usabilidade.

A integração entre financiamento coletivo e suporte técnico estruturado, proposta pela IncubePro, responde à lacuna mencionada por Bastos (2021), que aponta a falta de apoio técnico como uma limitação das plataformas atuais de *crowdfunding*. Esses elementos, em conjunto, permitem afirmar que o objetivo central do trabalho — desenvolver uma solução capaz de fomentar a inovação tecnológica por meio da colaboração entre desenvolvedores e investidores — foi atendido de forma coerente. O sistema não apenas oferece os meios técnicos para que essa conexão ocorra, mas também estabelece uma base sólida para que novas funcionalidades sejam agregadas, com potencial de gerar impacto positivo no ecossistema digital.

5 Conclusão

A construção da plataforma IncubePro permitiu confirmar a viabilidade de um sistema digital funcional, seguro e centrado na colaboração entre desenvolvedores e investidores para o financiamento e desenvolvimento de projetos de software. A arquitetura adotada, baseada no modelo MVC, demonstrou-se eficaz para garantir organização estrutural e separação de responsabilidades, aspectos essenciais para aplicações escaláveis e de manutenção facilitada.

As funcionalidades implementadas contemplaram os requisitos fundamentais do projeto, como autenticação segura, registro diferenciado de perfis de usuário, criação e gestão de projetos com metas de financiamento, e visualização de informações de forma clara e acessível. O uso de tecnologias modernas e consolidadas no ecossistema JavaScript contribuiu para a eficiência da aplicação e para a integridade dos dados tratados.

O sistema demonstrou capacidade de atender ao objetivo central do trabalho, oferecendo uma solução que promove a interação entre diferentes perfis do ecossistema tecnológico, fomenta a inovação e contribui para a valorização de talentos nacionais. A escolha das tecnologias e a forma como foram integradas permitiram a construção de um protótipo funcional com desempenho satisfatório em testes manuais.

Conclui-se, portanto, que a plataforma IncubePro é tecnicamente viável, funcionalmente completa em seu escopo inicial e alinhada às demandas de um ambiente digital colaborativo voltado ao estímulo da inovação. A aplicação atendeu aos critérios estabelecidos para sua concepção e execução, validando sua proposta e consolidando-se como uma solução promissora para fortalecer o ecossistema de desenvolvimento de software por meio do financiamento coletivo.

Referências

ABSTARTUPS. **Mapeamento do ecossistema brasileiro de startups**. Associação Brasileira de Startups, São Paulo, 2024. Disponível em: https://www.abstartups.com.br/wp-content/uploads/2024/11/Mapeamento-do-Ecossistema-Brasileiro-de-Startups-2024.pdf. Acesso em: 20 mai. 2025.

ALVES, Fábia Santos. **Um estudo das startups no Brasil**. Monografia. Universidade Federal da Bahia, 2013.

AMEDOMAR, André de Azevedo. **O crowdfunding de recompensas como alternativa de capital empreendedor para EBTs no Brasil: um estudo descritivo-exploratório**. Tese de Doutorado. Universidade de São Paulo, São Paulo, 2015.

BASTOS, Ane Natalie Pinheiro. **Investidores qualificados e crowdfunding de investimento.** 2021. 60 f. Dissertação (Mestrado em Ciências Econômicas) - Faculdade de Ciências Econômicas, Universidade do Estado do Rio de Janeiro, Rio de Janeiro, 2021.

CARDOSO, Luciano Santos; ZILBER, Moisés Ari. **Crowdfunding e inovação: um estudo sobre startups no Brasil**. Observatorio de La Economía Latinoamericana, 2017.

TI INSIDE. **Brasil é o 2º país com maior intenção de investimentos em TI, indica pesquisa da Grant Thornton**. TI Inside, São Paulo, 19 fev. 2024. Disponível em: https://tiinside.com.br/19/02/2024/brasil-e-o-2o-pais-com-maior-intencao-deinvestimentos-em-ti-indica-pesquisa-da-grant-thornton/. Acesso em: 20 mai. 2025.

CROSSLIGHT. **Alpine.js Documentation**. Disponível em: https://alpinejs.dev/. Acesso em: 24 maio 2025.

EJS. **EJS Documentation**. Disponível em: https://ejs.co/. Acesso em: 24 maio 2025.

NODE.JS FOUNDATION. **Node.js Documentation**. Disponível em: https://nodejs.org/en/docs. Acesso em: 25 maio 2025.

OPENJS FOUNDATION. **Express.js Documentation**. Disponível em: https://expressjs.com/. Acesso em: 25 maio 2025.

TAILWIND LABS. **Tailwind CSS Documentation**. Disponível em: https://tailwindcss.com/docs. Acesso em: 25 maio 2025.