

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE ENGENHARIA DE COMPUTAÇÃO

DIEGO BERGONSI DE FARIAS

**Desenvolvimento de Aplicativo Social
utilizando plataforma no-code**

Monografia apresentada como requisito parcial
para a obtenção do grau de Bacharel em
Engenharia da Computação

Orientador: Prof. Dr. Renato Perez Ribas

Porto Alegre
2024

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitor: Prof. Carlos André Bulhões Mendes

Vice-Reitora: Prof^a. Patricia Helena Lucas Pranke

Pró-Reitor de Graduação: Prof^a. Cíntia Inês Boll

Diretora do Instituto de Informática: Prof^a. Carla Maria Dal Sasso Freitas

Coordenador do Curso de Engenharia de Computação: Prof. Cláudio Machado Diniz

Bibliotecária-chefe do Instituto de Informática: Alexsander Borges Ribeiro

*“Share your knowledge.
It is a way to achieve immortality.”*
— DALAI LAMA

AGRADECIMENTOS

A vida em si é composta por inúmeras curvas, retornos, rótulas e, eventualmente, longas retas. A minha trajetória para o curso e formação em Engenharia da Computação não foi diferente. Tive diversas experiências em diferentes áreas até chegar onde estou e tudo isso contribuiu para uma formação incrivelmente rica. Mas, eu jamais conseguiria tudo isso, todas essas mudanças e a perseverança para alcançar o que eu realmente queria sem a ajuda de várias pessoas, para as quais dedico esse capítulo.

Agradeço imensamente à minha namorada Amanda Luiza Marques, que me acompanha desde o ensino médio no Colégio Mauá em Santa Cruz do Sul. Tenho nela literalmente meu porto seguro, em que sei que sempre terei a verdade, clareza e apoio para as decisões que tomarei. Não acredito que teria tanta determinação, foco e persistência para concluir essa importante etapa da minha vida sem ela ao meu lado. Agradeço, de mesma importância, minha família. Meu pai, Jorge Antonio de Farias, e minha mãe, Cristine Bergonsi de Farias, por sempre me darem toda a estrutura, apoio e incentivo para as diferentes áreas que experimentei antes de estudar Engenharia da Computação, de me apoiarem nas mudanças e, principalmente, de meu pai ter insistido na minha capacidade de conseguir passar no vestibular para Engenharia da Computação, algo que eu não acreditava que fosse capaz. Minhas irmãs, Camila Bergonsi de Farias e Luana Bergonsi de Farias, também sempre estiveram presentes como fortes pontos de apoio para conseguir manter o foco, objetivo e a determinação deste longo processo de graduação.

Agradeço, também, aos amigos que desenvolvi ao longo dos anos de universidade e que infelizmente acabamos nos separando ao longo destes anos, seja para troca de curso, troca de cidade ou até mesmo conseguindo se formar antes. Em especial, agradeço ao meu amigo Nicholas de Aquino Lau por me mostrar, auxiliar e guiar em diversas disciplinas em virtude de ele ter entrado no curso antes de mim, inúmeras vezes esclareceu situações bastante nebulosas.

Por fim, mas não com menos importância, agradeço a todos os professores que fizeram parte dessa trajetória da minha graduação, mas em especial ao professor Renato Perez Ribas, meu orientador, que permitiu e me mostrou uma possibilidade de escrever um trabalho de graduação com um foco em algo que eu realmente me sentisse feliz em desenvolver.

RESUMO

A sociedade, a cada ano incrivelmente mais conectada, peca em promover essa conexão para todas as classes sociais. O custo atual para o desenvolvimento de uma aplicação por um time de engenheiros é altíssimo, algo que apenas grandes empresas e conglomerados conseguem arcar para desenvolver. Mesmo tendo diversos cursos online para aprendizado de tecnologias e ferramentas para o desenvolvimento de um aplicativo, é um processo que leva muito tempo, exige dedicação exclusiva e deve ser contínuo, pois a cada dia novas funcionalidades são lançadas.

Neste contexto, este projeto destaca a viabilidade da plataforma no-code Adalo como uma ferramenta eficaz na criação e implementação ágil de aplicativos, proporcionando uma abordagem acessível e de baixo custo em comparação com os métodos tradicionais de desenvolvimento. O estudo examina as implicações sociais e econômicas do uso de tecnologias no-code, evidenciando como essa abordagem pode democratizar o acesso à inovação digital, contribuindo para o avanço de causas sociais e proporcionando uma solução viável para pessoas, empresas ou organizações com recursos financeiros limitados. A pesquisa abrange as etapas do desenvolvimento do aplicativo, desde a concepção do MVP até a sua transformação em um aplicativo plenamente funcional, destacando os desafios, benefícios e considerações éticas associadas ao uso de plataformas no-code em projetos sociais. Este trabalho busca contribuir para a compreensão do papel crucial da tecnologia no-code no setor social, promovendo a eficiência, a acessibilidade e a sustentabilidade na implementação de soluções tecnológicas para organizações com propósitos humanitários.

Palavras-chave: No-Code. Adalo. Desenvolvimento de Software. MVP.

Use of the Adalo no-code Platform for Creating MVPs and Real Applications

ABSTRACT

Society, increasingly interconnected each year, falls short of promoting this connection across all social classes. The current cost of developing an application by an engineering team is remarkably high, something only large companies and conglomerates can afford to undertake. Despite the availability of various online courses for learning technologies and tools for app development, it is a time-consuming process that demands exclusive dedication and must be continuous, as new features are launched every day.

In this context, this project highlights the feasibility of the no-code platform Adalo as an effective tool in the swift creation and implementation of applications, providing an accessible and cost-effective approach compared to traditional development methods. The study examines the social and economic implications of using no-code technologies, demonstrating how this approach can democratize access to digital innovation, contribute to the advancement of social causes, and offer a viable solution for people, companies, and organizations with limited financial resources. The research covers the stages of app development, from the conception of the MVP to its transformation into a fully functional application, emphasizing the challenges, benefits, and ethical considerations associated with the use of no-code platforms in social projects. This work aims to contribute to the understanding of the crucial role of no-code technology in the social sector, promoting efficiency, accessibility, and sustainability in the implementation of technological solutions for humanitarian organizations.

Keywords: No-Code, Adalo, Software Development, MVP.

LISTA DE ABREVIATURAS E SIGLAS

MVP	Minimum Viable Product (Produto Mínimo Viável)
API	Application Programming Interface
TI	Tecnologia da Informação
IAP	In-App Purchases (Compras Dentro do Aplicativo)
UI	User Interface (Interface do Usuário)
ABNT	Associação Brasileira de Normas Técnicas

LISTA DE FIGURAS

Figura 2.1	Telas para Restauração de Senha.....	16
Figura 2.2	Componentes da Plataforma Adalo	17
Figura 2.3	<i>Marketplace</i> de Componentes Externos na Adalo.....	17
Figura 2.4	Lógica de Negócios na Adalo.....	18
Figura 2.5	Personalização de Fontes.....	19
Figura 2.6	Planos Pagos na Adalo.....	20
Figura 4.1	Coleção de Entidades e Adição de Nova Entidade.....	32
Figura 4.2	Permissões da Entidade	33
Figura 4.3	Permissões da Entidade	33
Figura 4.4	Tela de <i>Login</i>	34
Figura 4.5	Formulário da Tela de <i>Login</i>	34
Figura 4.6	Campos Obrigatórios no Formulário	35
Figura 4.7	Submissão do Formulário	35
Figura 4.8	Criação de Conta	37
Figura 4.9	Telas de Restauração de Senha	38
Figura 4.10	Envio de Código para Restauração de Senha	39
Figura 4.11	Envio de Código para Restauração de Senha	40
Figura 4.12	Envio de Código para Restauração de Senha	41
Figura 4.16	Criação de Notícia	41
Figura 4.13	Permissão Inválida para Criar Projeto	42
Figura 4.14	Criação de Projeto	43
Figura 4.15	Permissões Inválidas para Criação de Notícia.....	44
Figura 4.17	Tela de Perfil	45
Figura 4.18	Tela de Perfil - Informações ONG	45
Figura 4.19	Tela Inicial	46
Figura 4.20	Notícia - Destaque	46
Figura 4.21	Projeto - Destaque	47
Figura 4.22	Projeto - Destaque	47

LISTA DE TABELAS

Tabela 4.1	Descrição da Entidade Usuário	30
Tabela 4.2	Descrição da Entidade Notícias	31
Tabela 4.3	Descrição da Entidade Categoria.....	31
Tabela 4.4	Descrição da Entidade Projeto.....	31

SUMÁRIO

1 INTRODUÇÃO	11
2 FUNDAMENTAÇÃO TÉCNICA E TEÓRICA	13
2.1 O que é uma plataforma <i>no-code</i>?	13
2.1.1 Definição e Fundamentos	14
2.1.2 Interface Intuitiva e Ferramentas Visuais	14
2.1.3 Processos, Lógica e Integrações	14
2.1.4 Colaboração, Atualizações e Manutenção	15
2.2 A plataforma <i>no-code</i> Adalo	15
2.2.1 Pontos Positivos	16
2.2.2 Pontos Negativos	18
2.2.3 Possíveis Melhorias	19
2.3 Desenvolvimento de <i>Software</i>: Tradicional x <i>No-Code</i>	20
2.3.1 Vantagens do Desenvolvimento de <i>Software</i> Tradicional	21
2.3.2 Desvantagens do Desenvolvimento de <i>Software</i> Tradicional	22
2.3.3 Comparação com Desenvolvimento na Adalo	23
2.4 Projetos Viáveis para Desenvolvimento na Adalo	25
2.4.1 Aplicações MVPs ou de Uso Interno	25
2.4.2 Gerenciamento de Eventos e <i>Marketplace</i>	25
2.4.3 Comunidades e Redes Sociais de Menor Escala	26
2.4.4 Necessidade de Boa Avaliação para um bom Desenvolvimento	26
3 REVISÃO BIBLIOGRÁFICA E PROPOSTA	27
3.1 Importância de Aplicativos no Mercado Contemporâneo	27
3.2 Conhecimento de Plataformas <i>no-code</i>	28
3.3 Proposta, Objetivos e Metodologia	28
3.3.1 Objetivos	28
3.3.2 Metodologia	29
4 DESENVOLVIMENTO E RESULTADOS	30
4.1 Criação das Entidades	30
4.1.1 Estruturação das Entidades	30
4.1.2 Mecanismo para Criar Entidades na Adalo	32
4.1.3 Permissões das Entidades	33
4.2 Criação das Telas	33
4.2.1 Tela de Acesso ao Aplicativo	34
4.2.2 Tela de Criação de Conta	36
4.2.3 Tela de Restauração de Senha	36
4.2.4 Criação de Projetos	39
4.2.5 Criação de Notícias	40
4.2.6 Tela de Perfil	42
4.2.7 Tela Inicial	42
4.2.8 Notícia em Destaque	43
4.2.9 Projeto em Destaque	44
5 CONCLUSÃO	48
REFERÊNCIAS	50

1 INTRODUÇÃO

A crescente interconexão da sociedade, evidenciada a cada ano, não é em si uma conexão total. No início da década de 2010, com a explosão das redes sociais, conseguimos nos conectar com pessoas ao redor do mundo inteiro de maneira extremamente rápida. As redes sociais, além disso, permitiram, também, que antigas amizades e laços fossem refeitos e que as pessoas ficassem cientes do que estava acontecendo na vida dos outros. Essa era, todavia, uma conexão superficial, pois tínhamos notícias de qualquer pessoa e região do mundo e rapidamente podíamos ficar informados de algo que aconteceu no outro lado do globo, mas não tínhamos muitas informações do que estivesse acontecendo extremamente perto, como projetos comunitários, novos negócios no bairro, possíveis áreas com perigo em determinado momento e diversas outras questões que impactariam a qualidade da comunidade em que vivemos.

Com uma década de existência dessas redes sociais, podemos perceber atualmente que majoritariamente todas focam no coletivo global, no saber o que acontece com todo o mundo invés de saber o que acontece no coletivo local, com pessoas e regiões próximas da vida de cada um. Dessa maneira, com o intuito de aproximar as pessoas e comunidades, nota-se a importância de uma aplicação com propósitos semelhantes aos dessas grandes redes sociais, mas que exiba o local, o bairro, o que realmente temos contato diariamente e que impacta profundamente a vida cotidiana.

Conforme explicita (ROCHA et al., 2019), o custo inicial para o desenvolvimento de um aplicativo pode facilmente passar dos R\$ 5.000,00. Este valor é totalmente desproporcional para alguém que gostaria de verificar se sua ideia é viável, ou para uma pequena empresa que busca alcançar mais clientes criando um aplicativo único para o seu negócio, como uma barbearia para que os clientes possam agendar os horários e, também, para organizações, como a ONG Coletivo Autônomo - Morro da Cruz (CRUZ, 2024), localizada em Porto Alegre, Rio Grande do Sul, que trabalha na comunidade Morro da Cruz e não possui os meios financeiros para arcar com o desenvolvimento para um aplicativo que seja um portal de notícias e informações a respeito do bairro.

Com esse contexto, e atrelado ao fato de que comunidades e bairros mais carentes da sociedade muitas vezes não são noticiados e cobertos por mídias tradicionais (DEMO, 2007), este projeto enfoca a viabilidade da plataforma no-code Adalo como uma solução eficaz, de baixo custo e ágil para a criação e implementação de aplicações. Adalo é uma plataforma online, acessada pelo navegador na internet, que permite a construção de sites

e aplicativos. Conforme (SANTOS, 2022), "A filosofia da plataforma já indica desde o início em seu slogan, "*Build for free, pay to publish*", ou seja, construa sua aplicação de graça mas pague para publicá-la."

Destaca-se a necessidade premente de uma abordagem acessível e de baixo custo em comparação com os métodos tradicionais de desenvolvimento. Este estudo investiga as implicações sociais e econômicas do uso de tecnologias no-code, sublinhando como essa abordagem pode democratizar o acesso à inovação e socialização digital. A pesquisa abrange todas as fases do desenvolvimento do aplicativo, desde a concepção do MVP (Produto Mínimo Viável, do inglês) até sua transformação em um aplicativo plenamente funcional. Ao longo do processo, são destacados os desafios inerentes, os benefícios obtidos e as considerações éticas associadas ao uso de plataformas no-code em projetos sociais. Este trabalho visa não apenas explorar e documentar as potencialidades da tecnologia no-code, mas também contribuir para uma compreensão mais profunda do papel crucial que essa abordagem pode desempenhar no setor social, promovendo acessibilidade, inclusão e diversificação na implementação de soluções tecnológicas para pessoas, empresas ou organizações com recursos financeiros limitados.

A estruturação do presente trabalho foi construída como explicitado a seguir. No Capítulo 2 será descrito o que são efetivamente plataformas *no-code*, quais são seus pontos positivos e negativos, quais tipos de projetos são recomendados para ser desenvolvidos em plataformas *no-code* e também é realizada uma comparação entre o desenvolvimento tradicional de *software* com o que é possível realizar na Adalo e plataformas semelhantes. Em seguida, no Capítulo 3, alguns trabalhos existentes que auxiliaram no desenvolvimento deste trabalho são destacados. Já no Capítulo 4 é descrito todo o desenvolvimento da aplicação na plataforma Adalo, bem como o fluxo de dados, interface do usuário e eventuais problemas em contratos. Por fim, o Capítulo 5 trás a conclusão, com base na experiência do autor no desenvolvimento da aplicação, sobre o uso da plataforma.

2 FUNDAMENTAÇÃO TÉCNICA E TEÓRICA

Este capítulo será destinado para dissertar a respeito da plataforma *no-code* Adalo, que foi escolhida para desenvolver o aplicativo, dentre as várias opções de plataforma *no-code*, em virtude de oferecer personalização completa - tanto para o *frontend* quanto para o *backend* - e por possibilitar facilidade de uso no futuro para atualização, implementação de novas funcionalidades e manutenção pelas pessoas membras da ONG, e comparar com o desenvolvimento tradicional de *software*. Primeiramente, será explicado e demonstrado o que é, detalhadamente, uma plataforma *no-code*. Posteriormente, será focado na plataforma Adalo em si, no que ela pode oferecer, seus pontos positivos, negativos e melhorias que poderiam ser feitas. Além disso, será feito um comparativo em relação ao desenvolvimento de aplicações, considerando o trabalho tradicional realizado por uma equipe de engenheiros, *designers* e gestores contra o que é possível desenvolver pela plataforma Adalo. Por fim, será explicitado porque *no-code* pode ser uma opção incrível para determinados tipos de projetos, como com menor fluxo de usuários ou poucas lógicas complexas, e também porque pode ser algo destinado a fracassar para outros projetos, que, por exemplo, tenham uma necessidade de várias camadas de segurança, ou que terão um tráfego de usuários na casa dos milhões ou, ainda, que necessita de lógicas de negócio extremamente complexas como a análise de crédito para um cliente em uma instituição financeira.

2.1 O que é uma plataforma *no-code*?

As plataformas *no-code* emergem como catalisadores inovadores na esfera do desenvolvimento de *software*, proporcionando uma abordagem revolucionária e acessível para a criação de aplicativos sem a necessidade de codificação extensiva. Estas ferramentas, impulsionadas pelo objetivo de democratizar o desenvolvimento de *software*, capacitam usuários, independentemente de suas habilidades técnicas, a materializarem suas ideias e concepções em aplicações funcionais de maneira ágil e eficaz.

2.1.1 Definição e Fundamentos

No-code, como o termo sugere, refere-se a uma abordagem de desenvolvimento de *software* que dispensa a necessidade de codificação manual extensiva. Tradicionalmente, a codificação era considerada a barreira de entrada para muitos aspirantes a desenvolvedores, exigindo uma compreensão profunda de linguagens de programação e uma curva de aprendizado significativa. As plataformas *no-code* buscam mitigar esses obstáculos, permitindo que usuários com conhecimentos variados em tecnologia possam criar aplicativos sem escrever código.

2.1.2 Interface Intuitiva e Ferramentas Visuais

Uma característica distintiva das plataformas *no-code* é a presença de interfaces intuitivas e ferramentas visuais. Em vez de dependerem de linhas de código, os usuários interagem com os elementos da interface gráfica, configurando funcionalidades por meio de arrastar e soltar, configurações e lógica condicional. Essa abordagem simplificada permite que indivíduos com diversos níveis de proficiência técnica construam aplicativos de maneira eficiente.

Outro ponto crucial nas plataformas *no-code* é a disponibilidade de componentes pré-desenvolvidos e *templates*. Esses elementos prontos para uso abrangem desde formulários e botões até integrações com APIs (Interface de Programação de Aplicações, do inglês) - "APIs podem ser definidas como um conjunto de regras e protocolos que permitem que diferentes sistemas de software se comuniquem e interajam uns com os outros"(LENZ; FRANCISCATTO, 2023), acelerando consideravelmente o processo de desenvolvimento. Essa abordagem modular não apenas economiza tempo, mas também garante a consistência e a qualidade das aplicações construídas.

2.1.3 Processos, Lógica e Integrações

Além da criação visual de interfaces, as plataformas *no-code* oferecem funcionalidades avançadas de automatização de processos e lógica de negócios. Por meio de construtores de fluxo entre telas e regras de automação, os usuários podem definir logicamente o comportamento do aplicativo em resposta a diferentes eventos. Isso não apenas

simplifica a implementação de lógica complexa, mas também capacita os usuários a criar aplicativos dinâmicos e adaptáveis.

A interoperabilidade é fundamental no desenvolvimento de *software*, e as plataformas *no-code* reconhecem essa necessidade. Muitas oferecem a capacidade de integrar-se a uma variedade de serviços e APIs externas, permitindo que os aplicativos *no-code* se conectem a outras ferramentas e sistemas. Isso é particularmente valioso em ambientes empresariais, onde a colaboração e a integração de dados são essenciais.

2.1.4 Colaboração, Atualizações e Manutenção

Outro aspecto inovador das plataformas *no-code* é a promoção da colaboração e acessibilidade. Diferentes membros de uma equipe, independentemente de suas habilidades técnicas, podem contribuir para o desenvolvimento do aplicativo. Isso reduz as barreiras tradicionais entre equipes técnicas e não técnicas, proporcionando uma abordagem mais inclusiva e eficaz para o desenvolvimento de *software*.

O ciclo de vida de uma aplicação *no-code* é facilitado pela capacidade de realizar atualizações dinâmicas e manutenção simplificada. Sem a necessidade de reescrever código, as alterações podem ser implementadas de forma rápida e eficiente. Isso contribui para a adaptabilidade contínua do aplicativo às necessidades cambiantes dos usuários e do ambiente.

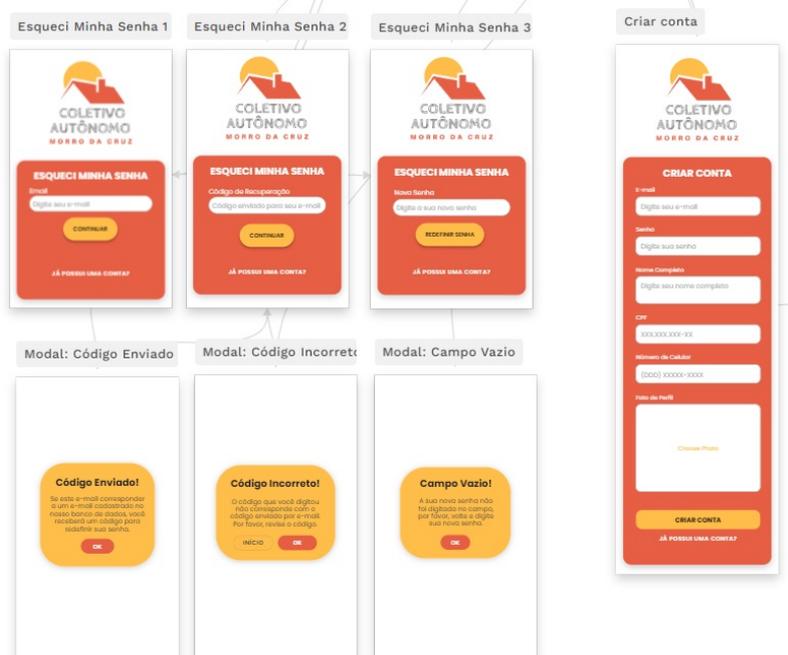
2.2 A plataforma *no-code* Adalo

Adalo destaca-se como uma plataforma *no-code* inovadora, permitindo que usuários criem aplicativos sem a necessidade de habilidades avançadas de programação. Essa ferramenta visa tornar o desenvolvimento de aplicativos mais acessível, capacitando empreendedores, *designers* e profissionais de negócios a transformarem suas ideias em realidade de forma eficiente. A seguir, serão demonstrados os pontos positivos, negativos e possíveis melhorias que podem ser realizadas na plataforma de acordo com a análise realizada pelo autor para o desenvolvimento do aplicativo para a ONG Coletivo.

2.2.1 Pontos Positivos

A Adalo destaca-se por sua interface intuitiva e amigável, projetada para usuários com diferentes níveis de habilidades técnicas. O processo de construção de aplicativos é simplificado, permitindo que os usuários arrastem e soltem elementos, personalizem estilos e configurem lógica de forma visual. Conforme visto na Figura 2.1, cada um dos retângulos é uma tela do aplicativo e dentro há cada componente, que é possível clicar, movimentar, alterar tamanho e outras personalizações. A ação representada nessas telas seria o caminho realizado pelo usuário para restaurar sua senha. Cada tela abaixo da principal é um aviso que é demonstrado na tela para o usuário caso ele cometa algum erro. Na tela maior, à direita, estão todos os campos necessários para o usuário criar uma conta.

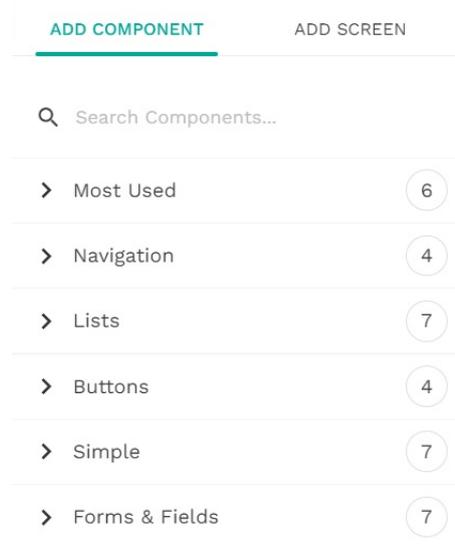
Figura 2.1: Telas para Restauração de Senha



Fonte: Autor

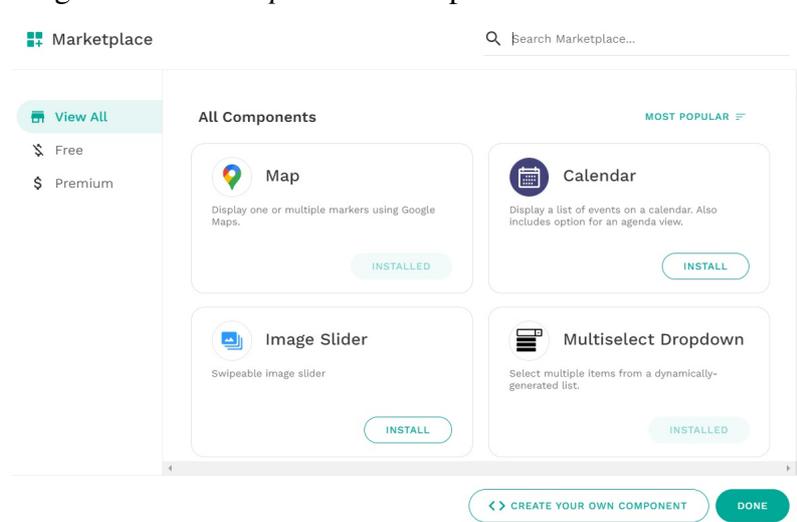
A plataforma oferece uma ampla variedade de componentes pré-desenvolvidos, conforme Figura 2.2, e *templates*, Figura 2.3, acelerando significativamente o processo de desenvolvimento. Isso não apenas economiza tempo, mas também garante consistência e profissionalismo no *design* dos aplicativos.

Figura 2.2: Componentes da Plataforma Adalo



Fonte: Autor

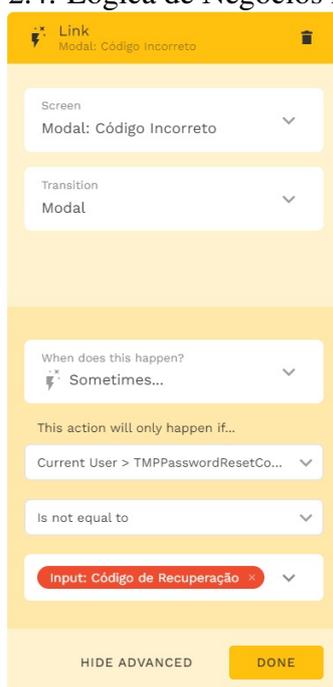
Figura 2.3: Marketplace de Componentes Externos na Adalo



Fonte: Autor

Adalo permite a criação de fluxos de trabalho automatizados por meio de construtores visuais, permitindo que os usuários definam a lógica de negócios e a interação entre diferentes partes do aplicativo. Essa funcionalidade é fundamental para criar aplicativos dinâmicos e interativos. Por exemplo, abaixo na Figura 2.4, foi definida uma lógica na qual será acessada determinada tela apenas se o código enviado para o usuário para restaurar a senha não for igual ao que o usuário digitou na tela do aplicativo, disparando um erro para ele.

Figura 2.4: Lógica de Negócios na Adalo



Fonte: Autor

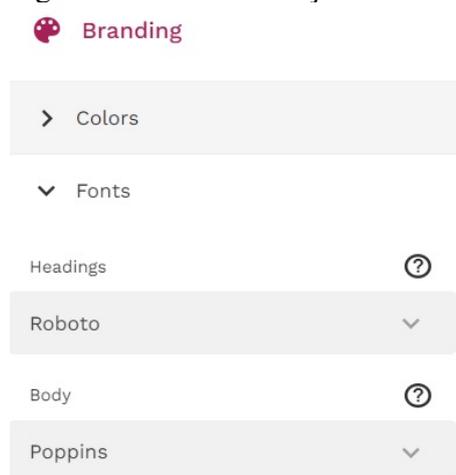
A plataforma oferece uma ampla gama de opções de *design*, desde a personalização de cores até a escolha de tipografia. Essa flexibilidade permite que os usuários criem aplicativos que estejam alinhados com suas marcas e requisitos visuais. Ainda, a Adalo facilita o processo de publicação de aplicativos em várias plataformas, incluindo iOS e Android. A simplicidade desse processo é fundamental para empreendedores e pequenas empresas que desejam levar seus aplicativos rapidamente ao mercado.

2.2.2 Pontos Negativos

A plataforma é poderosa para aplicativos mais simples e de média complexidade, pois pode apresentar limitações quando se trata de funcionalidades mais avançadas. Para projetos que exigem recursos altamente personalizados, como botões dinâmicos em formatos diferenciados que possuam uma animação específica para a identidade visual da marca, ou integrações complexas - como utilização de diversos serviços e bibliotecas externas, pode ser necessário considerar alternativas mais especializadas. Em casos de aplicativos muito complexos ou com grande quantidade de dados, a plataforma pode apresentar desafios de desempenho - como travamento ou lentidão para recarregamento das telas. Isso pode ser uma consideração crítica para aplicativos que visam um grande número de usuários simultâneos ou que manipulam grandes conjuntos de dados.

Apesar de oferecer fontes diferentes e seleção de cores, em certos casos a personalização pode ser considerada limitada para expor devidamente a identidade visual. Usuários avançados podem sentir a necessidade de uma maior amplitude de opções de personalização em determinados elementos da interface. Por exemplo, na Figura 2.5 vemos que em relação às fontes da identidade visual só é possível escolher a fonte para o corpo do texto ou para o título.

Figura 2.5: Personalização de Fontes



Fonte: (ADALO, 2024)

Apesar de a plataforma Adalo não cobrar nada para que se desenvolva todo o aplicativo, desde a identidade visual até a conexão entre todas as telas com todas as lógicas envolvidas, é necessário arcar com um investimento mensal para poder publicar o aplicativo. Além disso, para utilizar fontes alternativas ou, ainda, para desempenhar ações customizadas - como enviar uma mensagem de texto para o número de celular do usuário mediante alguma lógica de negócio - no aplicativo, é necessário ter o plano pago. Todavia, o plano pago de entrada já abrange essas questões e os planos mais caros são destinados principalmente a empresas que terão grande tráfego de usuários. A Figura 2.6 mostra todos os planos disponíveis na plataforma. Infelizmente todos os valores são cobrados em dólares, podendo haver flutuação no valor de acordo com a moeda.

2.2.3 Possíveis Melhorias

Expandir a funcionalidade disponível para alguns recursos na plataforma, como incluir suporte a pequenas customizações utilizando uma linguagem de programação propriamente dita - como a construção de um componente em JavaScript e trazer ele para a

Figura 2.6: Planos Pagos na Adalo



Fonte: (ADALO, 2024)

Adalo, permitindo a implementação de lógica de negócios mais complexa e personalizações avançadas. Somado a isso, investimentos contínuos em melhorias de desempenho podem ser uma área de foco, garantindo que a plataforma continue a oferecer uma experiência eficiente, mesmo em casos de aplicativos mais complexos. Por fim, de acordo com a experiência no desenvolvimento do presente aplicativo, oferecer uma camada mais profunda de personalização em áreas específicas seria muito benéfico, como uma importação de componentes *open-source* desenvolvidos em código ou, ainda, a personalização de telas padrão da plataforma, pois para restaurar a senha de um usuário, por padrão é direcionado para uma tela em inglês, na qual não é possível alterar texto, componentes ou cores.

Em conclusão, a Adalo se destaca como uma plataforma *no-code* acessível e poderosa, tornando o desenvolvimento de aplicativos mais democrático. Embora apresente pontos fortes significativos, é importante reconhecer suas limitações e áreas potenciais de melhoria para atender a uma variedade ainda maior de casos de uso e usuários. O compromisso contínuo com a inovação e a resposta às necessidades do mercado certamente contribuirão para o sucesso contínuo da plataforma no cenário do desenvolvimento de *software no-code*.

2.3 Desenvolvimento de *Software*: Tradicional x *No-Code*

O desenvolvimento de *software* tradicional, conduzido por equipes multidisciplinares de engenheiros de *software*, *designers* e gestores de projetos, tem sido o estado-da-arte em relação ao desenvolvimento de aplicações na indústria de TI (Tecnologia da

Informação) por décadas. No entanto, com a ascensão das plataformas *no-code*, como a Adalo, uma nova abordagem mais acessível e ágil está ganhando destaque, trazendo consigo vantagens e desafios únicos. O presente trabalho abordará minuciosamente essas vantagens para compreender o papel vital que o desenvolvimento tradicional desempenha na indústria.

2.3.1 Vantagens do Desenvolvimento de *Software* Tradicional

- **Controle Total sobre o Código:** Uma das vantagens mais marcantes do desenvolvimento tradicional é o controle absoluto sobre o código-fonte. As equipes têm a liberdade de construir o *software* do zero, ajustando cada linha de código de acordo com os requisitos específicos do projeto. Esse nível de controle é inestimável em situações que demandam personalizações profundas e adaptações complexas.
- **Flexibilidade Tecnológica Ilimitada:** O desenvolvimento tradicional permite uma flexibilidade tecnológica ilimitada. As equipes têm a liberdade de escolher entre uma variedade de linguagens de programação, *frameworks* e bibliotecas, adaptando-se às complexidades únicas do projeto. Isso é particularmente crucial em cenários que requerem integrações intrincadas ou funcionalidades altamente personalizadas.
- **Segurança e Conformidade Rigorosas:** Em setores que exigem altos padrões de segurança e conformidade, o desenvolvimento tradicional é a escolha preferida. As equipes podem implementar práticas rigorosas de segurança, realizar auditorias detalhadas e conduzir testes específicos para garantir a integridade e a confiabilidade do *software*. Esse comprometimento com a segurança é vital em setores como financeiro, saúde e governamental.
- **Adaptação a Projetos Complexos:** Projetos altamente complexos, que exigem arquiteturas robustas e soluções sob medida, encontram no desenvolvimento tradicional o ambiente ideal. As equipes podem criar soluções intrincadas, desenvolver algoritmos avançados e lidar com requisitos sofisticados, garantindo que o *software* atenda às expectativas em cenários desafiadores.
- **Desenvolvimento de Aplicações de Grande Escala:** Para projetos que visam uma escala significativa, como sistemas corporativos ou aplicativos utilizados por milhões de usuários simultaneamente, o desenvolvimento tradicional se destaca. Ele oferece a robustez necessária para suportar uma carga de trabalho extensa e garantir

desempenho consistente em diferentes cenários de uso.

- **Treinamento e Conhecimento Especializado:** O desenvolvimento tradicional valoriza a especialização técnica. As equipes são compostas por profissionais altamente treinados e especializados em suas respectivas áreas. Isso resulta em um ambiente propício para inovações e abordagens profundas ao resolver desafios complexos.
- **Soluções Altamente Personalizadas:** A capacidade de criar soluções altamente personalizadas é uma das maiores vantagens do desenvolvimento tradicional. Cada aspecto do *software* pode ser adaptado para atender às necessidades específicas do cliente ou do usuário final, resultando em uma experiência única e ajustada.
- **Manutenção e Atualizações Precisas:** As equipes têm total responsabilidade pela manutenção do software ao longo do tempo. Isso permite atualizações precisas, correções de *bugs* - um erro ou falha no design ou desenvolvimento de um *software* que pode gerar um comportamento errático, com resultados possivelmente incorretos - de maneira eficiente e implementação de novas funcionalidades de maneira controlada, garantindo a estabilidade contínua do sistema.

2.3.2 Desvantagens do Desenvolvimento de *Software* Tradicional

- **Custo e Tempo Elevados:** O desenvolvimento tradicional frequentemente demanda um investimento significativo em termos de custo e tempo (JORGENSEN; SHEPPERD, 2006). A contratação de especialistas, a criação extensiva de documentação e testes detalhados contribuem para um ciclo de desenvolvimento prolongado. Além disso, o contratante do serviço muitas vezes paga por um longo período no qual sequer está sendo desenvolvido *software*, pois é necessário um longo estudo prévio por parte dos *designers* da identidade da marca, de como as telas serão estruturadas e também demanda muito tempo dos engenheiros em como será a arquitetura da aplicação, quais tecnologias serão utilizadas, qual será o tráfego de usuários na aplicação, pois tudo isso impactará na escolha das tecnologias para não afetar o desempenho da aplicação no futuro.
- **Curva de Aprendizado Pronunciada:** Participar de projetos de desenvolvimento tradicional requer uma curva de aprendizado substancial. A necessidade de conhecimentos avançados em programação cria uma barreira significativa para aqueles

sem experiência técnica. Além disso, é necessário, para conseguir um emprego na área, no mínimo uma formação acadêmica em Sistemas de Informação, Ciência da Computação, Engenharia da Computação ou Engenharia de Software, algo que, além do tempo, requer muito dinheiro durante todo o processo para conseguir se graduar.

- **Dependência de Especialistas:** A dependência de engenheiros altamente especializados pode resultar em gargalos em momentos de alta demanda. Isso pode levar a atrasos no desenvolvimento, especialmente quando a equipe é confrontada com desafios inesperados. Considerando as próprias experiências profissionais que tive, nas três empresas que trabalhei tivemos uma falta de profissionais mais capacitados, com mais anos de experiência. Por mais que seja uma área em expansão exponencial, a formação de novos profissionais leva tempo e esses novos profissionais levarão, no mínimo, cerca de 6 a 8 anos para serem especialistas em uma área.

2.3.3 Comparação com Desenvolvimento na Adalo

Plataformas *no-code*, como a Adalo, têm a oportunidade de preencher uma lacuna que existe em virtude de dois grandes pontos negativos supracitados, o custo e tempo elevados para produzir uma aplicação de maneira tradicional e, também, a dependência de engenheiros e profissionais especialistas para a construção de aplicações mais complexas. Considerando isso, podemos observar as vantagens que a plataforma pode oferecer nessas situações e desvantagens que eventualmente podemos enfrentar em comparação ao desenvolvimento tradicional de *software*. Todavia, destaca-se que o presente trabalho não tem por objetivo considerar que apenas o desenvolvimento por plataformas *no-code* é o suficiente para qualquer aplicação. O desenvolvimento de *software* tradicional é basicamente utilizado para a grande maioria das aplicações. Conforme exposto pela empresa *Apple*¹, (APPLE, 2024), pode-se analisar que todos os aplicativos que o público mais deseja, os mais baixados da loja virtual, não poderiam ser construídos puramente por uma plataforma *no-code*, isso acontece em virtude de que a lógica das regras de negócio e as camadas de segurança necessária para o funcionamento do aplicativo não são atualmente cobertas em uma plataforma *no-code*, visto que a Adalo, por exemplo, oferece apenas modificação de permissões para cada usuário, sendo essa a única camada de segurança.

¹<https://www.apple.com/br/>

- **Custo e Tempo para desenvolver na Adalo:** Ao contrário do desenvolvimento tradicional, para prototipar e desenvolver as telas e conexões entre elas na Adalo não custará absolutamente nada, é possível desenvolver quase toda a aplicação sem despender nenhuma quantia. Isso é extremamente benéfico em virtude de que eventualmente pode-se utilizar a plataforma para testar se será possível construir a aplicação, sem gastar nada, antes de recorrer a uma empresa especializada em TI para isso. Além disso, por mais que exista uma curva de aprendizado da plataforma, o tempo necessário para aprender o que é necessário para desenvolver aplicações na Adalo é muito menor do que o tempo que levaria para estruturar a arquitetura de um *software* e desenvolver a aplicação.
- **Disponibilidade de Especialistas:** Sendo uma pessoa com experiência em desenvolvimento de *software* ou não, a Adalo conta com uma funcionalidade na qual você pode contratar, temporariamente, um especialista para lhe ajudar. Isso é extremamente benéfico pois em algumas situações, durante a construção da aplicação para a ONG Coletivo, diversos obstáculos para implementar funcionalidades mais complexas e avançadas foram encontrados. Tendo em vista a minha graduação e experiência na área como desenvolvedor, foi possível realizar pesquisas para esclarecer e implementar o que era necessário, porém, para usuários sem experiência prévia, a disponibilidade imediata de um profissional para auxiliar é muito eficaz, com custo variável de acordo com a complexidade do problema a ser resolvido.
- **Publicação da Aplicação:** Após construir gratuitamente toda a aplicação na Adalo, pode-se publicá-la na internet, onde pode ser acessada a partir de um navegador, ou em lojas virtuais para celulares caso seja um aplicativo, como *AppStore*² e *Google Play*³. Todavia, para realizar essa publicação há um custo, pois é necessário ser assinante de, no mínimo, o plano mais básico da Adalo. Porém, essa assinatura será um valor extremamente mais baixo do que a contratação de uma empresa para a construção da aplicação. Conforme (ROCHA et al., 2019), o custo apenas para um Engenheiro de *Software* Sênior é de no mínimo R\$ 1.655,60 para uma semana, enquanto que o custo mensal para poder publicar o seu aplicativo na Adalo será de cerca de R\$ 200,00 por mês considerando o plano mais básico, conforme exposto no site da plataforma, na aba de Planos e Preços (ADALO, 2024). Essa diferença é crucial para o público que não conseguiria arcar com um investimento financeiro

²<https://www.apple.com/br/app-store/>

³<https://play.google.com/store/apps>

tão grande.

2.4 Projetos Viáveis para Desenvolvimento na Adalo

A plataforma Adalo é especialmente adequada para projetos que se encaixam em determinados cenários e requisitos. Embora seja uma ferramenta poderosa para desenvolvimento *no-code*, é importante considerar suas características e limitações ao escolher projetos. A partir da análise realizada pelo autor no desenvolvimento do aplicativo para a ONG Coletivo, foi elaborada uma lista de tipos de projetos que mais se adequam ao que a plataforma tem a oferecer.

2.4.1 Aplicações MVPs ou de Uso Interno

Adalo é ideal para a criação rápida de protótipos e MVPs. Buscando validar uma ideia de aplicativo com um lançamento inicial para obter *feedback* dos usuários, a agilidade das plataformas *no-code* em geral pode acelerar significativamente esse processo.

Também, para empresas que precisam de soluções personalizadas para gerenciamento interno, acompanhamento de projetos ou outras ferramentas de negócios, a Adalo oferece uma maneira eficiente de construir aplicativos sem depender de uma equipe de desenvolvimento tradicional. Todavia, se esse aplicativo for se comunicar com diferentes serviços de autenticação da empresa ou se depender de outros serviços internos já utilizados pela empresa, a plataforma Adalo pode não ser capaz de oferecer essa lógica de integração.

2.4.2 Gerenciamento de Eventos e *Marketplace*

Para a organização de eventos, um aplicativo que gerencie inscrições, programação e interação com participantes pode ser facilmente construído na plataforma. Também, para criar catálogos de produtos, portfólios de serviços ou plataformas de *e-commerce* de pequena escala, a Adalo oferece recursos de design visual e integração de dados que podem ser aplicados facilmente. Além disso, também oferece integração com serviços IAP (*in-app purchases* - compras dentro do aplicativo, do inglês).

Somado a isso, aplicativos que realizam pequenos controles, como hábitos alimen-

tares, atividades físicas ou reservas em um restaurante, também são facilmente desenvolvidos na plataforma. A atualização de propriedades do usuário é a principal complexidade nesses aplicativos, e a Adalo oferece diversos serviços internos para gerenciamento dos usuários cadastrados.

2.4.3 Comunidades e Redes Sociais de Menor Escala

Para construir comunidades online ou redes sociais específicas, a Adalo fornece ferramentas para criar perfis de usuários, fóruns e recursos de interação social sem complexidade alguma. Esse tópico, abordado neste presente trabalho, se mostrou muito eficaz na plataforma, pois caso seja necessário, para a comunidade, realizar a filtragem do usuário por localização, a Adalo também permite integração com APIs externas, como a de mapas oferecida pela Google.

2.4.4 Necessidade de Boa Avaliação para um bom Desenvolvimento

A importância de escolher sabiamente os tipos de aplicativos para desenvolver na Adalo reside na otimização do processo de desenvolvimento, na maximização da eficiência e na obtenção de resultados tangíveis de maneira mais rápida. Reconhecer as limitações e as forças dessas plataformas é fundamental para tomar decisões informadas sobre quais projetos são mais adequados para o desenvolvimento no-code.

Ao considerar a Adalo para o desenvolvimento de um projeto específico, é importante avaliar a complexidade do aplicativo desejado, os requisitos de personalização - tanto de interface quanto de lógica - e a escala projetada. Projetos mais simples, ágeis e de menor escala são geralmente os mais recomendados para a plataforma Adalo.

3 REVISÃO BIBLIOGRÁFICA E PROPOSTA

3.1 Importância de Aplicativos no Mercado Contemporâneo

O mercado como um todo está em constante transformação. A partir de 2020, isso se intensificou ainda mais, sendo cada vez mais necessária a interação entre o consumidor e o mercado de maneira digital. Nesse contexto, há um crescimento na necessidade por aplicativos para realizar esse relacionamento e comunicação (GUIDINI, 2018), pois muitos consumidores são fortemente influenciados digitalmente e têm maior probabilidade de despende algum valor monetário do que se tivesse que se deslocar até um local físico (ROSA; CASAGRANDA; SPINELLI, 2017).

Para comércios menores, a expansão do seu comércio para o mundo digital a partir de aplicativos representa um passo crucial na adaptação ao cenário comercial contemporâneo. Ao invés de depender exclusivamente de desenvolvedores terceirizados, os proprietários desses estabelecimentos podem construir soluções digitais personalizadas para atender às necessidades específicas de seus negócios. Essas soluções proporcionam uma entrada no mundo digital sem a necessidade de investimentos significativos, permitindo que pequenos comércios alcancem clientes, a partir do seu próprio aplicativo, diretamente em seus dispositivos móveis. Em um ambiente de negócios dinâmico, a capacidade de responder rapidamente às mudanças nas demandas do mercado é fundamental. Aplicativos facilitam esse processo, permitindo que os empreendedores estejam sempre alinhados com as expectativas em constante evolução de seus clientes.

A personalização é um fator diferencial, possibilitando que os proprietários adaptem seus aplicativos às particularidades de seus negócios. Essa flexibilidade garante uma experiência única para os clientes, destacando-se em um mercado muitas vezes saturado. A autonomia proporcionada pelo desenvolvimento próprio de aplicativos coloca os empreendedores no controle de suas soluções digitais. Essa independência não apenas economiza custos, mas também capacita os proprietários a inovar continuamente, adaptando-se às mudanças do mercado. A importância de aplicativos simples para comércios menores vai além da mera presença digital. Essas soluções representam uma ferramenta acessível, ágil e personalizada para impulsionar o crescimento, melhorar a experiência do cliente e garantir a relevância em um ambiente comercial cada vez mais digitalizado (GUIDINI, 2018).

3.2 Conhecimento de Plataformas *no-code*

Atualmente, existem várias plataformas *no-code* no mercado, cada uma com sua vantagem e desvantagem. O presente trabalho foi desenvolvido na plataforma Adalo em virtude, principalmente, das vantagens da plataforma expostas em (SANTOS, 2022) em comparação às demais plataformas. Além disso, muitas plataformas focam em um mercado específico. Conforme (SANTOS, 2022), a Shopify¹ oferece também um serviço de criação de aplicativos *no-code*, todavia, seguindo o mercado da própria empresa, é fortemente focado em comércio eletrônico. Outras plataformas com propósitos específicos também existem, como a AirTable e Google Sheets, as quais são "plataformas voltadas para a criação de bancos de dados internos e públicos, a fim de administrar negócios e obter informações sobre sua operação"(SANTOS, 2022).

Ainda, (SANTOS, 2022) disserta sobre diversas outras plataformas que oferecem desenvolvimento *no-code*, desde para programadores até para designers e gerentes de produtos. Em relação especificamente às plataformas para desenvolvimento de aplicativos, a Thunkable foi a plataforma concorrente para o desenvolvimento do aplicativo para a ONG Coletivo, todavia a Adalo foi escolhida em virtude do MVP proposto, desenvolvido e exposto por (SANTOS, 2022).

3.3 Proposta, Objetivos e Metodologia

O presente trabalho tem como proposta o desenvolvimento de uma aplicação para a ONG Coletivo - Morro da Cruz, na qual poderá ser gerenciada notícias e eventos coletivos em relação à comunidade.

3.3.1 Objetivos

Foi decidido o uso da plataforma *no-code* Adalo para atingir esse objetivo em virtude de que muitas pessoas que integram a ONG não possuem conhecimento necessário para trabalhar com código no desenvolvimento de uma aplicação, portanto, ao desenvolver um aplicativo na Adalo, os membros da ONG poderiam, no futuro, continuar trabalhando na aplicação através dessa plataforma, sem precisar escrever código.

¹<https://www.shopify.com/br>

3.3.2 Metodologia

1. **Levantamento de Requisitos:** O ponto de partida consistiu em um levantamento abrangente de requisitos, onde as necessidades e objetivos do aplicativo foram identificados em detalhes. Esta fase envolveu a colaboração estreita com membros da ONG para assegurar uma compreensão profunda das expectativas e funcionalidades essenciais.
2. **Prototipagem e Wireframing:** Utilizando as ferramentas de prototipagem disponíveis na plataforma Adalo, foram criados *wireframes* e protótipos interativos. Esta etapa foi crucial para visualizar a arquitetura da interface do usuário (UI - interface do usuário, do inglês) e estabelecer uma base sólida para o desenvolvimento subsequente.
3. **Desenvolvimento de Componentes Visuais:** A criação de componentes visuais para a UI foi uma fase crucial, enfatizando o design intuitivo e a usabilidade. O editor visual da Adalo permitiu o posicionamento preciso de elementos, embora tenha sido observada a ocorrência de alguns desafios relacionados ao alinhamento, resolvidos com iterações e ajustes contínuos. Os componentes visuais foram desenvolvidos tendo contínuo *feedback*, para que estivesse alinhado exatamente da maneira que a ONG gostaria que fizesse e também para que estivesse de acordo com a identidade visual da ONG.
4. **Testes e Validação:** Por fim, seria realizada uma validação com os usuários moradores do Morro da Cruz buscando testar se os requisitos supracitados foram alcançados e, também, se haveria a necessidade de incluir mais alguma funcionalidade que o usuário necessitasse.

4 DESENVOLVIMENTO E RESULTADOS

4.1 Criação das Entidades

Ao iniciar o desenvolvimento da aplicação, foram criadas as entidades presentes na aplicação. Uma entidade é uma representação de um conjunto de informações sobre determinado conceito do sistema. Toda entidade possui atributos, que são as informações que referenciam a entidade - (DEV MEDIA, 2024)

4.1.1 Estruturação das Entidades

A entidade *usuários* é criada por padrão na Adalo com os campos *e-mail*, *password* e *username*. Como *username* não seria utilizado na aplicação, foi renomeado para CPF. A Tabela 4.1 abaixo mostra todas as propriedades que foram criadas para a entidade *usuário*.

Tabela 4.1: Descrição da Entidade Usuário

Propriedade	Descrição
E-mail	E-mail do usuário cadastrado no aplicativo
Password	Senha do usuário cadastrado no aplicativo
CPF	CPF do usuário
Full Name	Nome completo do usuário
Permissions	Permissões que o usuário tem no aplicativo
PhoneNumber	Número de telefone celular do usuário
Address	Endereço de residência do usuário
Profile Photo	Foto de perfil do usuário
TMPNewsRating	Propriedade temporária para guardar a nota atribuída a uma notícia pelo usuário
TMPPasswordResetCode	Propriedade temporária para restaurar a senha do usuário em caso de esquecimento
Projects	Relação de projetos criados pelo usuário (um para muitos)
News	Relação de notícias criadas pelo usuário (um para muitas)

Fonte: Elaborado pelo Autor

Posteriormente foi criada a entidade *notícias*, para representar as notícias da comunidade que seria publicadas no aplicativo. Na Tabela 4.2 abaixo pode ser vista cada propriedade e sua descrição.

A terceira entidade foi a de *categoria* - que representará a categoria de uma notícia ou de um projeto, que ficará a cargo de um administrador, pois será o único tipo de usuário

Tabela 4.2: Descrição da Entidade Notícias

Propriedade	Descrição
NewsTitle	Título da notícia
NewsDesc	Descrição da notícia
NewsImage	Imagem de capa da notícia
Reviews	Quantidade de avaliações na notícia
Rating	Avaliação da notícia dada pelos usuários
User	Relação com o usuário que criou a notícia
Category	Relação com a tabela de categorias

Fonte: Elaborado pelo Autor

com permissões suficientes para adicionar, atualizar ou remover categorias. A Tabela 4.3 abaixo demonstra as propriedades e respectivas descrições da entidade.

Tabela 4.3: Descrição da Entidade Categoria

Propriedade	Descrição
CategoryName	Nome da categoria
News	Relação das notícias atreladas à categoria (uma para muitas)
Projects	Relação dos projetos atrelados à categoria (uma para muitas)

Fonte: Elaborado pelo Autor

Por fim, a última entidade adicionada foi a *projeto*, a qual representará todos os projetos presentes na comunidade. Abaixo, na Tabela 4.4, pode-se ver as propriedades e descrições.

Tabela 4.4: Descrição da Entidade Projeto

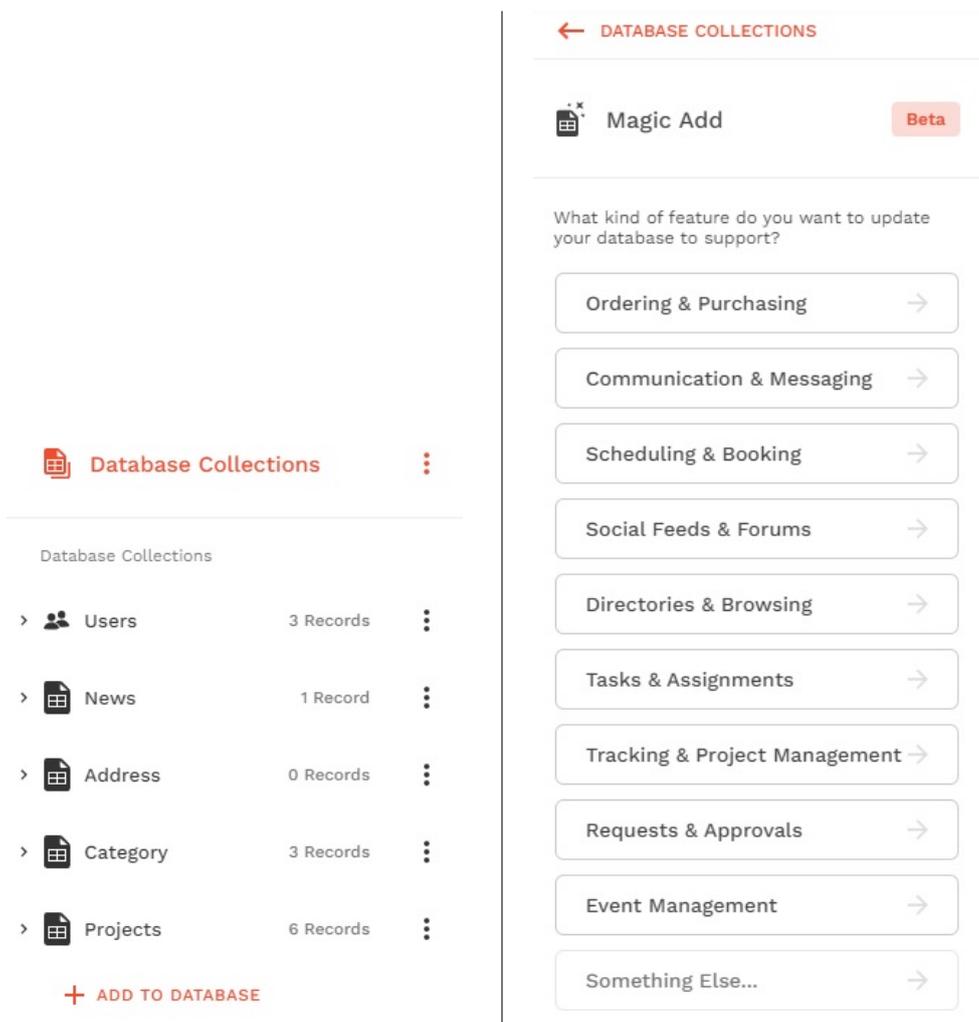
Propriedade	Descrição
Name	Nome do projeto
Desc	Descrição do projeto
Image	Imagem capa do projeto
isActive	Controle <i>boolean</i> se projeto está ativo ou não
isRecurrent	Controle <i>boolean</i> se projeto é recorrente ou não
isUrgent	Controle <i>boolean</i> se projeto é urgente ou não
MonetaryGoal	Meta financeira para o projeto
CurrentGoal	Valor financeiro já alcançado
VolunteersGoal	Meta de voluntários para o projeto
CurrentVolunteers	Quantidade de voluntários já alcançada
FinishesAt	Data de término do projeto
User	Relação com a entidade <i>usuário</i> - criador do projeto
Category	Relação com a entidade <i>categoria</i> - categoria do projeto

Fonte: Elaborado pelo Autor

4.1.2 Mecanismo para Criar Entidades na Adalo

A Figura 4.1, à esquerda, mostra a estrutura das entidades na plataforma Adalo. É possível verificar, de maneira rápida, a quantidade de registros que cada entidade possui e, também, adicionar uma nova entidade. Para adicionar uma nova entidade, você pode optar pela maneira tradicional, de atribuir um nome a essa entidade e adicionar propriedades, conforme supracitado, ou pode escolher a opção *beta* da Adalo, Figura 4.1, à direita, na qual ele tentará, através de mecanismos de inteligência artificial, supor quais são as propriedades que você quer na entidade de acordo com o tipo dela.

Figura 4.1: Coleção de Entidades e Adição de Nova Entidade

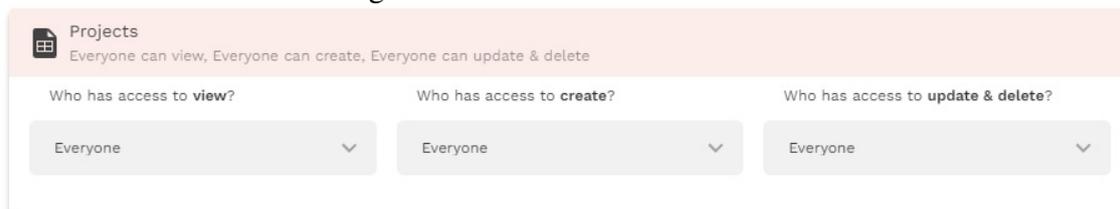


Fonte: (ADALO, 2024)

4.1.3 Permissões das Entidades

Além disso, é possível selecionar as permissões de cada entidade. Por exemplo, conforme Figura 4.2, pode-se controlar quem poderá ver, criar, atualizar ou deletar registros. Isso é um passo adicional da plataforma em direção à segurança, porém não é um controle muito forte pois é possível apenas filtrar nos usuários *logados* ou não.

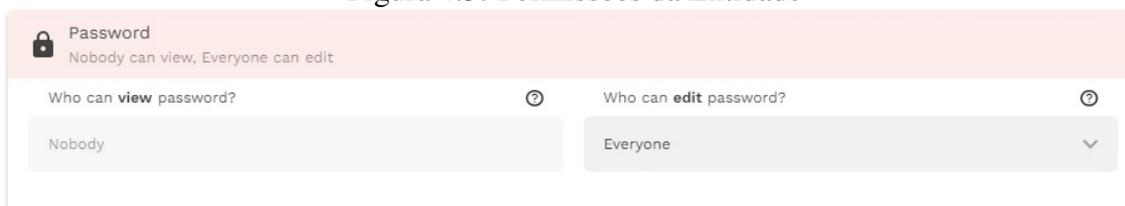
Figura 4.2: Permissões da Entidade



Fonte: (ADALO, 2024)

Porém, essas mesmas permissões podem causar uma brecha de segurança caso o desenvolvimento do aplicativo não seja feito na língua inglesa. Para realizar a restauração de senha, é necessário criar telas adicionais customizadas, pois a tela padrão da Adalo é totalmente em inglês e não é possível ser modificada. Portanto, para que seja possível alterar a senha, é necessário modificar as permissões da senha na entidade *usuário* para que qualquer pessoa possa modificar a senha, conforme Figura 4.3. É necessária atenção redobrada para não tornar possível, mediante algum caminho entre as telas do aplicativo, que qualquer pessoa consiga fazer isso, mas sim apenas a pessoa dona da conta que deseja alterar a senha.

Figura 4.3: Permissões da Entidade



Fonte: (ADALO, 2024)

4.2 Criação das Telas

Em seguida, foi iniciada a criação das telas. Em um primeiro momento foram desenvolvidas as telas de acesso inicial ao aplicativo, como tela de *login*, criar conta e as telas de restauração de senha. Posteriormente, foram desenvolvidas as telas do aplicativo,

como listagem das notícias e projetos recentes, além de telas específicas para todos os projetos e todas as notícias. Por fim, as telas referentes às configurações de conta foram desenvolvidas.

4.2.1 Tela de Acesso ao Aplicativo

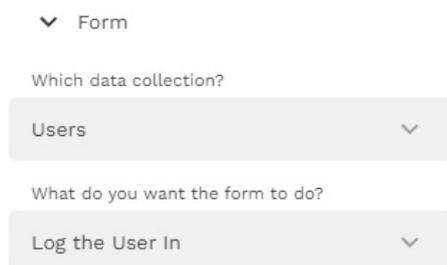
Caso o usuário não possua conta ou não esteja *logado* no aplicativo, a primeira tela que ele verá será a mostrada na Figura 4.4. O formulário presente nessa tela é escolhido na criação da tela na plataforma Adalo, conforme Figura 4.5.

Figura 4.4: Tela de *Login*



Fonte: (ADALO, 2024)

Figura 4.5: Formulário da Tela de *Login*



Fonte: (ADALO, 2024)

Caso o usuário não preencha algum dos campos, uma mensagem de erro será destacada para ele, conforme Figura 4.6, e caso o usuário tente entrar com credenciais inválidas, é possível lançar uma mensagem de erro para ele.

Figura 4.6: Campos Obrigatórios no Formulário

The screenshot shows a configuration panel for 'Form Fields'. At the top, there is a dropdown menu labeled 'Fields'. Below it, the 'Form Fields' section is visible. It contains three main components: 1. An 'Email' field with a trash icon. 2. A 'Password' field with a trash icon, which is highlighted in green. This field has a 'Label' set to 'Senha', a 'Placeholder' set to 'Digite sua senha', and a 'Required Error Text' set to 'Campo obrigatório.' with a toggle switch turned on. 3. A 'DONE' button at the bottom right.

Fonte: (ADALO, 2024)

Como uma boa prática de segurança no desenvolvimento de *software*, não é informado exatamente ao usuário se foi o *e-mail* ou senha que estavam errados, conforme visto na Figura 4.7.

Figura 4.7: Submissão do Formulário

The screenshot shows a configuration panel for 'Submit Button'. It features a dropdown menu labeled 'Submit Button'. Below it, the 'Text' section shows a button labeled 'ENTRAR'. The 'Click Actions' section is highlighted in yellow and contains: 1. A 'Log In' action with a lightning bolt icon and the text 'User'. 2. A description: 'Clicking the Submit Button on the form will log in the User'. 3. A 'Login Error Message' box containing the text 'E-mail ou senha inválidos.'. 4. A 'HIDE ADVANCED' button and a 'DONE' button at the bottom. Below this, there is a 'Link' section with a lightning bolt icon and the text 'Início', and a trash icon.

Fonte: (ADALO, 2024)

4.2.2 Tela de Criação de Conta

Caso o usuário ainda não esteja cadastrado no aplicativo, ele poderá criar uma conta, sendo direcionado para a tela mostrada na Figura 4.8. Neste momento é observado um grande ponto negativo da plataforma Adalo, a qual não permite que sejam aplicadas regras específicas para cada campo presente na entidade. Por exemplo, para a propriedade *CPF* seria interessante que fosse formatado na maneira como mostra o *placeholder*, ou seja, no formato *XXX.XXX.XXX-XX*. Todavia, a plataforma Adalo apenas permite que o campo seja considerado um texto, não podendo aplicar regras de formatação nele. Além disso, como estamos utilizando um formulário que criará um novo usuário automaticamente na entidade *usuário*, processo que é automatizado pela plataforma, não é possível utilizar serviços externos, como para validar CPF e número de telefone.

Ainda, caso optasse que a propriedade CPF fosse apenas números, não seria possível delimitar a quantidade de números máximos e mínimos que essa propriedade deveria ter. O número de CPF possui 11 dígitos numéricos, porém, neste cenário, o usuário poderia digitar apenas um número ou, ainda, mais de 11 que não seria possível validar, para verificar se está dentro das regras para um número de CPF, e impedir o cadastro. Isso não é possível atingir na Adalo, sendo necessário integração com serviços externos.

Essas validações de número de telefone, CPF e outras propriedades são facilmente realizadas no desenvolvimento de *software* tradicional. Além disso, caso não queira realizar essa verificação no código em si, poderia sem dificuldades integrar com serviços terceirizados que realizam automaticamente essa função. Ao término da criação de conta, o usuário é automaticamente direcionado para a tela inicial do aplicativo para usuários cadastrados, visto na Seção X.

4.2.3 Tela de Restauração de Senha

Caso o usuário tenha esquecido sua senha, a partir da tela inicial ele poderá solicitar uma nova senha. Conforme supracitado nas seções anteriores, como a tela padrão de restauração de senha da plataforma Adalo é totalmente em inglês e não é possível ser modificada, três telas diferentes tiveram de ser criadas para suprir essa necessidade, além de uma propriedade adicional na entidade *usuário* - *TMPPasswordResetCode*. A lógica para a restauração de senha consiste em três etapas, descritas após a Figura 4.9.

Figura 4.8: Criação de Conta

The image shows a smartphone screen with the following elements:

- Time: 12:02
- Signal, Wi-Fi, and battery icons in the top right.
- Logo: A stylized house with a sun above it, with the text "COLETIVO AUTÔNOMO MORRO DA CRUZ" below it.
- Section Header: "CRIAR CONTA" in white text on a red background.
- Form Fields:
 - E-mail:** Input field with placeholder "Digite seu e-mail".
 - Senha:** Input field with placeholder "Digite sua senha".
 - Nome Completo:** Input field with placeholder "Digite seu nome completo".
 - CPF:** Input field with placeholder "XXX.XXX.XXX-XX".
 - Número de Celular:** Partially visible input field at the bottom.

Fonte: (ADALO, 2024)

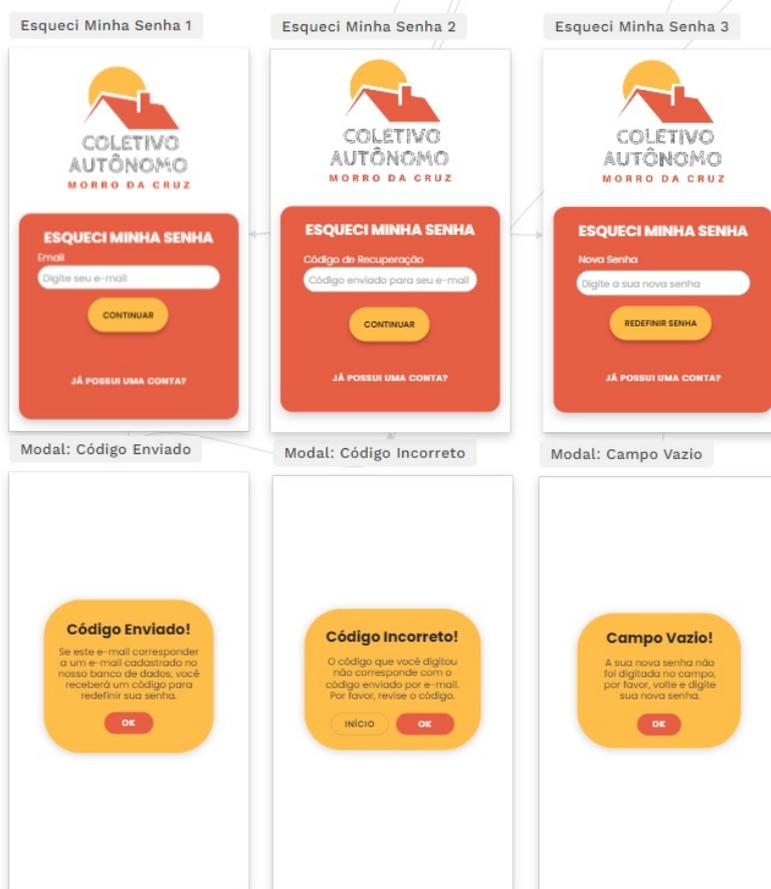
1. **Digitar e-mail:** O usuário precisará, inicialmente, digitar seu e-mail. Para que seja possível coincidir com um e-mail cadastrado no aplicativo, o formulário contém todos os e-mails cadastrados, porém isso é ocultado para o usuário. Após o usuário digitar seu e-mail e clicar em continuar, um *pop-up* aparecerá na tela informando que caso exista uma conta cadastrada com aquele e-mail, este e-mail receberá um código temporário para restauração de senha. Esse código temporário, atribuído à propriedade *TMPPasswordResetCode*, é um número aleatório entre 100.000 e 999.999, gerado automaticamente quando o usuário clica no botão. Este processo é demonstrado na Figura 4.10 abaixo.

O envio do código de restauração é realizado através de uma integração com um serviço externo, chamado Brevo¹ - responsável pelo envio automatizado de e-mail para o usuário, modalidade que só pode ser usufruída caso tenha o plano pago da plataforma Adalo. Para o desenvolvimento desta aplicação foi utilizado o período gratuito da plataforma para teste, porém essa restauração de senha não seria possível sem isso. Para envio do código, através da plataforma Brevo é enviado o seguinte pacote JSON² para o e-mail do usuário, Figura 4.11.

¹<https://www.brevo.com/pt/>

²<https://www.json.org/json-pt.html>

Figura 4.9: Telas de Restauração de Senha



Fonte: (ADALO, 2024)

2. **Verificação do Código Enviado:** Como o código aleatório gerado na etapa anterior foi salvo na propriedade temporária supracitada, pode ser verificado neste momento se o código que o usuário digitou corresponde efetivamente ao código que foi enviado para o e-mail dele. Caso não corresponder, uma mensagem de erro será mostrada para o usuário, conforme Figura 4.12.
3. **Restauração da Senha:** Na última etapa é, então, realizada a restauração da senha em si. Nessa etapa que existe, infelizmente, a brecha de segurança supracitada na Figura 4.2, pois é necessário que seja alterada a permissão para que *qualquer* pessoa possa alterar a senha. Portanto é muito importante que só seja possível chegar nesta etapa quem realmente for o proprietário da conta e tiver recebido o código correto.

Figura 4.10: Envio de Código para Restauração de Senha



Fonte: (ADALO, 2024)

4.2.4 Criação de Projetos

Para criar um novo projeto, é necessário que o usuário possua a permissão *morador* ou *admin*. Caso o usuário não possua uma dessas permissões, seria informado para ele, conforme visto na Figura 4.13. As permissões só podem ser alteradas por administradores e um usuário, ao se cadastrar no aplicativo, recebe automaticamente a permissão *visitante*.

Caso, todavia, o usuário possua as permissões necessárias, ele será direcionado para a tela mostrada na Figura 4.14. Nessa tela, o usuário deverá informar o nome e descrição do projeto. Também, poderá especificar se é um projeto recorrente, se é um projeto urgente e, também, qual é o objetivo financeiro, se existir, dele. Por fim, deverá informar, caso não seja um projeto recorrente, a data de término do projeto. O usuário ainda deverá escolher a qual categoria pertencerá esse projeto, para filtragem em outras telas.

Figura 4.11: Envio de Código para Restauração de Senha

```
{
  "sender": {
    "name": "ONG Coletivo - Morro da Cruz",
    "email": "comunicacaoativomdc@gmail.com"
  },
  "to": [
    {
      "email": "email@email.com",
      "name": "Nome do Usuário"
    }
  ],
  "subject": "Redefinição de Senha",
  "htmlContent":
  "Este e-mail foi enviado pois foi
  solicitado uma redefinição de senha para a sua conta.
  Se não foi você, por favor, ignore. Utilize esse código para redefinir sua senha: {{code}}"
}
```

Fonte: Elaborado pelo Autor

4.2.5 Criação de Notícias

A criação de notícias segue a mesma lógica existente para criar um novo projeto. Todavia, para notícias, a permissão *morador* é substituída pela permissão *editor*, pois há vários jornalistas parceiros da ONG. A Figura 4.15 demonstra a mensagem de erro exibida ao usuário caso a sua permissão de acesso não seja *editor* ou *admin*.

Tendo as permissões necessárias, o usuário é direcionado para a tela mostrada na Figura 4.16. Nela, será necessário informar o título e descrição da notícia, bem como a qual categoria ela pertence.

Figura 4.12: Envio de Código para Restauração de Senha



Fonte: (ADALO, 2024)

Figura 4.16: Criação de Notícia



Fonte: (ADALO, 2024)

Figura 4.13: Permissão Inválida para Criar Projeto



Fonte: (ADALO, 2024)

4.2.6 Tela de Perfil

Na tela de perfil será destacado o nome, CPF e telefone do usuário. Nela, é possível alterar a foto de perfil, atualizar senha, e-mail, endereço, telefone, solicitar permissões adicionais para os administradores e também sair da conta. Essas informações estão presentes na Figura 4.17. Além disso, mais abaixo na mesma tela, mostrado na Figura 4.18, há o acesso rápido para algumas informações referentes à ONG Coletivo.

4.2.7 Tela Inicial

A tela inicial, que é a primeira que o usuário verá após entrar no aplicativo, é mostrada na Figura 4.19. Nela, temos as últimas cinco notícias e projetos mais recentes. Além disso, é possível filtrar por categoria pelo menu *drop-down* no canto superior direito.

Figura 4.14: Criação de Projeto

The image shows a smartphone screen with a red background. At the top, the status bar shows the time 2:30, signal strength, Wi-Fi, and battery icons. Below the status bar is a navigation bar with a back arrow and a profile icon. The main content area is titled "CRIAR PROJETO" in white text. Below the title are three input fields: "Nome do Projeto" with the placeholder "Digite o nome do projeto", "Descrição" with the placeholder "Digite a descrição do projeto", and "Imagem do projeto" with a button labeled "Escolher Foto". At the bottom, there are two checkboxes: "Selecionar se projeto é recorrente" and "Selecionar se projeto é urgente". Below these is the text "Qual é o objetivo financeiro do projeto?".

Fonte: (ADALO, 2024)

4.2.8 Notícia em Destaque

Na Figura 4.20 é mostrado como cada notícia será mostrada no aplicativo. Nela, será possível visualizar a manchete, imagem e o texto da notícia em si e após ler toda a notícia no fim da página será possível avaliar a notícia. Dessa maneira, o público poderá opinar se a notícia é válida ou útil para a comunidade em si. Nesse momento percebemos outra funcionalidade que a plataforma Adalo não consegue entregar. Para melhorar a leitura da notícia, buscou-se deixar o texto de maneira justificada, ou seja, do início até o fim da tela. Porém, na Adalo é apenas possível alinhar à esquerda, no centro ou à direita, o que não é o ideal para textos longos. A avaliação da notícia foi possível ser realizada a partir da utilização de componentes externos da plataformas, visto na Figura 2.3, que foi desenvolvido por outros usuários e ofertada na Adalo gratuitamente.

Figura 4.15: Permissões Inválidas para Criação de Notícia



Fonte: (ADALO, 2024)

4.2.9 Projeto em Destaque

De maneira semelhante, cada projeto pode ser visto em maior destaque. Nessa tela, será visto o título, imagem e descrição de cada projeto, bem como a data de criação e término do projeto, visto na Figura 4.21, e ao final da página será visto a meta financeira do projeto e o valor já atingido até o momento, conforme Figura 4.22. Também, é possível digitar um valor de quanto gostaria de contribuir financeiramente para aquele projeto. Após digitado o valor e clicar no botão *contribuir*, será direcionado para uma conversa no WhatsApp³ com a ONG Coletivo, na qual automaticamente será informado o nome da pessoa, CPF, número de telefone e o valor que ela deseja contribuir. A doação, por requisição da ONG, será organizada nessa conversa pelo aplicativo WhatsApp.

³<https://www.whatsapp.com>

Figura 4.17: Tela de Perfil



Fonte: (ADALO, 2024)

Figura 4.18: Tela de Perfil - Informações ONG



Fonte: (ADALO, 2024)

Figura 4.19: Tela Inicial



Fonte: (ADALO, 2024)

Figura 4.20: Notícia - Destaque



Fonte: (ADALO, 2024)

Figura 4.21: Projeto - Destaque



Fonte: (ADALO, 2024)

Figura 4.22: Projeto - Destaque



Fonte: (ADALO, 2024)

5 CONCLUSÃO

Em conclusão, a experiência geral no desenvolvimento do aplicativo utilizando a plataforma Adalo apresentou desafios substanciais, impactando significativamente a percepção positiva do processo. Embora a abordagem *no-code* promova agilidade e acessibilidade, a ocorrência recorrente de *bugs*, especialmente em relação ao alinhamento de componentes na UI, emergiu como uma preocupação significativa.

A dificuldade em alcançar uma consistência visual adequada e um alinhamento preciso dos elementos da UI afetou diretamente a qualidade estética e a usabilidade do aplicativo desenvolvido. Essas limitações comprometeram a capacidade de entregar experiências de usuário coesas e visualmente atraentes, o que é essencial para o sucesso de qualquer aplicativo.

Além disso, os desafios enfrentados em relação à restauração de senha, particularmente em associação com o serviço externo da Brevo, acrescentaram uma camada adicional de complexidade ao projeto. *Bugs* relacionados a essa funcionalidade essencial não apenas prejudicaram a segurança do aplicativo, mas também impactariam fortemente na confiança dos usuários na plataforma.

Embora reconheça as vantagens da abordagem *no-code*, é imperativo destacar que a eficácia da plataforma Adalo pode ser mitigada por essas limitações. A experiência com *bugs* frequentes levanta questões sobre a confiabilidade e a robustez da plataforma para projetos mais complexos ou que exigem um alto grau de personalização.

Dessa forma, enquanto a Adalo oferece uma solução acessível e ágil para o desenvolvimento de aplicativos, é crucial ponderar cuidadosamente os *trade-offs* entre a facilidade de uso e a capacidade de superar desafios técnicos. Com a experiência obtida ao desenvolver o aplicativo para a ONG Coletivo, percebe-se que a plataforma Adalo não consegue entregar resultados adequados para uma aplicação em funcionamento, pois não oferece segurança ao desenvolvedor de que a aplicação funcionará corretamente, sem nenhum problema, no telefone do usuário. Todavia, ela entrega, com muita qualidade e de maneira totalmente gratuita, uma aplicação MVP excepcional, na qual é possível verificar e testar todas as interações possíveis que serão oferecidas ao usuário.

Este trabalho representa apenas o primeiro passo no desenvolvimento da aplicação proposta. Embora tenha sido possível alcançar um estágio inicial de construção utilizando a plataforma *no-code* Adalo, nota-se que há espaço para aprimoramentos e expansões futuras. A continuidade do projeto envolverá a correção de *bugs*, implementação de funci-

onalidades remanescentes e refinamentos na interface do usuário. Além disso, explorar-se-á a possibilidade de transição para o desenvolvimento em código, visando aproveitar ao máximo as oportunidades de personalização e escalabilidade. Com esses esforços contínuos, espera-se criar uma aplicação robusta e eficaz, atendendo às necessidades dos usuários e adaptando-se às demandas em constante evolução do mercado.

Por fim, considerações aprofundadas sobre os requisitos específicos do projeto, a complexidade da UI desejada e a integração com serviços externos são fundamentais para determinar se a plataforma atende adequadamente às necessidades de desenvolvimento. A busca contínua por melhorias e atualizações na plataforma pode ser vital para mitigar esses desafios e fortalecer sua posição como uma opção viável no cenário de desenvolvimento de aplicativos *no-code*.

REFERÊNCIAS

- ADALO. **Planos e Preços**. 2024. Acessado em 14/01/2024. Disponível em: <<https://pt-br.adalo.com/pricing>>.
- APPLE. **Apple destaca os principais apps e jogos de 2023 na App Store**. 2024. Acessado em 14/01/2024. Disponível em: <<https://www.apple.com/br/newsroom/2023/12/apple-spotlights-the-top-apps-and-games-of-2023-on-the-app-store/>>.
- CRUZ, O. C. A. M. da. **ONG Coletivo Autônomo - Morro da Cruz**. 2024. Acessado em 13/01/2024. Disponível em: <<https://coletivomdc.org/>>.
- DEMO, P. Marginalização digital: digital divide. **Boletim técnico do Senac**, v. 33, n. 2, p. 5–19, 2007.
- DEVMEDIA. **Modelagem de Dados 1: Entidades**. 2024. Acessado em 24/02/2024. Disponível em: <<https://www.devmedia.com.br/modelagem-de-dados-1-entidades/4140#:~:text=Uma%20entidade%20%C3%A9%20uma%20representa%C3%A7%C3%A3o,informa%C3%A7%C3%B5es%20que%20referenciam%20a%20entidade.>>
- GUIDINI, P. A comunicação com o mercado por meio de aplicativos: desafios e oportunidades. **Signos Do Consumo**, Universidade de São Paulo, v. 10, n. 1, p. 59–69, 2018.
- JORGENSEN, M.; SHEPPERD, M. A systematic review of software development cost estimation studies. **IEEE Transactions on software engineering**, IEEE, v. 33, n. 1, p. 33–53, 2006.
- LENZ, G. H.; FRANCISCATTO, R. Apis na era digital: Um enfoque na segurança e conscientização. **Anais do Encontro Anual de Tecnologia da Informação**, v. 12, n. 1, p. 97–97, 2023.
- ROCHA, B. M. d. et al. Custos relacionados à criação de aplicativos: Uma visão. UFSC: Florianópolis, SC, 2019.
- ROSA, R. de O.; CASAGRANDA, Y. G.; SPINELLI, F. E. A importância do marketing digital utilizando a influência do comportamento do consumidor. **Revista de tecnologia aplicada**, v. 6, n. 2, 2017.
- SANTOS, G. M. d. Estudo do uso de plataformas no-code para geração de mvps. 2022.