



UNIVERSIDADE FEDERAL DE SERGIPE  
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA  
DEPARTAMENTO DE COMPUTAÇÃO

## **Ferramentas No-Code e Low-Code em MVPs (Minimum Viable Products)**

Trabalho de Conclusão de Curso

José Henrique Bernardino Andrade



São Cristóvão – Sergipe

2024

UNIVERSIDADE FEDERAL DE SERGIPE  
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA  
DEPARTAMENTO DE COMPUTAÇÃO

José Henrique Bernardino Andrade

## **Ferramentas No-Code e Low-Code em MVPs (Minimum Viable Products)**

Trabalho de Conclusão de Curso submetido ao Departamento de Computação da Universidade Federal de Sergipe como requisito parcial para a obtenção do título de Bacharel em Ciência da Computação.

Orientador(a): Prof. Dr. Edward David Moreno Ordoñez

São Cristóvão – Sergipe

2024

# Agradecimentos

Muitas pessoas foram necessárias para chegar até aqui, impossível agradecer a todos, porém gostaria de deixar alguns agradecimentos, em especial para minha mãe que trabalhou desde sempre para me fornecer o melhor, também ao meu falecido pai que me despertou o interesse por tecnologia e me incentivou ao máximo nos estudos. Minha avó Maria Alexandrina que sempre me apoiou nos melhores e piores momentos, também as minhas tias Anivia e Luzia que me apoiaram durante todo o tempo.

Não posso esquecer dos amigos, que sempre estavam presentes nos momentos de dificuldade. Em especial os amigos que fiz na SofTeam, tenho certeza que as amizades que fiz serão para sempre. Também aos amigos que fiz no IFS e que me acompanham até hoje.

Aos professores da universidade, em especial ao meu orientador Edward que me ajudou durante todo o período desse trabalho. Também ao professor Bruno Prado que mesmo sendo “Fácil demais” me incentivou a entender e buscar mais do assunto.

*“Um ser perfeito deveria conservar sempre seu equilíbrio e nunca permitir que uma paixão ou desejo passageiro lhe perturbasse a tranquilidade.”*  
*(Mary Shelley)*

# Resumo

Nos últimos tempos, temos observado um notável aumento na proliferação de novos produtos digitais, concebidos com abordagens distintas para solucionar uma ampla gama de problemas em diversos nichos. Entretanto, à medida que o número de soluções existentes continua a crescer, a capacidade de inovar sem um investimento significativo em desenvolvimento e pesquisa torna-se cada vez mais desafiadora, dada a considerável incerteza envolvida. Conseqüentemente, muitas organizações têm optado por conduzir testes preliminares antes de adentrar no competitivo mercado, por meio da criação de versões simplificadas de seus produtos, amplamente conhecidas como MVPs (Minimum Viable Products). Embora essa abordagem possa ser relativamente simples para empresas estabelecidas com recursos financeiros substanciais, startups e empresas cujos departamentos de tecnologia estão em estágios iniciais podem encontrar desafios significativos nesse caminho.

O objetivo fundamental deste estudo é propor uma solução destinada a reduzir tanto os custos quanto o tempo associado ao desenvolvimento dessas soluções, por meio da adoção de ferramentas que exigem um mínimo ou ausência de codificação na concepção dos MVPs. Dessa forma, almejamos alcançar um cenário de mercado mais competitivo, onde indivíduos possam submeter suas ideias a testes sem a necessidade de um profundo domínio de programação, ao mesmo tempo em que reduzem o investimento financeiro inicial necessário.

O presente trabalho visa mapear ferramentas No-Code e Low-Code e testá-las na prática, destacando seus pontos fortes e fracos. Foram mapeadas 17 ferramentas de diferentes tipos e foram criadas três versões de um mesmo produto em nível de MVP, utilizando as plataformas Wix, OutSystems e Adalo, com o intuito de gerar um comparativo entre elas.

**Palavras-chave:** MVP, No-code, Low-Code, Produto, projetos.

# Abstract

In recent times, we have witnessed a remarkable increase in the proliferation of new digital products, conceived with distinct approaches to solving a wide range of problems in various niches. However, as the number of existing solutions continues to grow, the ability to innovate without significant investment in development and research becomes increasingly challenging, given the considerable uncertainty involved. Consequently, many organizations have opted to conduct preliminary tests before entering the competitive market, by creating simplified versions of their products, widely known as MVPs (Minimum Viable Products). Although this approach may be relatively straightforward for established companies with substantial financial resources, startups and companies whose technology departments are in early stages may encounter significant challenges on this path.

The primary objective of this study is to propose a solution aimed at reducing both the costs and the time associated with the development of these solutions, through the adoption of tools that require minimal or no coding in the conception of MVPs. In this way, we aim to achieve a more competitive market scenario, where individuals can submit their ideas to tests without the need for deep programming expertise, while also reducing the initial financial investment required.

The present work aims to map No-Code and Low-Code tools and test them in practice, highlighting their strengths and weaknesses. Seventeen tools of different types were mapped, and three versions of the same product at the MVP level were created using the Wix, OutSystems, and Adalo platforms, with the purpose of generating a comparison between them.

**Keywords:** MVP, No-code, Low-Code, Product.

# Lista de ilustrações

Figura 1 – Gráfico de vida de um produto . . . . .	15
Figura 2 – Ciclo da metodologia Lean . . . . .	17
Figura 3 – Plataforma de desenvolvimento OutSystems . . . . .	24
Figura 4 – Tela feita usando o tutorial . . . . .	25
Figura 5 – Lista de modelos Mendix . . . . .	25
Figura 6 – Gerenciamento da aplicação . . . . .	26
Figura 7 – Edição de site no Bubble . . . . .	27
Figura 8 – Edição de dados da Bubble . . . . .	27
Figura 9 – Criação de site com Adalo . . . . .	28
Figura 10 – Criação de dados da Adalo . . . . .	28
Figura 11 – Site dinâmico criado com Adalo . . . . .	29
Figura 12 – Uso AppSheet . . . . .	29
Figura 13 – Interface do Power Apps . . . . .	30
Figura 14 – Interface de edição . . . . .	30
Figura 15 – Interface de visualização de rede . . . . .	31
Figura 16 – Interface de edição de modelo . . . . .	31
Figura 17 – Interface de edição de modelo . . . . .	32
Figura 18 – Edição do sistema . . . . .	33
Figura 19 – Interface do wix para criação de site . . . . .	34
Figura 20 – Interface do Wordpress . . . . .	35
Figura 21 – Diagrama de casos de uso . . . . .	38
Figura 22 – Diagrama de classes . . . . .	39
Figura 23 – Diagrama de classes . . . . .	40
Figura 24 – Exemplo de como criar um campo novo em uma coleção . . . . .	42
Figura 25 – Exemplo da ferramenta de gestão dos dados . . . . .	43
Figura 26 – <i>Templates</i> disponibilizados pela plataforma . . . . .	44
Figura 27 – Home da aplicação . . . . .	45
Figura 28 – Tela de cadastro . . . . .	45
Figura 29 – Tela de detalhes da empresa . . . . .	46
Figura 30 – Abas da plataforma OutSystem . . . . .	47
Figura 31 – Aba de interface da plataforma OutSystem . . . . .	48
Figura 32 – Query usada para coletar as empresas de empresas . . . . .	48
Figura 33 – Ação criada para o cadastro de uma empresa . . . . .	49

Figura 34 – Página inicial desenvolvida utilizando OutSystems . . . . .	49
Figura 35 – Tela de detalhes desenvolvida utilizando OutSystems . . . . .	50
Figura 36 – Tela de cadastro desenvolvida utilizando OutSystems . . . . .	50
Figura 37 – Tela de gerenciamento desenvolvida utilizando OutSystems . . . . .	51
Figura 38 – Listagem de <i>templates</i> do Wix . . . . .	52
Figura 39 – Tabela de empresas criada no Wix . . . . .	52
Figura 40 – Tabela de repostas do formulário no Wix . . . . .	53
Figura 41 – Listagem de empresas desenvolvida com Wix . . . . .	53
Figura 42 – Formulário de cadastro de empresas desenvolvido com Wix . . . . .	54
Figura 43 – Tela de detalhes das empresas desenvolvida com Wix . . . . .	54
Figura 44 – Home gerado pelo chatgpt . . . . .	56
Figura 45 – Home gerado pelo chatgpt . . . . .	57

# Lista de tabelas

Tabela 1 – Ferramentas mapeadas . . . . .	20
Tabela 2 – Estudos encontrados . . . . .	22
Tabela 3 – Patentes encontrados . . . . .	23
Tabela 4 – Requisitos funcionais . . . . .	38
Tabela 5 – Requisitos não-funcionais . . . . .	39
Tabela 6 – Comparativo entre as ferramentas . . . . .	55

# Sumário

<b>1</b>	<b>Introdução</b>	<b>11</b>
1.1	Objetivos	12
1.1.1	Objetivo Geral	12
1.1.2	Objetivos Específicos	12
1.2	Metodologia	12
1.3	Estrutura do Documento	13
<b>2</b>	<b>Fundamentação Teórica</b>	<b>14</b>
2.1	Base administrativa	14
2.1.1	Ciclo de vida de um produto	14
2.1.1.1	Introdução	15
2.1.1.2	Crescimento	16
2.1.1.3	Maturidade	16
2.1.2	Declínio	16
2.1.3	A curva S	16
2.1.4	Metodologia Startup Enxuta	16
2.1.5	<i>MVP</i>	17
2.2	Base Tecnológica	18
2.2.1	Movimento de pouco de código	18
2.2.2	Diferenças entre <i>low-code</i> e <i>no-code</i>	18
2.2.3	Principais ferramentas	19
<b>3</b>	<b>Trabalhos relacionados</b>	<b>21</b>
3.1	Procedimento para Pesquisa dos Trabalhos	21
3.2	Estudos Encontrados	21
3.3	Estudo de patentes	23
3.4	Mapeamento de ferramentas	23
3.4.1	OutSystems	23
3.4.2	Mendix	25
3.4.3	Bubble	26
3.4.4	Adalo	28
3.4.5	AppSheet	29
3.4.6	Microsoft Power Apps	29
3.4.7	Appian	30
3.4.8	Zoho Creator	32
3.4.9	Wix	33

3.4.10	Wordpress	34
3.4.11	Demais soluções	35
3.5	Aprendizados retirados	35
<b>4</b>	<b>Definição do produto para teste</b>	<b>37</b>
4.1	Problema	37
4.2	Solução	37
4.2.1	Casos de uso	38
4.2.2	Requisitos	38
4.2.3	Fluxo sistema	39
4.2.4	Diagrama de classes	39
4.3	Escolha da ferramenta	40
<b>5</b>	<b>Desenvolvimento da solução</b>	<b>41</b>
5.1	Solução do MVP usando Adalo	41
5.1.1	Desenvolvimento	41
5.1.2	Telas desenvolvidas	44
5.1.3	Conclusão	46
5.2	Solução do MVP usando Outsystem	46
5.2.1	Desenvolvimento	47
5.2.2	Telas desenvolvidas	49
5.2.3	Conclusão	51
5.3	Solução do MVP usando Wix	51
5.3.1	Desenvolvimento	52
5.3.2	Telas desenvolvidas	53
5.3.3	Conclusão	54
5.4	Comparação das ferramentas analisadas	55
5.5	Opção alternativa	56
<b>6</b>	<b>Considerações Finais e Trabalhos Futuros</b>	<b>58</b>
	<b>Referências</b>	<b>60</b>

# 1

## Introdução

Em fevereiro de 2001, emergiu o Manifesto Ágil, marcando o advento de uma abordagem revolucionária na engenharia de *software*, com ênfase na entrega ágil e na geração de valor. A evolução subsequente da engenharia de *software* trouxe consigo a resolução de problemas cada vez mais complexos, impulsionada pelo crescimento de *frameworks* destinados a otimizar o processo de desenvolvimento. No entanto, para atender às demandas do mercado que muitas vezes exige celeridade, o movimento de uso de pouco código (*low-Code*) tem surgido desde 1980, contudo o termo *low-Code* surgiu apenas em 2014 (RUSCIO et al., 2022).

As ferramentas de baixo código (*low-Code*) destacam-se ao viabilizar a criação de aplicações com poucas linhas de código. Um exemplo notável é o WordPress, amplamente conhecido por capacitar a criação de *websites* com relativa simplicidade. Segundo informações divulgadas pela própria plataforma, aproximadamente 43% da *internet* é alimentada por meio do WordPress. Nessa mesma trajetória, surgiram as ferramentas *no-code*, que dispensam a escrita de código, permitindo a criação de aplicações por meio de interações visuais. Tais inovações conferiram uma nova perspectiva ao panorama da engenharia de *software*, proporcionando oportunidades para o desenvolvimento ágil de aplicações, otimizando tanto o tempo quanto os recursos empregados.

Nos últimos anos, o mercado presenciou um aumento substancial na oferta de produtos digitais, acelerado pela pandemia da COVID-19. Conforme reportado pela ABES (2023) no relatório de 2023, o Brasil direcionou 11.858 milhões de dólares em investimentos de *software* no ano de 2022, posicionando-se como o 12º maior investidor mundial em *software*.

A despeito de ser um produto digital, este segue um ciclo de vida estruturado, em consonância com o conceito de Levitt (1965) apresentado na Harvard Business Review de 1965. Este ciclo compreende as fases de introdução, crescimento, maturidade e declínio, sendo igualmente aplicável aos produtos digitais.

Nos estágios iniciais, a pesquisa de mercado, a análise de stakeholders e a identificação

das necessidades dos clientes são vitais. Mesmo após uma pesquisa minuciosa, não se tem um conhecimento pleno sobre o cliente. Durante a fase de introdução, é comum desenvolver um Produto Mínimo Viável (*MVP*) como meio de validar a proposta antes da escala.

O *MVP* deve ser ágil e econômico, visando validar a concepção do produto e permitir ajustes rápidos caso necessário. Contudo, a alocação de recursos para a criação de um novo produto pode ser onerosa e demorada para algumas organizações. Nesse contexto, as ferramentas *no-code* e *low-Code* apresentam-se como soluções promissoras.

Diante dessas considerações, o estudo proposto abrange um exame comparativo das ferramentas *no-code* e *low-Code* para o lançamento de *MVPs*, proporcionando uma abertura estratégica para a introdução de produtos de baixo custo e esforço, focalizando primordialmente o consumidor final e minimizando a ênfase na tecnologia subjacente.

## 1.1 Objetivos

### 1.1.1 Objetivo Geral

Realizar um mapeamento das ferramentas *no-code* e *low-Code* disponíveis no mercado e levantar seus pontos forte e fracos e ao final desenvolver o *MVP* de uma solução para divulgação de empresas usando uma ferramenta *low-Code* (OutSystems), *no-code* (Adalo) e com uma alta popularidade (Wix).

### 1.1.2 Objetivos Específicos

- Mapear as ferramentas *no-code* e *low-Code* disponíveis no mercado;
- Analisar as ferramentas com base nas necessidades de um projeto em *MVP*;
- Realizar comparativo das ferramentas;
- Construir um exemplo de *MVP* com as ferramentas que mais se destacaram.

## 1.2 Metodologia

Este trabalho tem carácter comparativo. Inicialmente foi realizado uma pesquisa na literatura para entendimento das necessidades de um produto em *MVP*, criando critérios para comparação das ferramentas, mapeamento das ferramentas que se adéquem ao conceito de *no-code* ou *low-Code* e seus testes. Ao final foi realizado a construção um *MVP* de um produto de divulgação de empresas usando as ferramentas Adalo, OutSystems e Wix. Em ordem esses foram os passos realizados:

- Pesquisas em bases teóricas sobre ciclo de vida de produtos;

- Mapeamento das ferramentas;
- Teste e análise das ferramentas;
- Modelagem do produto de divulgação de empresas;
- Construção do *MVP* com as ferramentas Adalo, OutSystems e Wix.

### 1.3 Estrutura do Documento

O conteúdo deste trabalho é dividido da seguinte forma:

- Capítulo 1 - Introdução: Apresentação geral do trabalho;
- Capítulo 2 - Fundamentação Teórica: Detalhamento de toda teoria necessária para compreensão dos assuntos tratados;
- Capítulo 3 - Trabalhos relacionados: Procura em bases de artigos, patentes e análise das ferramentas;
- Capítulo 4 - Definição do produto para teste: Modelagem do produto a ser testado e escolha das ferramentas;
- Capítulo 5 -Desenvolvimento da solução: Amostragem do software desenvolvido e detalhamento do seu desenvolvimento;
- Capítulo 6 - Considerações finais e Trabalhos Futuros: Considerações finais em relação ao trabalho e possíveis trabalhos relacionados.

# 2

## Fundamentação Teórica

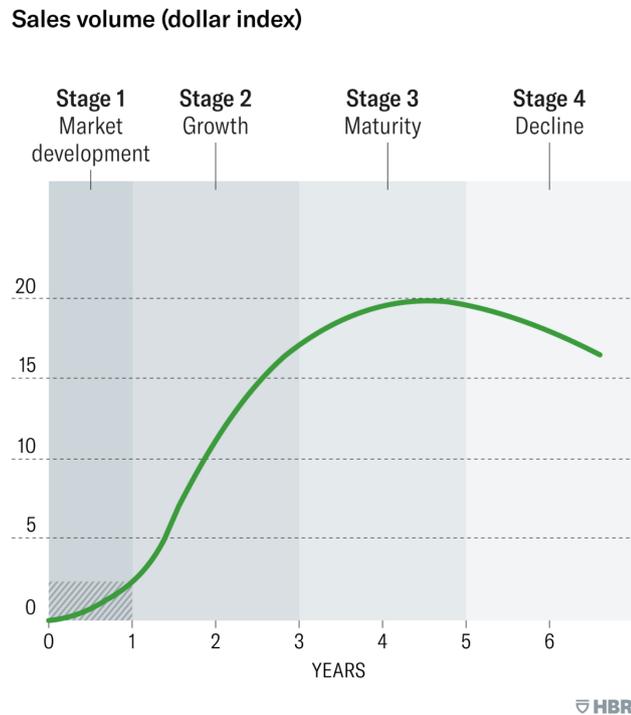
Este estudo possui uma forte relação com duas áreas distintas, sendo ela o conhecimento administrativo e o conhecimento técnico de tecnologia e engenharia de *software*, por isso nesse capítulo o dividiremos em dois blocos indistintos.

### 2.1 Base administrativa

#### 2.1.1 Ciclo de vida de um produto

O ciclo de vida do produto é um conceito fundamental na gestão de produtos e marketing, e sua compreensão é crucial para o sucesso de qualquer produto ou serviço no mercado. Theodore Levitt, renomado economista e professor de Harvard Business School, desempenhou um papel significativo na formulação e difusão desse conceito. Seu primeiro estudo foi publicado na *Harvard Business Review* de 1965, no gráfico abaixo conseguimos ver a relação entre número de vendas no tempo:

Figura 1 – Gráfico de vida de um produto

**Exhibit I  
Product Life Cycle: Entire Industry**

Fonte: [Levitt \(1965\)](#)

Podemos desdobrar essas etapas como partes fundamentais do ciclo de vida do produto, nomeadamente: a fase de Introdução, a fase de Crescimento, a fase de Maturidade e, por último, a fase de Declínio. Vamos agora analisar detalhadamente as características distintivas de cada uma dessas etapas.

### 2.1.1.1 Introdução

Na fase de Introdução, observamos a gênese da demanda pelo produto, caracterizada pela falta de consciência geral acerca de sua necessidade. Este estágio é frequentemente permeado pela incerteza, exigindo substanciais investimentos em pesquisa, desenvolvimento e *marketing*. A duração dessa etapa pode variar, notadamente em função da complexidade do produto. Em casos que envolvem inovações absolutas, é comum a necessidade de estimular a demanda e demonstrar a superioridade do produto em relação a substitutos. Em todo caso, esta fase se revela particularmente desafiadora e dispendiosa, podendo rapidamente se tornar um período de dificuldades econômicas.

### 2.1.1.2 Crescimento

Na etapa do “Boom” que sucede a fase de Introdução, observamos um notório incremento nas vendas e na confiança dos consumidores, caso o produto seja bem-sucedido. Com o aumento significativo de sua popularidade, é comum que concorrentes pré-existentes reajam, procurando preservar sua presença no mercado. Simultaneamente, novos competidores tendem a surgir, adotando abordagens que oscilam entre a emulação direta e a introdução de melhorias em soluções já estabelecidas, que agora apresentam menor grau de incerteza.

### 2.1.1.3 Maturidade

Nesta fase, evidencia-se uma desaceleração nas vendas, sinalizando a saturação do mercado como principal indicativo. A concorrência por meio de preços se intensifica, demandando a implementação de estratégias de *marketing* inovadoras. Neste contexto, a eficiência se configura como elemento fundamental para a manutenção da competitividade.

## 2.1.2 Declínio

Em algumas abordagens literárias, esta fase pode ser denominada como o declínio ou “morte” de um produto. Todavia, é importante ressaltar que não implica necessariamente o término, pois alguns concorrentes conseguem manter sua presença, embora essa tarefa demande uma considerável preparação e organização. A redução das vendas frequentemente se revela fatal para uma parcela substancial dos produtos, resultando no encerramento da maioria deles.

### 2.1.3 A curva S

Há uma abordagem de solução para produtos que alcançaram a fase de maturidade, conhecida como a “Curva em S” popularizada por Everett M. Rogers [Johnson \(2022\)](#). A teoria da “Curva em S” sugere uma remodelação do produto durante o estágio de maturidade, o que, embora exija um esforço considerável e custos de desenvolvimento, pode revitalizar o crescimento. Um exemplo amplamente reconhecido dessa estratégia é observado na indústria da televisão, que constantemente passa por renovações para evitar a queda no estágio de declínio.

## 2.1.4 Metodologia Startup Enxuta

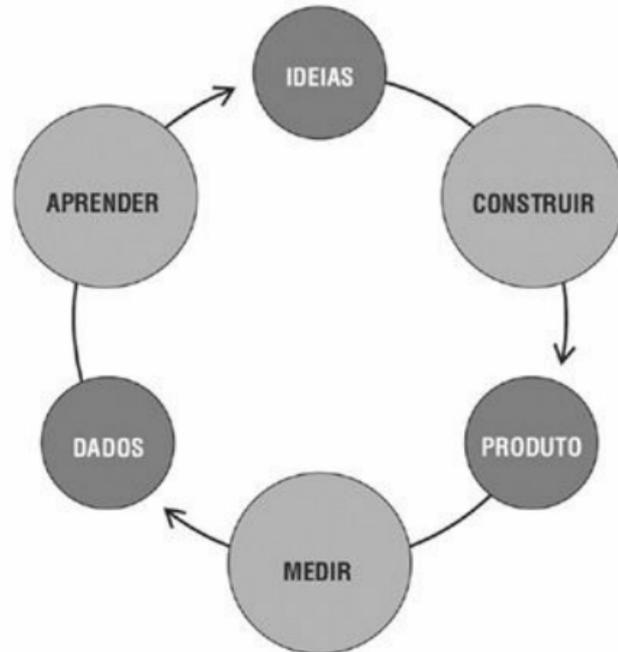
A metodologia Lean Startup foi criada por [Ries \(2012\)](#), onde consiste em uma alternativa da administração clássica para startups. Segundo o próprio Eric temos que:

Uma startup é uma instituição humana projetada para criar novos produtos e serviços sob condições de extrema incerteza.

O Eric definiu um processo cíclico focado no aprendizado constante sobre o problema que está lidando. Como podemos ver na imagem abaixo:

Figura 2 – Ciclo da metodologia Lean

### CICLO DE FEEDBACK CONSTRUIR-MEDIR-APRENDER



**Minimizar o tempo TOTAL através do ciclo**

Fonte: [Ries \(2012\)](#)

Como podemos observar, o modelo Startup Enxuta é um modelo incremental onde o foco é sempre o aprendizado, transformando as ideias em produto e medindo seus resultados para adquirir mais conhecimento. Esse método exige uma preparação da equipe para realizar pequenas entregas o mais rápido para poderem ser gerados dados, é comumente criado Produto Mínimo Viável.

Esse modelo se tornou comum não só em pequenas startups, mas também em grandes empresas, sendo responsável pelo crescimento acelerado de diversos produtos. O ganho que esse modelo traz é o foco aprender, e usar o aprendizado para testar novamente em ambientes de grande incerteza, é essencial

#### 2.1.5 MVP

O MVP ou Produto Mínimo viável é um produto que possui apenas o básico para validar o produto, então são eliminados fluxos desnecessários e muitas vezes é feita uma versão mais simples.

Existem diversos casos de sucesso de *MVPs*, um citado pelo próprio autor do livro (RIES, 2012) é o da “Zappos”, onde para verificar se as pessoas comprariam calçados *online*, tirou fotos de calçados de lojas físicas e aguardava alguém comprar o produto *online*, quando a compra era concluída, ele retornava na loja e comprava o produto. Não é difícil ver nesse caso que esse produto não é escalável e seriam necessários um esforço operacional enorme, porém o foco da Zappos naquele momento era apenas verificar se as pessoas comprariam sapatos pela *internet*.

## 2.2 Base Tecnológica

### 2.2.1 Movimento de pouco de código

Não é possível indicar com exatidão o início do movimento *low-code*, segundo o Ruscio et al. (2022) o movimento de usar cada vez menos código é popular desde os anos 80 com ferramentas CASE (Computer-Aided Software Engineering), RAD (Rapid Application Development) e mais recentemente MDE (model-driven engineering). O termo foi evidenciado pelo Richardson e Rymer (2016a) decifinado uma ferramenta *low-code* como:

Plataformas que permitem a entrega rápida de aplicativos empresariais com um mínimo de programação manual e um investimento mínimo inicial em configuração, treinamento e implantação.

Desde essa época, tivemos um avanço nessa área e com a introdução do *no-code* e criação de mais ferramentas *low-code*.

### 2.2.2 Diferenças entre *low-code* e *no-code*

As ferramentas *no-code* e *low-code* hoje são famosas e diversas empresas usam em seu dia-a-dia, um exemplo muito famosos é o WordPress, onde é possível a construção de sites de forma rápida, fácil e gratuita, porém o que de fato são essas plataformas? Como o nome já sugere, são ferramentas que trazem uma alternativa ao desenvolvimento tradicional, trazendo como principal diferença o uso de pouco desenvolvimento de código para criação de software ou no caso do *no-code*, nenhum código é necessário.

O principal ganho dessas ferramentas é o baixo nível de conhecimento de programação exigido para criação e alteração da aplicação, e também a agilidade para implementação das mesmas. Essas ferramentas possuem também facilidades como modelos de interface pre-prontas, estrutura de dados modelada, etc. Com essas ferramentas as etapas iniciais de um projeto de software podem ser puladas como as definições de arquitetura e modelo de classes, trazendo desde o seu planejamento que fica por parte da escolha da ferramenta mais adequada para seu objetivo.

Por mais que essas plataformas tenham diversos benefícios, ainda possuem problemas relacionados a segurança, desempenho e adaptabilidade. Causando casos comuns de invasões nos sistemas usando falhas de segurança ou se aproveitando das limitações para aplicar ataques como DDOS, também conhecido como ataque de negação de serviço, sendo importante saber escolher bem a plataforma que irá implementar seus projetos, escolhendo se baseando nos riscos, custos e necessidades específicas.

### 2.2.3 Principais ferramentas

A fim de mapear possíveis ferramentas para serem alvos deste presente estudo foi feito uma pesquisa sobre quais ferramentas disponíveis e qual sua descrição; para otimizar os resultados foram usadas inteligências artificiais para realizar a busca por essas ferramentas, o prompt usado foi o seguinte: “Você agora é um pesquisados e precisa realizar um mapeamento de ferramentas *low-code* e *no-code*, retorne em formato de tabela com as seguintes colunas: ferramentas, fornecedor e descrição”. As inteligências artificiais usadas foram: *ChatGPT*, *Copilot* e *Gemini*, as informações extraídas passaram por uma validação de veracidade, para aumentar o leque de ferramentas mapeadas foi usado também ferramentas de buscas tradicionais como *Google* e *Bing*. Os resultados estão na Tabela 1.

Tabela 1 – Ferramentas mapeadas.

Nome	<i>low-code</i>	<i>no-code</i>	Preço	Link
OutSystems	Sim	Não	Grátis para um aplicativo.	< <a href="https://www.outsystems.com/pt-br/">https://www.outsystems.com/pt-br/</a> >
Mendix	Sim	Não	Grátis.	< <a href="https://www.mendix.com/">https://www.mendix.com/</a> >
Bubble	Não	Sim	Teste grátis, porém o recomendado para <i>MVPs</i> é 29 dólares por mês.	< <a href="https://bubble.io/">https://bubble.io/</a> >
Adalo	Não	Sim	Teste grátis, mas para publicar é necessário 36 dólares por mês.	< <a href="https://pt-br.adalo.com/">https://pt-br.adalo.com/</a> >
AppSheet	Sim	Sim	Grátis.	< <a href="https://www.appsheet.com">https://www.appsheet.com</a> >
Microsoft Power Apps	Sim	Não	Teste grátis, depois de 30 dias custa 128 reais por mês a cada usuário.	< <a href="https://powerapps.microsoft.com/pt-br/">https://powerapps.microsoft.com/pt-br/</a> >
Appian	Sim	Não	Não informado no site.	< <a href="https://appian.com/">https://appian.com/</a> >
Zoho Creator	Sim	Não	8 dólares por mês a cada usuário.	< <a href="https://www.zoho.com/pt-br/creator/">https://www.zoho.com/pt-br/creator/</a> >
Wix	Não	Sim	Grátis, plano premium a partir de 13 reais.	< <a href="https://pt.wix.com/">https://pt.wix.com/</a> >
WordPress	Sim	Sim	Gratuito e código aberto.	< <a href="https://br.wordpress.org/">https://br.wordpress.org/</a> >
Airtable	Não	Sim	Gratuito, plano premium a partir de 10 dólares mensais por usuário.	< <a href="https://airtable.com/">https://airtable.com/</a> >
Glide	Não	Sim	Gratuito, plano premium a partir de 25 dólares mensais.	< <a href="https://www.glideapps.com/">https://www.glideapps.com/</a> >
Quick Base	Não	Sim	Teste grátis, depois de 30 dias custa a partir 35 dólares por mês a cada usuário.	< <a href="https://www.quickbase.com/">https://www.quickbase.com/</a> >
Caspio	Sim	Sim	194,40 reais por mês.	< <a href="https://www.caspio.com/">https://www.caspio.com/</a> >
Betty Blocks	Sim	Sim	Não informado no site.	< <a href="https://www.bettyblocks.com/">https://www.bettyblocks.com/</a> >
Retool	Sim	Sim	Gratuito, plano premium a partir de 10 dólares mais 5 dólares por usuário.	< <a href="https://retool.com/">https://retool.com/</a> >
Stacker	Sim	Não	A partir de 59 dólares por mês.	< <a href="https://www.stackerhq.com/">https://www.stackerhq.com/</a> >

Fonte: Autor (2023).

Com esse mapeamento foi possível mapear as possíveis ferramentas disponíveis, algumas ferramentas como WordPress são bastante difundidas pela *internet*, porém muitas novas apareceram na lista, nos capítulos posteriores iremos separar aquelas que possuem maior aderência com o objetivo e testá-las.

# 3

## Trabalhos relacionados

Como o foco do trabalho é explorar as ferramentas do tipo *no-code* e *low-code* para desenvolvimento de *MVP* usados, foram usadas ferramentas de buscas para procurar por artigos que envolvessem assuntos relacionado e que pudesse gerar *insights* para o presente trabalho.

### 3.1 Procedimento para Pesquisa dos Trabalhos

Para a busca foram utilizados quatro *sites*, sendo eles: *Scopus*, *Web of Science* e *Google Scholar*, com o objetivo de procurar artigos e revistas que falassem sobre o tema deste trabalho. Para usar essas ferramentas foi criada uma *string* de busca, sendo modificada apenas para se encaixar na sintaxe de cada site. A *string* usada foi a seguinte: (“*minimum viable product*” or “*MVP*”) and (“*no code*” OR *no-code* OR “*low code*” OR *low-code*).

### 3.2 Estudos Encontrados

Durante a busca foi realizado um trabalho de análise de cada um dos artigos e revistas encontradas visando avaliar se faria sentido para o estudo e também remoção de duplicatas. No total foram encontrados quatro artigos e uma revista. Na Tabela 2 podemos ver os resultados encontrados.

Tabela 2 – Estudos encontrados.

<b>Título</b>	<b>Referência</b>
A no-code App Design Platform for Mobile Health Research: Development and Usability Study	(S LA H; RE, 2022)
The Development of a Hypertension Prevention and Financial-Incentive mHealth Program Using a no-code Mobile App Builder: Development and Usability Study	(A; S, 2023)
Desenvolvimento de um aplicativo em plataforma low code integrado a business intelligence para vistorias de unidades residenciais	(LEAL, 2023)
Process Optimization in the Steel Industry using Machine Learning adopting an Artificial Intelligence Low Code Platform	(MATEO; REDCHUK, 2022)
Vendor Landscape: The Fractured, Fertile Terrain Of Low-Code Application Platforms	(RICHARDSON; RYMER, 2016b)

Fonte: Autor (2023).

No artigo de S La H e RE (2022), temos o detalhamento do desenvolvimento de uma ferramenta *no-code* para criação de aplicativos de saúde (mHealth) sendo a primeira plataforma focada nesse tipo de contexto, tendo como objetivo criação de aplicativos para monitorar métricas de saúde, criação de conteúdo informativo e lançamento de pesquisas para usuários. O desenvolvimento foi dividido em quatro fases, sendo elas levantamento de requisitos, desenvolvimento da plataforma, teste de usabilidade e correção dos pontos levantados no teste. Para desenvolver a plataforma foram destinados 7 *sprints* (com tempo de 4 semanas cada sprint), totalizando 7 meses. A plataforma desenvolvida se chama Pathverse e está disponível no link: <<https://pathverse.ca>>.

No artigo de A e S (2023) vimos na prática como plataforma Pathverse, o objetivo do aplicativo criado era ajudar na prevenção de hipertensão. A meta era criar o aplicativo em 8 semanas, atingindo essa marca e realizando ao final um teste de usabilidade para garantir que o aplicativo conseguiu atingir os padrões de qualidade. O aplicativo foi totalmente focado em trazer lições para os usuários, o artigo descreve como um processo totalmente focado na pesquisa e ideação, sendo o desenvolvimento a parte menos focada, mostrando que o aplicativo desenvolvido no artigo S La H e RE (2022) atingiu seus objetivos.

No artigo de Leal (2023) temos uma visão diferente de implementação de ferramentas foco, sendo usadas para trazer inteligência para o processo de vistoria de construções, nesse caso o foco é na agilidade e assertividade da solução. A ferramenta escolhidas foi a “PowerApps” integrado com outras ferramentas da *Microsoft* como: *Sharepoint*, *Power Automate* e *Power BI*. A autora escolheu utilizar da conexão das ferramentas da *Microsoft* para criar o aplicativo que conseguisse satisfazer as necessidades do seu contexto.

O artigo de Mateo e Redchuk (2022) fala sobre a implementação de uma ferramenta *low-code* na indústria de metal, focada melhoria operacional, como citado no artigo detecção de falhas e análise de dados usando IA, eles usaram o método de Lean Startup para criar os modelos de forma rápida e testar se funcionava, sem demandar muito esforço (tanto de desenvolvimento quando operacionais). Esse foi o único estudo que fala sobre Internet das Coisas (IoT ou IIoT,

para uso industrial), porém cita a importância de quebrar a implementação em pequenas partes e lançar um *MVP* antes de gastar muito tempo com o desenvolvimento.

O último estudo é uma revista (RICHARDSON; RYMER, 2016b) falando sobre a possibilidade que ferramentas *low-code* podem gerar e quais fornecedores estavam disponíveis na época, separando por segmentos ao final.

### 3.3 Estudo de patentes

A fim de verificar quais patentes estão relacionadas ao universo de *low-code* e *no-code* foi realizada uma busca na ferramenta *INPI*, entretanto não foram encontradas nenhuma patente ou registro de software *relacionado*. Para não deixar essa seção vazia, foi realizada uma pesquisa em patentes internacionais usando a ferramenta *WIPO*, porém devido ao elevado número de resultados (1800 resultados), foram selecionados apenas dois que continham maior compatibilidade com estudo.

Tabela 3 – Patentes encontrados.

<b>Título</b>	<b>Registro</b>
LOW-CODE AND NO-CODE CUSTOMIZATION USER INTERFACE COMPONENTS	US20230146421
COMPUTERIZED SYSTEM AND METHOD FOR A DISTRIBUTED LOW-CODE / NO-CODE COMPUTING ENVIRONMENT	WO2021096953

Fonte: Autor (2023).

Não foram encontrados patentes sobre as ferramentas em si, apenas sobre ferramentas para auxiliar estas plataformas. Nas patentes encontradas temos uma forma de hospedagem para essas ferramentas e também uma forma de processar as customizações em UI feitas nessas interfaces. Os documentos são bem técnicos, então não são de grande relevância para esse estudo.

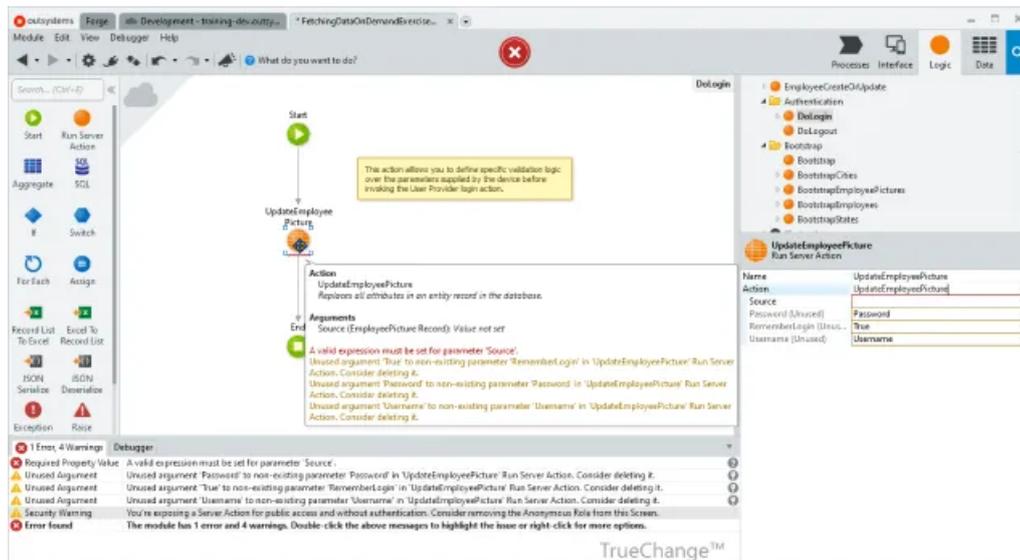
### 3.4 Mapeamento de ferramentas

Considerando as ferramentas mapeadas na sessão 2.2.3 agora vamos analisar como cada uma das ferramentas se apresenta para o mercado, esta análise serve apenas como filtro inicial, para ser possível entender qual o foco de cada uma e quais apresentam maior sentido com o alvo.

#### 3.4.1 OutSystems

A plataforma OutSystems tem um foco em aplicações rápidas, possui uma inteligência artificial que auxilia bastante no desenvolvimento, além disso, reforça sua segurança e suporte para aplicações mais robustas.

Figura 3 – Plataforma de desenvolvimento OutSystems



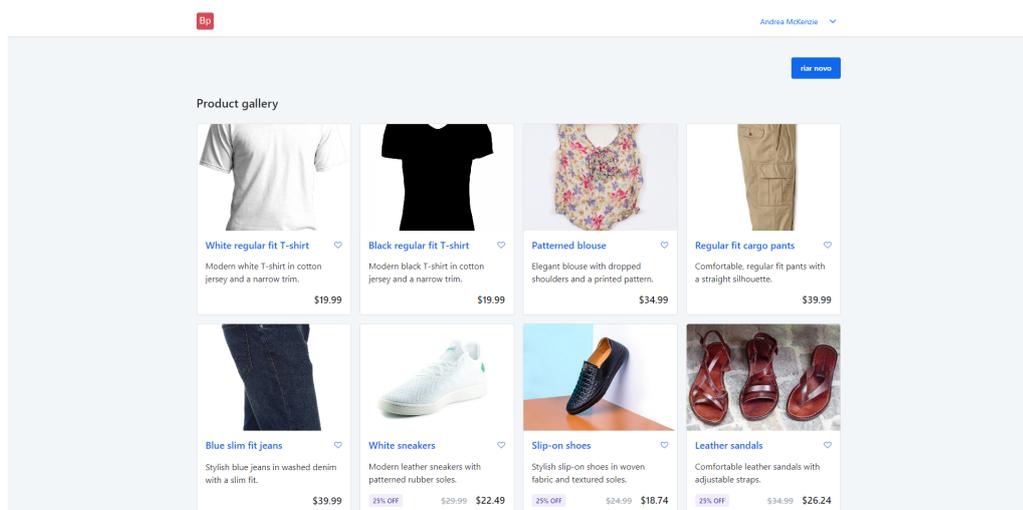
Fonte: [outsystem \(2023\)](#)

A plataforma foi submetida a um teste gratuito, com acesso total às suas funcionalidades. O objetivo primordial desse teste consistiu na avaliação das principais funcionalidades, restrições e na compreensão da curva de aprendizado para sua utilização.

O OutSystems disponibiliza um ambiente de desenvolvimento gratuito, eliminando a necessidade de custos suplementares com ambiente e domínio, porém para qualquer operação de criação ou edição do site, a instalação de um *software* específico é requerida.

Inicialmente, a plataforma oferece um guia introdutório sobre a criação da tela inicial, apresentado na Figura 4. No entanto, a sugestão subsequente envolve a integração com um banco de dados próprio. Os bancos de dados suportados abrangem desde opções mais simples, como o Sharepoint, até sistemas mais complexos, como *SQLs* e *NonSql*.

Figura 4 – Tela feita usando o tutorial



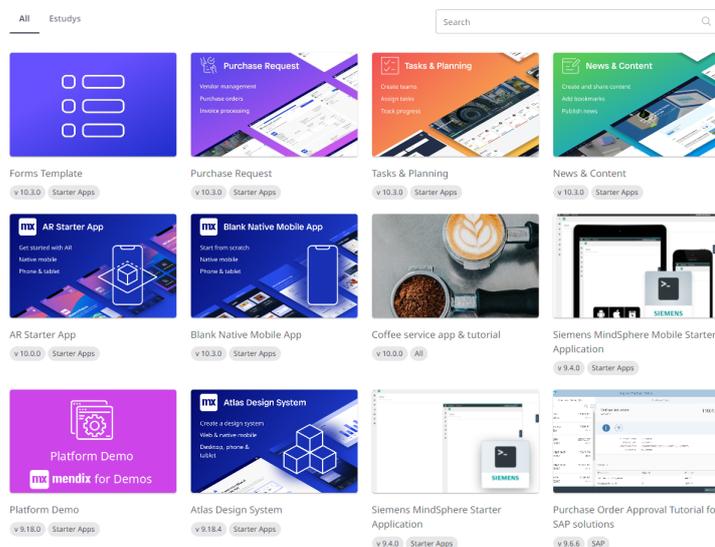
Fonte: outsystem (2023)

A plataforma proporciona um ambiente propício ao desenvolvimento de novos produtos, embora demande conhecimentos básicos em programação por causa da sua comunicação focada em pessoas de desenvolvimento.

### 3.4.2 Mendix

A Mendix foca bastante em sua escalabilidade, facilidade e agilidade de desenvolvimento, possui uma biblioteca de componentes, como é possível ver na Figura 5, o que facilita bastante na hora de desenvolver um novo sistema.

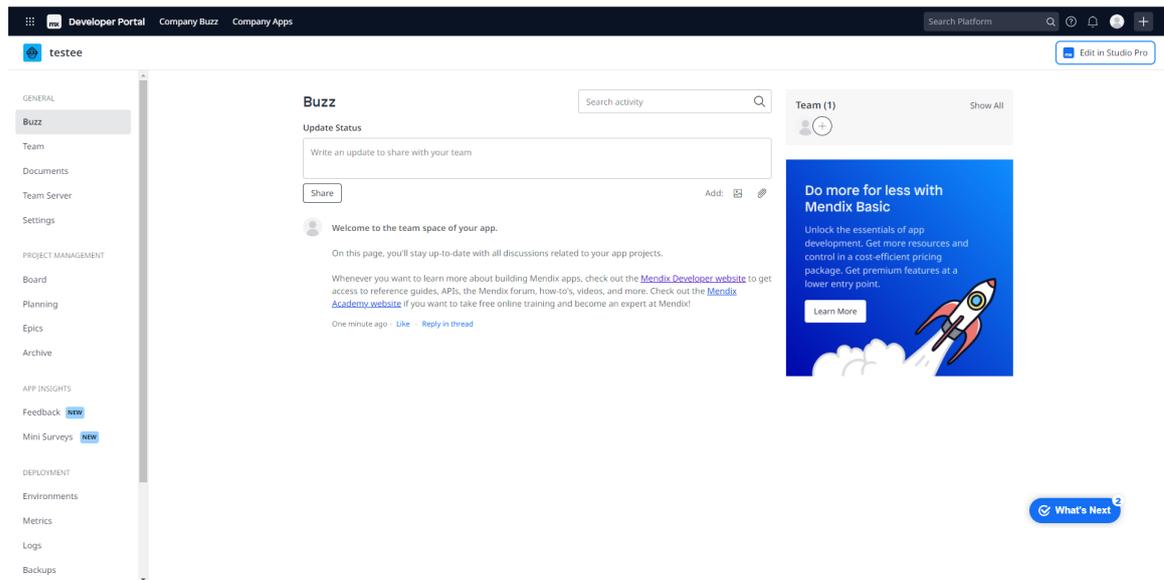
Figura 5 – Lista de modelos Mendix



Fonte: Autor (2023)

A plataforma possui diversas ferramentas de gerenciamento, cobrindo desde o desenvolvimento da ferramenta até o gerenciamento e acompanhamento de métricas, como está na figura 6. Considerando uma equipe pequena e com um certo nível de maturidade, é possível utilizar apenas essa plataforma para controle do produto, já que é possível adicionar quantos usuários achar conveniente sem custos adicionais.

Figura 6 – Gerenciamento da aplicação



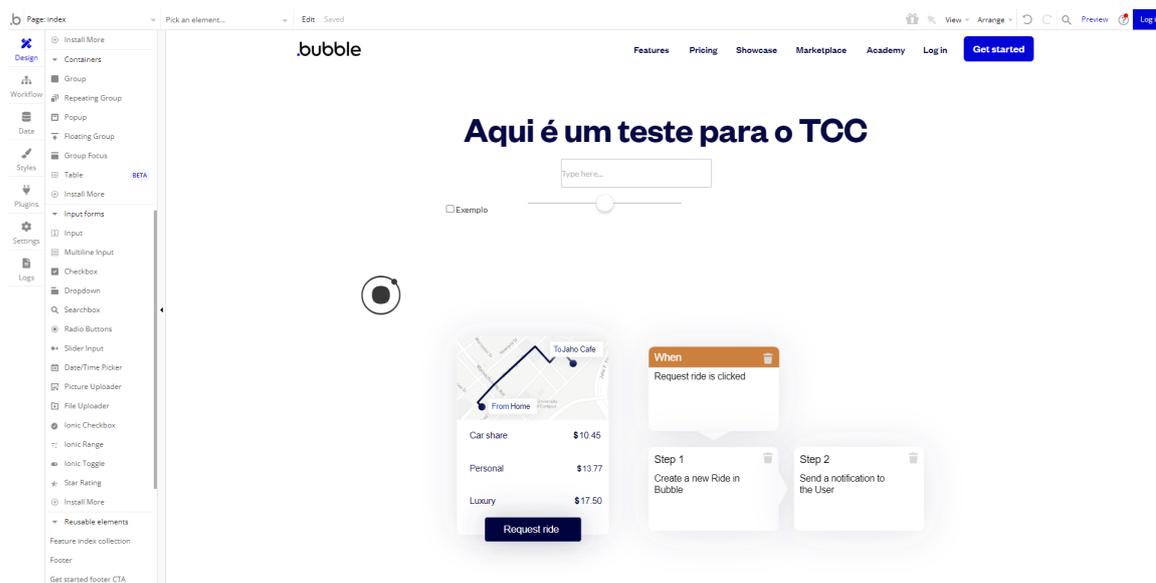
Fonte: Autor (2023)

Ao testar o Mendix foi encontrado uma enorme barreira, mesmo dizendo que possui suporte para *no-code* e até *low-code* a plataforma incentiva a baixar um *software* que apenas serve como *IDE* para os produtos criados nela. O sistema possui uma versão de edição de *UI*, porém não foi possível testar a mesma, sendo assim essa ferramenta está descartada do estudo.

### 3.4.3 Bubble

O Bubble surpreende pela sua facilidade de testar a plataforma, apenas clicando em uma parte do site sem a necessidade de login ou qualquer validação é possível testar como funciona a edição, na Figura 7 é possível ver um exemplo de teste rápido realizado.

Figura 7 – Edição de site no Bubble

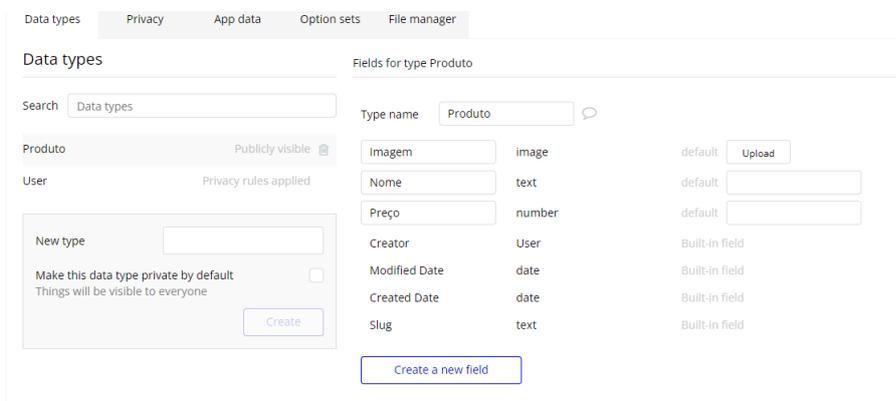


Fonte: Autor (2023)

A ferramenta é focada em sites e apresenta diversas opções de componentes de interface já prontos, facilitando na criação de um novo site. A ferramenta também conta com a criação de fluxos, bancos de dados e gerenciamento de biblioteca de design.

A ferramenta de criação e gerenciamento do banco de dados é um ponto forte dessa ferramenta, sendo possível criar tabelas com relacionamentos e tipos de dados diversos, desde os tipos básicos como texto e número, até mais complexos como arquivos e imagens, na Figura 8 podemos ver um pouco dessa ferramenta.

Figura 8 – Edição de dados da Bubble

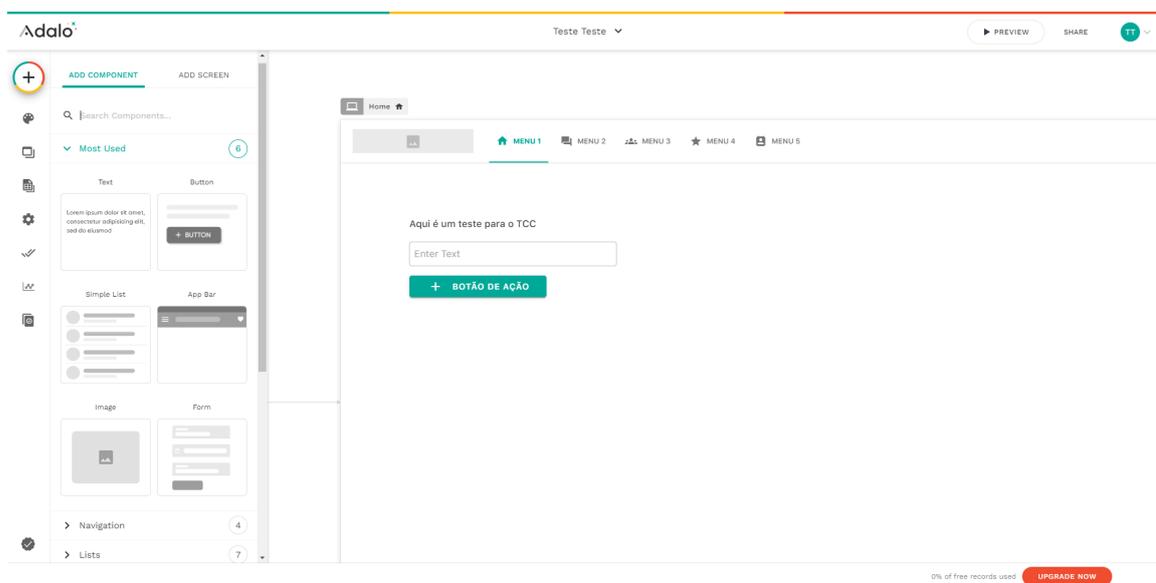


Fonte: Autor (2023)

### 3.4.4 Adalo

O Adalo é uma simples plataforma para criação de aplicativos e sites, tendo suporte para criação de várias telas interligadas. Tem suporte para diversos componentes mais usados e também conta com uma biblioteca extra.

Figura 9 – Criação de site com Adalo

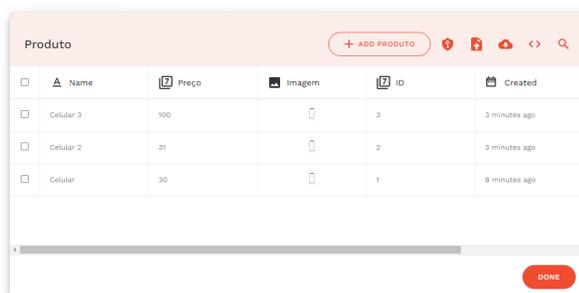


Fonte: Autor (2023)

Na Figura 13 é possível ver alguns componentes prontos. A usabilidade dessa plataforma surpreende, sendo a mais fácil desde o processo inicial, não foi possível testar a publicação, pois é uma funcionalidade paga, porém na opção de *preview* é possível ver que o site fica bem feito e com a responsividade já pronta, de forma automática.

Essa ferramenta possui também uma fácil edição aos dados, sendo mais intuitivo que o Bubble, como podemos ver na Figura 10, sendo possível realizar as mesmas operações, porém com termos mais amigáveis e uma interface mais agradável.

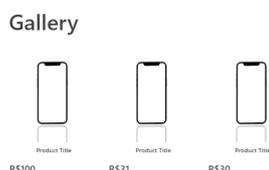
Figura 10 – Criação de dados da Adalo



Fonte: Autor (2023)

Em testes usando o bando de dados da Adalo foi possível facilmente fazer uma lista usando os dados criados no bando, demandando quase nenhum esforço ao desenvolvedor. O resultado pode-se ver na Figura 11.

Figura 11 – Site dinâmico criado com Adalo

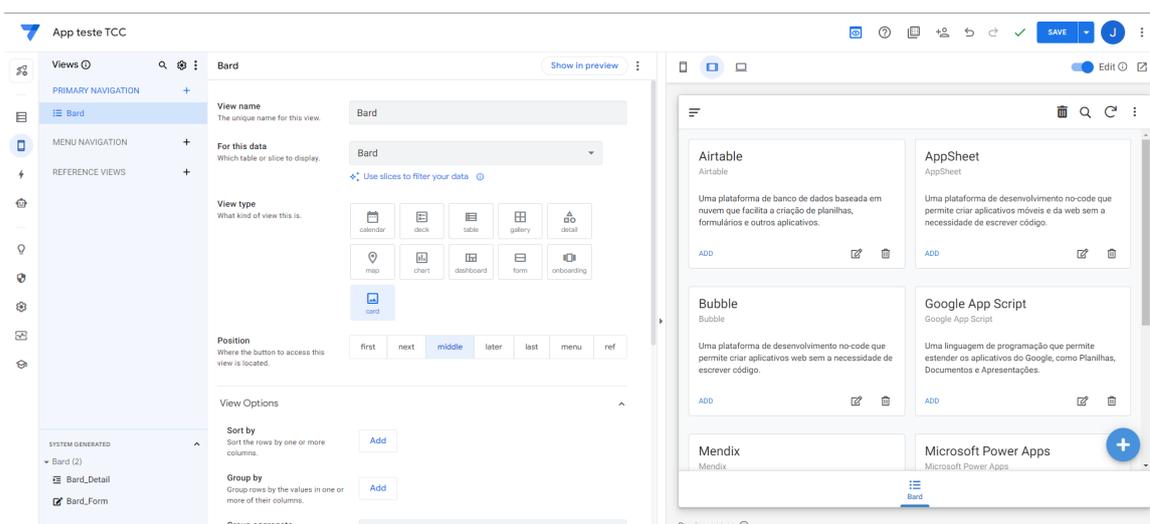


Fonte: Autor (2023)

### 3.4.5 AppSheet

Esta solução oferece uma opção rápida para transformar tabelas em aplicativos, facilitando a parte de gerenciamento de dados, já que tudo é tratado como planilha, dessa forma podemos dizer que é possível criar um aplicativo para gerenciar essas planilhas de forma mais dinâmica. Por ser uma ferramenta da própria Google é possível usar uma planilha do próprio Google como banco de dados, porém existem outras opções de integrações. Observe o teste feito com uma planilha usada no mapeamento de ferramentas:

Figura 12 – Uso AppSheet



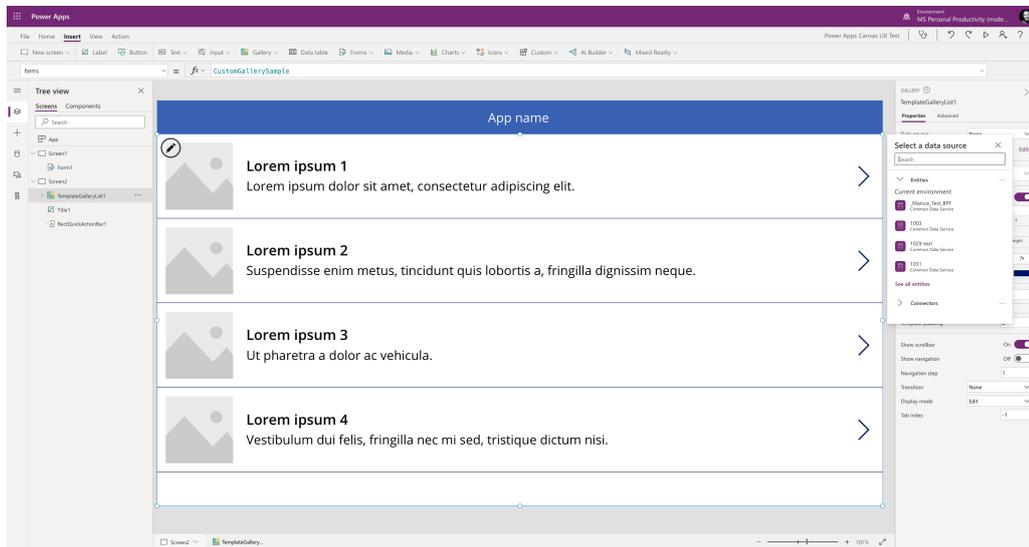
Fonte: Autor (2023)

### 3.4.6 Microsoft Power Apps

Não foi possível testar a solução, porém é disponibilizado no site um tour guiado onde é mostrado as diversas opções disponíveis. As conexões disponíveis são bem interessantes e tem

uma ótima sinergia com outras soluções da Microsoft. A plataforma tem maior utilidade para sistemas internos, isso pode significar um foco maior em eficiência operacional como foi citado no artigo [Leal \(2023\)](#).

Figura 13 – Interface do Power Apps

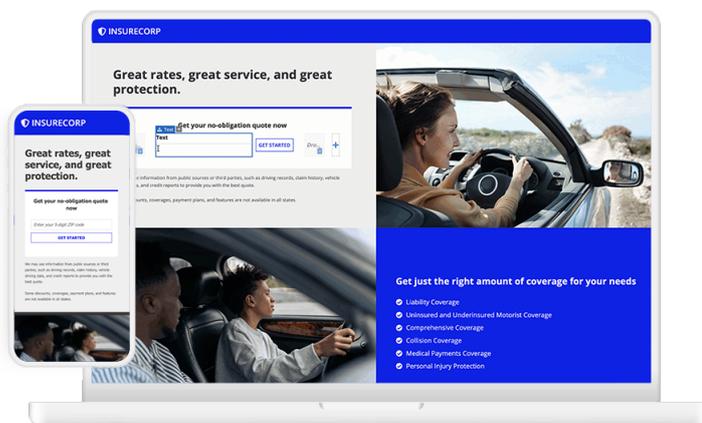


Fonte: [PowerApps \(2023\)](#)

### 3.4.7 Appian

A Appian tem um foco totalmente em processos internos, possuindo ferramentas bem importantes para eficiência operacional. Segundo a própria empresa, o sistema tem oportunidades para fins financeiros, governamentais, etc. Entrando em detalhes nas ferramentas disponíveis, pode-se ver na [Figura 14](#) à edição de telas é bem simples, sendo muito parecido com as demais analisadas.

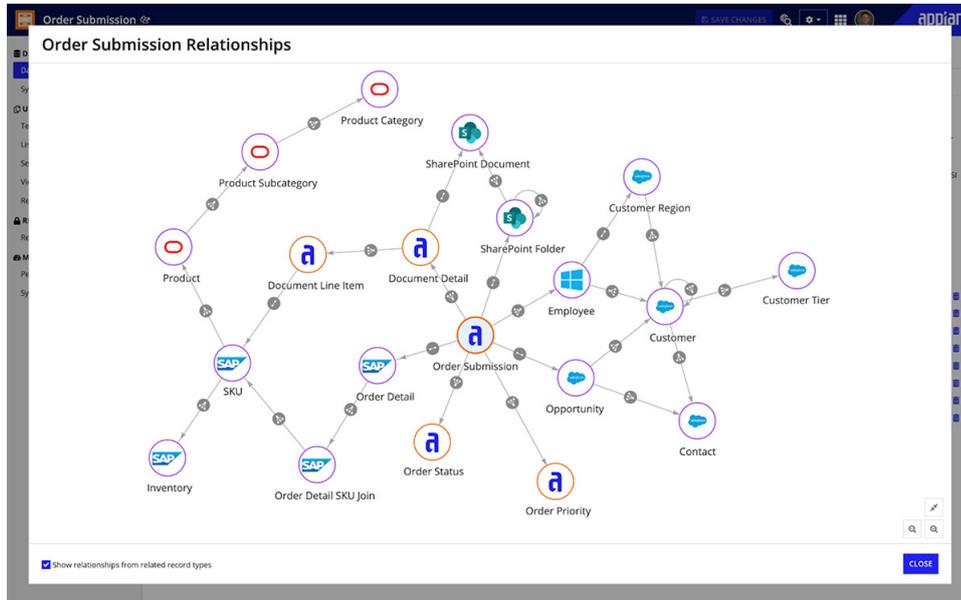
Figura 14 – Interface de edição



Fonte: [Appian \(2023\)](#)

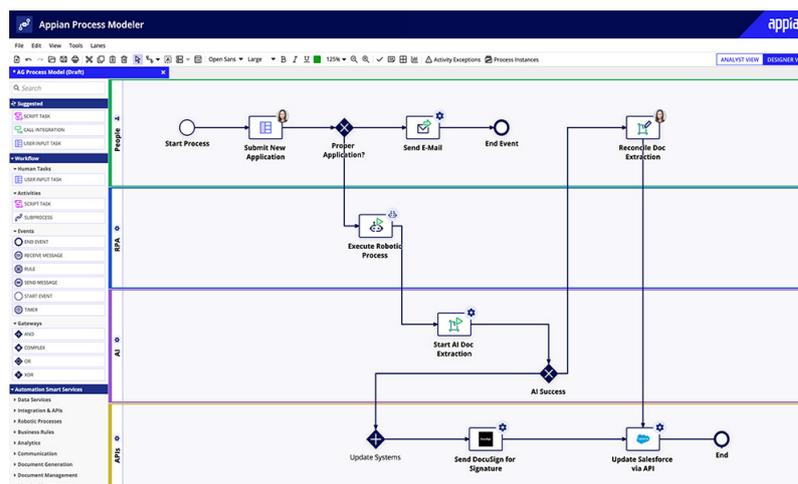
Reforçando o potencial de integração com outras ferramentas, temos ferramentas de rede para entender as integrações e conexões, como pode-se ver na Figura 15, o visual é bem intuitivo e fácil de entender. Seguindo na ideia de integrações, temos uma forma de criar um modelo de conexões mais complexo, visto na Figura 16.

Figura 15 – Interface de visualização de rede



Fonte: Appian (2023)

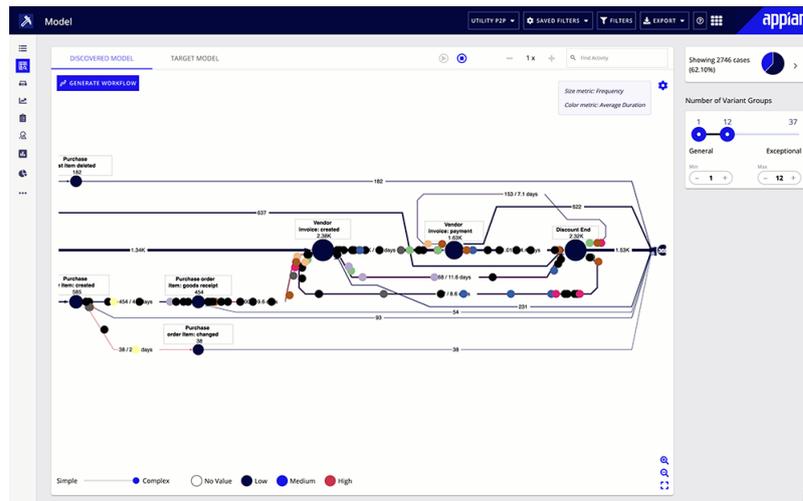
Figura 16 – Interface de edição de modelo



Fonte: Appian (2023)

Por último, o Appian traz uma ferramenta que pode fazer bastante diferença em empresas médias e grande, onde é possível visualizar como está o processo de negócio e identificar possíveis gargalos, na Figura 17 é possível ver como é a visualização.

Figura 17 – Interface de edição de modelo



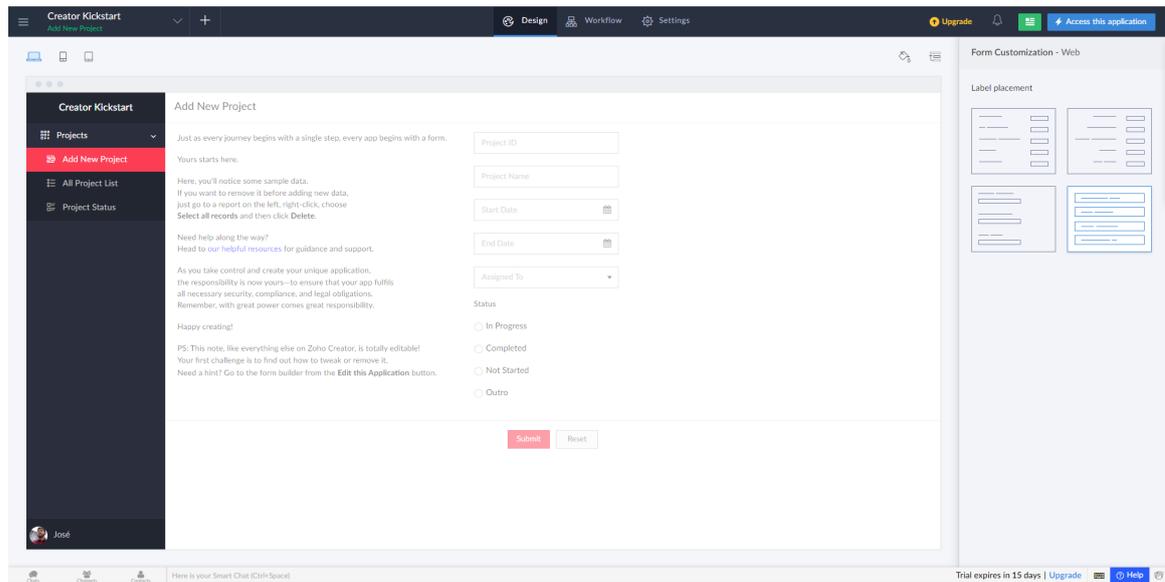
Fonte: Appian (2023)

### 3.4.8 Zoho Creator

A empresa Zoho já apresenta uma grande gama de produtos para empresas como, por exemplo, as ferramentas relacionadas a e-mail, gerenciamento de contas, etc. Então é um detalhe importante a ser considerado caso se esteja necessitando de outras ferramentas disponibilizadas pela Zoho.

Sobre a ferramenta em si, temos uma construção bastante rápida, em testes assim que foi respondido um breve formulário já foi gerado um sistema automaticamente para gerenciamento de projetos. O sistema gerado já possuía possibilidade para ser usado na forma padrão, inclusive via dispositivos moveis, porém, a edição dessa plataforma também é bem fácil, seguindo um padrão das demais como podemos ver na Figura 18.

Figura 18 – Edição do sistema



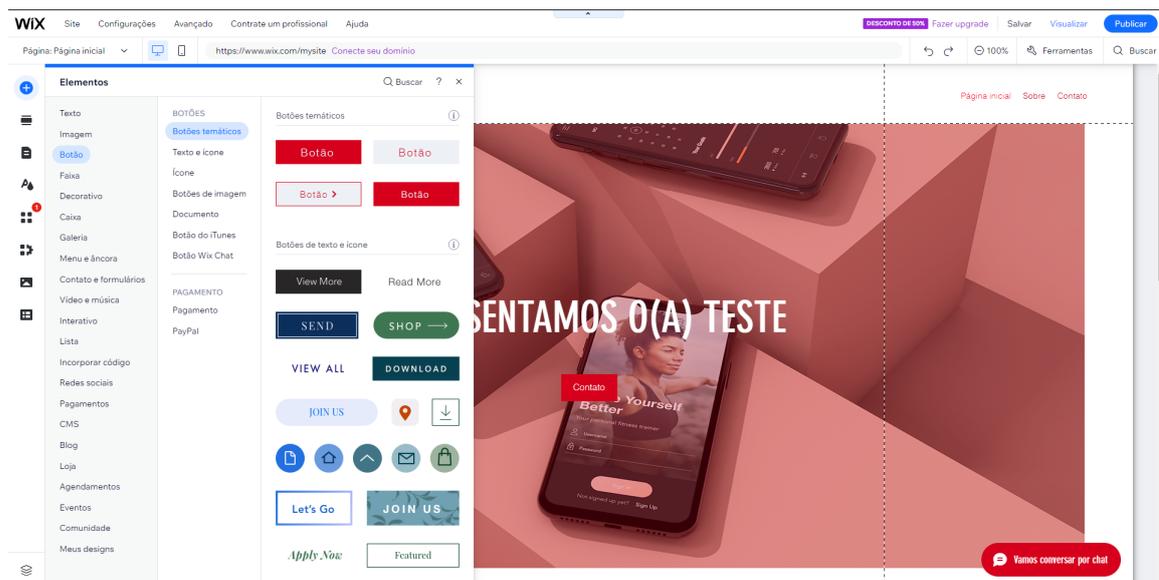
Fonte: Autor (2023)

Essa plataforma tem como objetivos a criação de sistemas internos, como, por exemplo, dashboards e gerenciamento de clientes ou produtos. Sendo assim não foram encontradas uma forma fácil de criar uma interface personalizada.

### 3.4.9 Wix

Wix é uma das soluções mais conhecidas, criação de sites sem código e de forma gratuita, como fornece diversos modelos, hospedagem gratuita e apresenta uma interface muito intuitiva, é a primeira opção para diversos pequenos negócios e também para pessoas físicas que querem se divulgar. O Wix apenas apresenta um problema com relação a sites gratuitos que é que a marca d'água apenas é removida pagando. A interface também é bem intuitiva e simples, como podemos ver na Figura 19

Figura 19 – Interface do wix para criação de site



Fonte: Autor (2023)

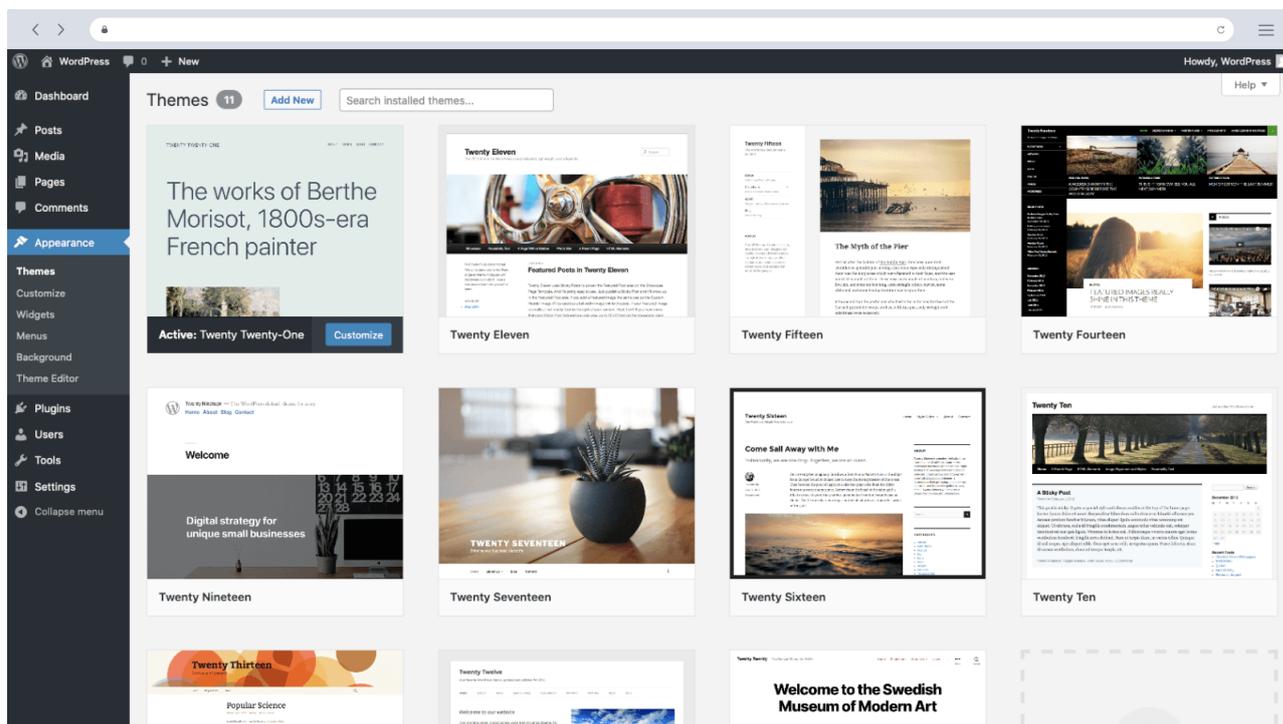
O Wix também possui uma forma fácil de criação de *e-commerce*, sendo rápido e fácil começar a vender. Para pequenos empreendedores que desejam começar a vender pela internet parenta ser uma excelente opção.

### 3.4.10 Wordpress

O Wordpress é a opção mais popular para criação de sites usando pouco código, porém por ter necessidade de subir em um servidor para começar a usar, se cria uma barreira para algumas pessoas. Ainda assim, o Wordpress está presente na história da internet, sendo até hoje a ferramenta principal usada por agências de marketing e pequenas empresas de software.

Esta solução pode ter um atrito inicial no uso, porém por ser muito popular existem diversos conteúdos na internet explicando desde como instalar até como otimizar o site. Na Figura 20 podemos ver como é a interface dessa ferramenta.

Figura 20 – Interface do Wordpress



Fonte: [wordpress \(2023\)](#)

O grande benefício do Wordpress é a sua comunidade disponibilizando diversos *templates* (gratuitos e pagos) e também *plugin* que auxiliam na construção dos *sites*. Graças ao seu código aberto é possível realizar modificações usando PHP para se adequar a sua necessidade.

### 3.4.11 Demais soluções

As soluções restantes apresentavam características muito comuns com as anteriores, sendo apenas uma forma diferente de corrigir problemas de eficiência operacional, elas não apresentaram diferenciais relevantes, não significando que são ruins, porém para fins de estudos, não faz sentido serem relevados, visto que há outras ferramentas parecidas, como as citadas acima.

## 3.5 Aprendizados retirados

Com esta busca foi possível identificar que há um grande interesse em ferramentas que facilitam a criação de sites e aplicativos de forma rápida e barata, podemos ver que é uma tendência mais atual, considerando que dos artigos buscados apenas um é de antes de 2020. No artigo do [Richardson e Rymer \(2016b\)](#) podemos visualizar com dados, como ferramentas *low-code* podem trazer maior agilidade no desenvolvimento, um exemplo citado pelo mesmo da *Spanish insurance provider* que diminuiu a previsão desenvolvimento de 2,7 anos para 13

semanas, nos demais artigos o ponto sempre citado é que ferramentas desse tipo não eliminam os profissionais de desenvolvimento, apenas diminuem os custos.

Um fato interessante é que no artigo de [S La H e RE \(2022\)](#) é mostrado o desenvolvimento de uma plataforma *no-code* focada em aplicativos de saúde e essa plataforma é usada no artigo do [A e S \(2023\)](#), acredito que o principal motivo seja que tem autores em comum nos dois artigos, contudo ao analisar os artigos, podemos ver que a proposta feita pelo PathVerse (plataforma criada no artigo citado) foi atingida, não apenas provando com os testes da época, mas sendo reforçado por outro artigo. Nesse caso vemos uma plataforma bem específica e por isso não possui muitas funções que seriam usadas em demais contextos.

Com a leitura dos artigos supracitados é possível retirar-se aprendizados sobre como essas ferramentas tem potencial de modificar o mundo da tecnologia, até mesmo com um potencial maior do que o próprio objetivo deste estudo. Uma ferramenta *no-code* e *low-code* tem potencial de criar sistemas complexos que atinjam os objetivos dos envolvidos sem muito custo. Uma grande oportunidade para empresas que possuem pouco desenvolvedores e os mesmos estão envolvidos em projetos mais prioritários, criando uma forma de inovar em outros departamentos das empresas sem a necessidade de atrapalhar a operação total da empresa. Outro ponto importante é que diferentes empresas em diferentes etapas do ciclo de vida podem se beneficiar destas ferramentas, desde *startups* que vão validar ideias, até empresas grande que podem inovar sem alterar seu fluxo padrão de trabalho.

Na busca por ferramentas foram encontradas diversas ferramentas com alto valor, o que pode ser um empecilho para uma startup que está iniciando sua operação, porém esse custo pode acabar sendo menor do que um desenvolvedor a depender da velocidade de desenvolvimento. Muitas outras ferramentas foram retiradas do mapeamento por serem mais ferramentas de CRM com customização, vimos que muitas empresas focam essas ferramentas para uso interno o que pode ajudar na automatização de processos internos sem atrapalhar a equipe de TI.

# 4

## Definição do produto para teste

A fim de promover uma compreensão pragmática sobre a utilidade e o potencial das ferramentas *no-code* e *low-Code*, é imprescindível conduzir um teste empírico destas, com o intuito de avaliar sua aplicabilidade. Neste capítulo, propõe-se a apresentação de um projeto de produto, delineando os parâmetros mínimos necessários para validar sua viabilidade no mercado.

### 4.1 Problema

O cenário empreendedor em Sergipe demonstra um crescimento progressivo, acarretando uma série de vantagens para o estado. Este crescimento promove um incremento significativo nas oportunidades de emprego, notadamente nas esferas tecnológicas, além de instigar um ambiente propício à inovação regional.

Entretanto, um desafio recorrente reside na dificuldade enfrentada por tais empresas para angariar investimentos e conquistar uma base sólida de clientes. Esta dificuldade, por vezes, obstrui a continuidade de suas operações, comprometendo-as pela escassez de recursos disponíveis.

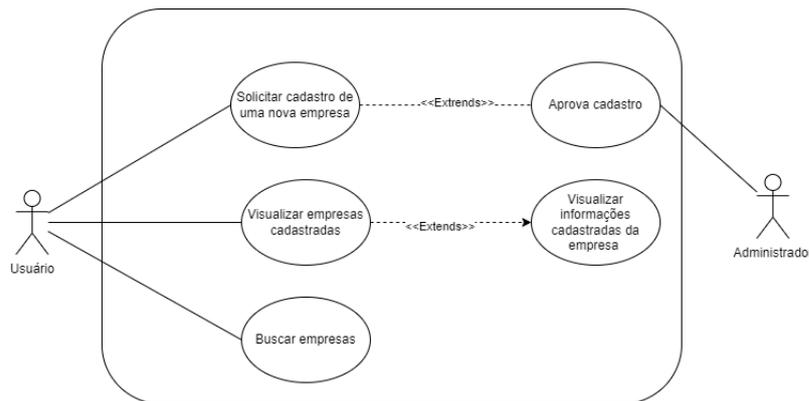
### 4.2 Solução

Para solucionar essa questão, foi proposta a ideia de um portal centralizado para *startups*. A hipótese é que essa plataforma ofereceria mais oportunidades para as empresas encontrarem clientes e investidores. Seria um espaço que reuniria diferentes *stakeholders*, como investidores, pessoas em busca de serviços e as próprias *startups*.

### 4.2.1 Casos de uso

Nesse momento tentaremos realizar o sistema mais simples possível a fim de validar se uma plataforma de busca por empresas teria uma boa aderência no mercado. Sendo assim, temos apenas um ator sendo o usuário, nesse momento não faz sentido colocar mais atores já que o foco tem que ser nessa visualização das empresas.

Figura 21 – Diagrama de casos de uso



Fonte: Autor (2024)

### 4.2.2 Requisitos

O produto mínimo para essa solução serve como centralizador de *startups*, para isso teremos os seguintes requisitos funcionais:

Tabela 4 – Requisitos funcionais.

Identificação	Descrição
RF001	Usuário poderá solicitar a adição de uma startup, suas respectivas informações, no sistema
RF002	Usuário poderá visualizar a lista de <i>startups</i> cadastradas no sistema
RF003	Usuário poderá visualizar as informações das <i>startups</i> cadastradas no sistema
RF004	Usuário poderá buscar por <i>startups</i> no sistema

Fonte: Autor (2023).

Neste momento o foco é na validação do produto, verificando se possui aderência do público-alvo e resolve de forma parcial ou completa o problema. Dessa forma, não será focado em um sistema completo, podendo haver trabalho manual envolvido ou até mesmo uso de outras ferramentas.

Quanto aos requisitos não-funcionais, o importante é que a plataforma consiga ser usada e que o objetivo seja atingido, mesmo que não tenha o melhor desempenho, dessa forma os requisitos não-funcionais ficam da seguinte forma:

Tabela 5 – Requisitos não-funcionais.

Identificação	Descrição	Classificação (Importante, Essencial ou Desejável)	Tipo
RFN001	O sistema deverá funcionar nos principais navegadores para computador	Essencial	Portabilidade
RFN002	O sistema deverá ser implementado sem necessidade do desenvolvedor programar ou programando apenas algumas linhas de código	Essencial	Implementação
RFN003	O sistema deve retornar todos os dados em até 1 minuto	Essencial	Eficiência

Fonte: Autor (2023).

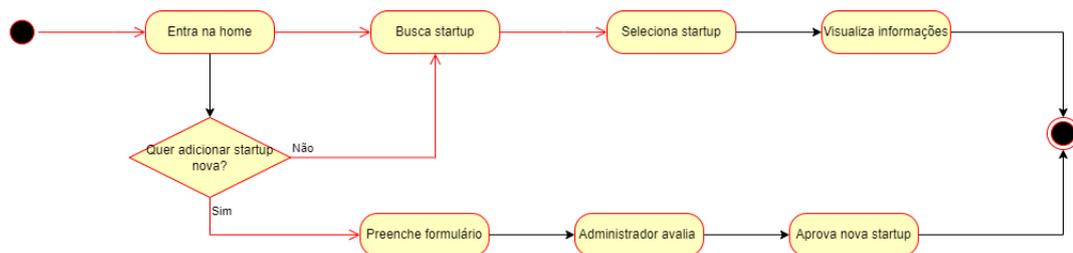
### 4.2.3 Fluxo sistema

Inicialmente, o produto deve apresentar as *Startups* cadastradas, incluindo informações que permitam aos usuários compreender sobre a natureza da empresa e seus planos. Além disso, é fundamental disponibilizar uma maneira para os usuários entrarem em contato com as *Startups*.

A parte de administração não precisa estar inclusa dentro do sistema em si, basta que a ferramenta escolhida para o desenvolvimento forneça uma forma de gerir os dados da plataforma.

Na Figura abaixo, são apresentados os dois fluxos mencionados anteriormente, envolvendo apenas dois atores: o usuário e o administrador. Nesse contexto, consideramos que a startup necessita de aprovação, pois há a possibilidade de indivíduos mal-intencionados tentarem comprometer a integridade da base de dados.

Figura 22 – Diagrama de classes

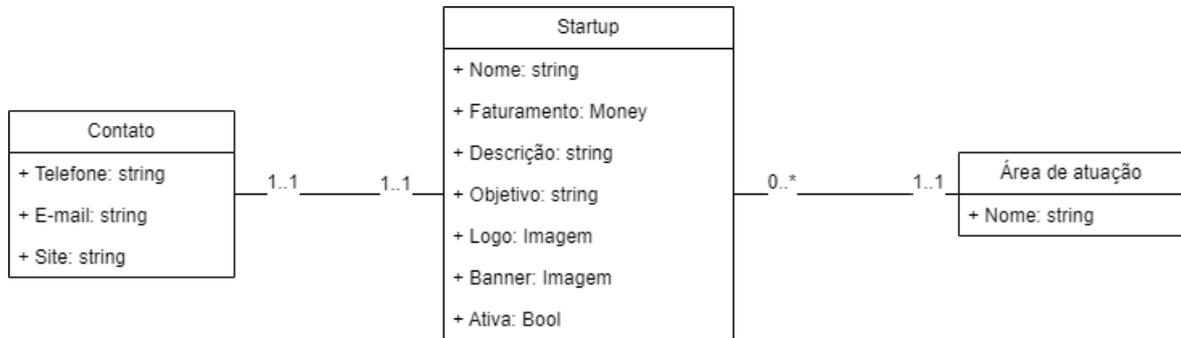


Fonte: Autor (2024)

### 4.2.4 Diagrama de classes

Para que o sistema possua uma estrutura de dados capaz de acompanhar a evolução do produto após a validação, foi desenvolvido o diagrama de classes, conforme representado na Figura 23. É importante salientar que este diagrama não abrange potenciais classes que serão geradas pelas próprias ferramentas, além das classes relacionadas a usuários e administradores.

Figura 23 – Diagrama de classes



Fonte: Autor (2024)

### 4.3 Escolha da ferramenta

Considerando a finalidade acadêmica deste estudo, optou-se não por uma única ferramenta, mas por três distintas, cada uma caracterizada por particularidades únicas. Tal abordagem possibilita, com base nos resultados obtidos, a escolha de outra ferramenta que proporcione uma experiência comparável. As ferramentas eleitas para análise são OutSystems, Adalo e Wix.

A escolha da OutSystems fundamenta-se na sua capacidade de integração com os bancos de dados mais amplamente utilizados no âmbito comercial, além de apresentar uma capacidade de gerar produtos mais robustos, podendo ser evoluído facilmente. Sendo escolhida como a ferramenta de estilo *low-code* a ser testada.

A Adalo destaca-se pela sua interface amigável, a qual se alinha de forma mais congruente com os objetivos do projeto, representando a opção mais otimista entre as três. Adicionalmente, a Adalo compartilha similaridades na experiência de uso com outras ferramentas, proporcionando *insights* sobre como outras opções poderiam se comportar. Sendo a opção escolhida de ferramenta *no-code*.

Por fim, o Wix, ainda amplamente utilizado na internet, se vende como uma ferramenta fácil de criação de sites gratuitos. A escolha dessa ferramenta se dá pelo principal motivo de que muitos dos casos os idealizadores dos produtos possuem pouca ou nenhuma noção de desenvolvimento de software, com Wix é possível realizar um trabalho manual pelo administrador do sistema.

# 5

## Desenvolvimento da solução

Para obter resultados mais consisos não foi definido uma interface fixa a ser seguida, assim pode-se usar dos modelos e particularidades de cada ferramenta. Como o objetivo não é criar um produto perfeito, mas sim um *MVP*, o importante é que o funcionamento esteja adequado com os requisitos listados no [4.2.2](#).

### 5.1 Solução do MVP usando Adalo

O processo de desenvolvimento utilizando Adalo inicialmente apresentou certa complexidade, decorrente da escassez de tutoriais incorporados na própria plataforma. No entanto, é possível mitigar essa lacuna ao recorrer aos recursos informativos disponíveis em seu *blog*, os quais se revelaram valiosos na orientação para a construção do *site*.

No tocante ao desenvolvimento e ao desempenho da aplicação, observou-se uma satisfação geral. Surpreendentemente, apenas 6 horas de trabalho foi necessárias para conceber um *site* plenamente funcional, atendendo a praticamente todos os requisitos estabelecidos.

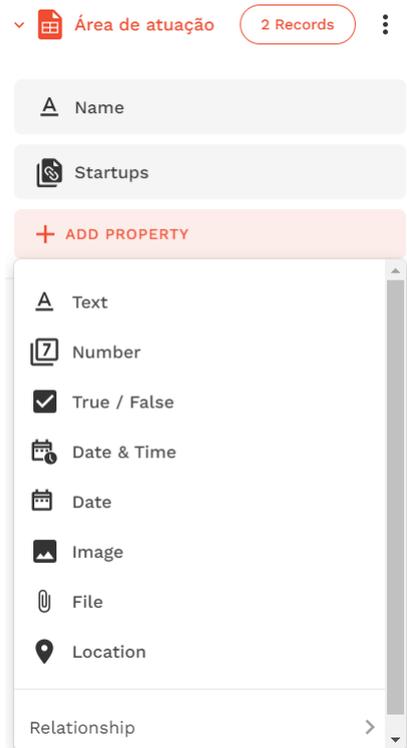
#### 5.1.1 Desenvolvimento

O período de desenvolvimento abrangeu um total de um 6 horas, englobando tanto o processo de aprendizado quanto o efetivo desenvolvimento. Todo o trabalho foi conduzido exclusivamente por meio das ferramentas fornecidas pela plataforma, demandando apenas consultas aos blogs internos da própria plataforma. As etapas de desenvolvimento compreenderam:

- Criação do banco de dados;
- Criação do *site*;
- Integração com a base de dados.

Por ser o aspecto central de nossa escolha, inicialmente foi realizada a configuração completa do banco de dados, levando apenas 1 hora para a criação de todas as tabelas com suas devidas conexões. Como previamente destacado, a plataforma oferece uma interface facilitada para a criação de tabelas, como exemplificado na Figura 24, e também para a gestão das mesmas, ilustrada pela Figura 25.

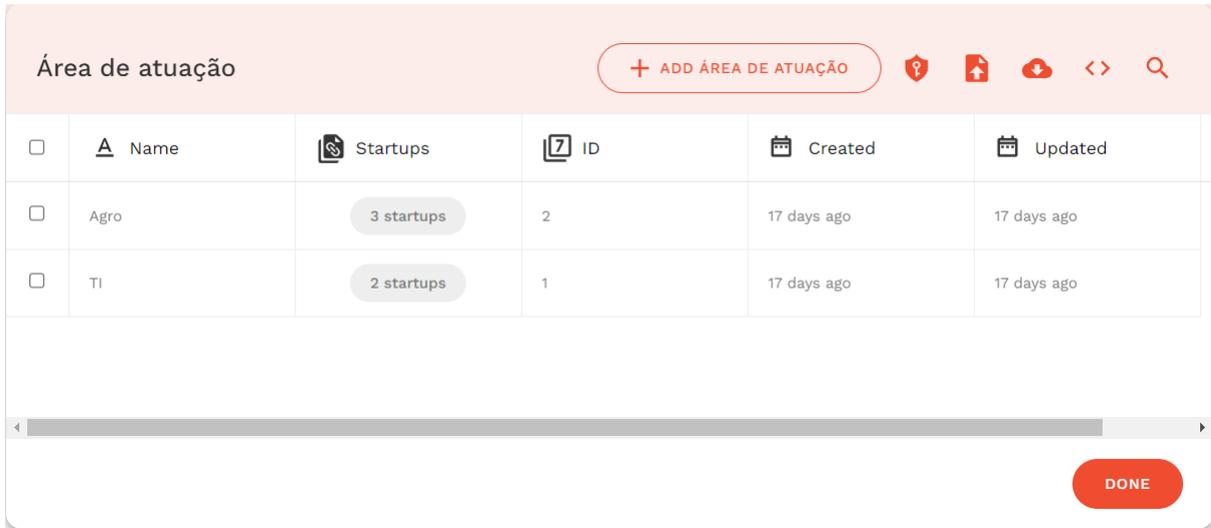
Figura 24 – Exemplo de como criar um campo novo em uma coleção



Fonte: Autor (2024)

Cada coleção do banco de dados requer obrigatoriamente um campo textual, não sujeito a exclusão, além de um identificador (ID) e campos para registro de data de criação e atualização, conforme evidenciado na Figura 25. Além disso, a ferramenta permite a atualização, exclusão e criação simplificada de novas entradas na tabela.

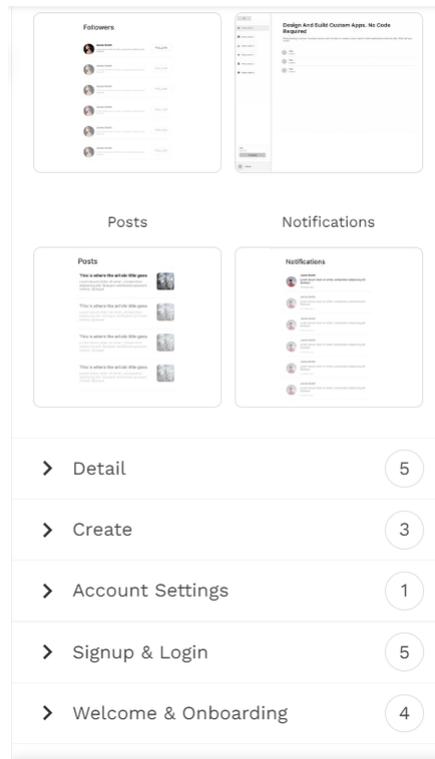
Figura 25 – Exemplo da ferramenta de gestão dos dados



<input type="checkbox"/>	<b>A</b> Name	Startups	ID	Created	Updated
<input type="checkbox"/>	Agro	3 startups	2	17 days ago	17 days ago
<input type="checkbox"/>	TI	2 startups	1	17 days ago	17 days ago

Fonte: Autor (2024)

A segunda etapa envolveu a elaboração do *layout* das páginas, levando apenas 3 horas para finalização. Embora seja possível utilizar *templates* fornecidos pela própria plataforma para criar o *site*, facilitando a integração, como apresentado na Figura 26, optou-se por uma experiência mais personalizada, utilizando apenas a seção de *templates* para a criação da tela de cadastro de uma *startup*.

Figura 26 – *Templates* disponibilizados pela plataforma

Fonte: Autor (2024)

Mesmo abstendo-se do uso de *templates* pré-configurados, não foram enfrentadas dificuldades, dado que a plataforma oferece uma gama diversificada de componentes para a construção das telas conforme a necessidade. A última etapa consistiu na integração com o banco de dados, levando apenas 2 horas, efetuada em cada componente que interage com as informações, um trabalho relativamente demorado, porém vital para utilizar diversas informações do banco na mesma tela, mesmo que provenientes de tabelas distintas.

### 5.1.2 Telas desenvolvidas

As telas foram todas desenvolvidas usando apenas componentes prontos sem o uso de código em nenhuma das partes, inclusive, isso causou alguns problemas que poderiam ser resolvidos com uso de CSS.

Para que o produto conseguisse atingir seu objetivo, é necessário ter uma página de visualização de empresas, cadastro e também de detalhes da empresa. A Adalo já fornece uma gestão de banco de dados completa, então não tem necessidade de criar telas para administração.

Agora, ao adentrarmos nos pormenores de cada tela, na Figura 27, observamos a seção que exhibe todas as *startups* registradas, oferecendo recursos de filtragem por área de atuação e nome das *startups*.

Figura 27 – Home da aplicação



Fonte: Autor (2024)

Na tela de cadastro, visível na Figura 28, é viável registrar uma nova *startup*. Entretanto, por razões de segurança, é imperativo que um responsável aprove a inclusão, alterando o campo de ativo para *True*, para que a empresa seja listada.

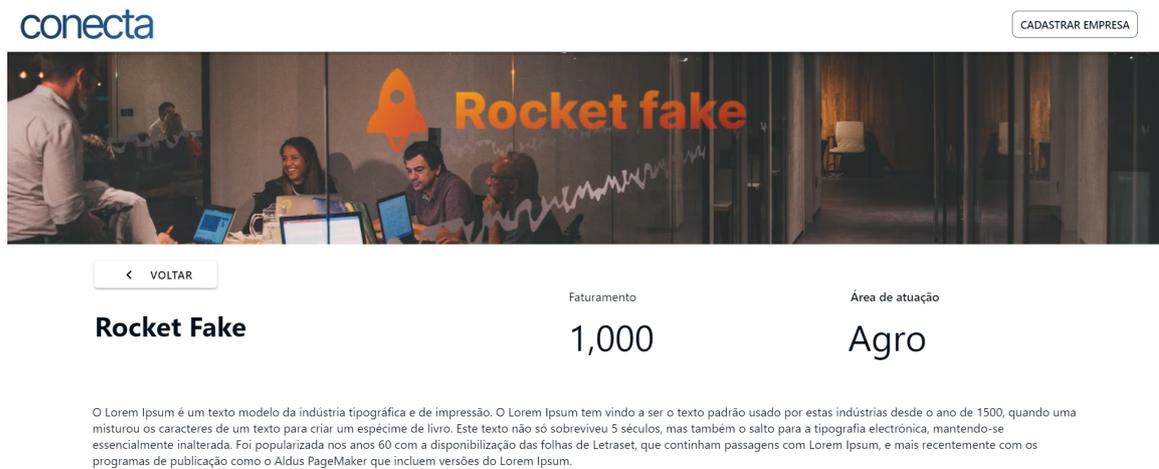
Figura 28 – Tela de cadastro



Fonte: Autor (2024)

Por fim, a tela de detalhes da empresa, associada à Figura 29, apresenta todos os dados cadastrados da empresa. Essa tela desempenha a função central de exibir as informações da empresa, sendo o núcleo do propósito deste produto.

Figura 29 – Tela de detalhes da empresa



Fonte: Autor (2024)

### 5.1.3 Conclusão

A ferramenta Adalo revelou-se uma excelente opção para aqueles que não possuem conhecimentos em programação, constituindo-se como uma alternativa fácil e ágil para o desenvolvimento de produtos. Notavelmente, ela pode ser empregada mesmo após o período de validação.

Entretanto, um ponto negativo desta ferramenta é exigência de uma atualização de plano para uma das opções pagas a fim de realizar a publicação. O valor mais acessível para tal é de 36 dólares por mês, o que pode representar um custo considerável, dependendo da disponibilidade financeira para investimento inicial.

## 5.2 Solução do MVP usando Outsystem

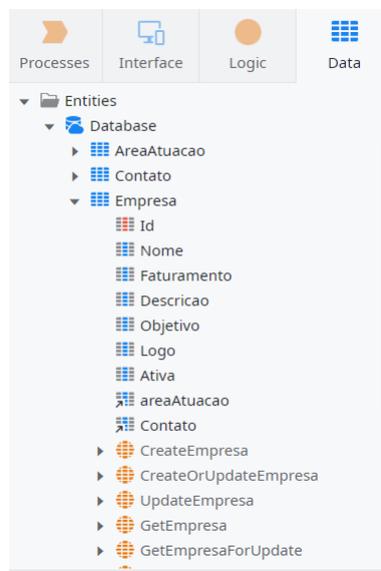
A plataforma OutSystems, desde o princípio, mostrou-se como a mais desafiadora, devido à sua dinâmica menos amigável, porém mais focada em robustez e organização. Ao contrário da plataforma anterior, não oferece uma interface amigável para o gerenciamento da aplicação, sendo necessário o *download* de um *software* adicional. Além disso, sua interface é totalmente voltada para o desenvolvimento, sem a intenção de auxiliar no gerenciamento do banco de dados.

Esta plataforma apresenta uma estrutura distinta, dividida em quatro seções: Processo, Interfaces, Lógica e Dados. Embora as seções de Processo e Lógica não tenham sido utilizadas, elas oferecem ferramentas que podem ser empregadas em sistemas que exigem integrações com outras plataformas. A interface da plataforma assemelha-se a uma estrutura de pastas, o que pode parecer confuso inicialmente, porém em sistemas de maior complexidade, essa abordagem facilitaria a localização dos elementos necessários.

## 5.2.1 Desenvolvimento

O processo de criação do banco de dados na plataforma OutSystems assemelha-se ao da plataforma anterior, com a distinção de que ao gerar uma entidade, também é automaticamente gerado o CRUD (Create, Read, Update, Delete) correspondente à entidade, o que posteriormente será utilizado nas interfaces. O desenvolvimento do banco de dados demandou apenas uma hora. No entanto, devido a uma limitação técnica que impossibilitou a adição de mais de um campo com imagem, não foi viável incluir o banner desejado. Contudo, esta limitação não afetou o produto final. A Figura 30 ilustra a estrutura de abas da plataforma OutSystems.

Figura 30 – Abas da plataforma OutSystem



Fonte: Autor (2024)

Considerando a estrutura de blocos e seus componentes intimamente integrados com os dados, não foi possível separar a parte de integração com o desenvolvimento de telas, totalizando seis horas. Esse tempo de desenvolvimento foi superior ao observado na plataforma Adalo, principalmente devido à necessidade de criar uma tela adicional para a gestão das empresas.

Na Figura 31, é possível visualizar com mais detalhes o funcionamento da plataforma OutSystems. Na parte esquerda da figura, encontra-se a lista de componentes disponíveis, abrangendo desde elementos simples, como texto, até componentes mais complexos, como gráficos e mapas. Os blocos nessa plataforma são organizados de forma diferente em comparação com a Adalo, onde os blocos ficam livres na interface. Na OutSystems, os blocos ocupam espaços específicos e mantêm um espaçamento entre os componentes, facilitando o alinhamento da interface. Porém, isso exigiu conhecimentos de CSS para manter os espaçamentos corretos.

Figura 31 – Aba de interface da plataforma OutSystem

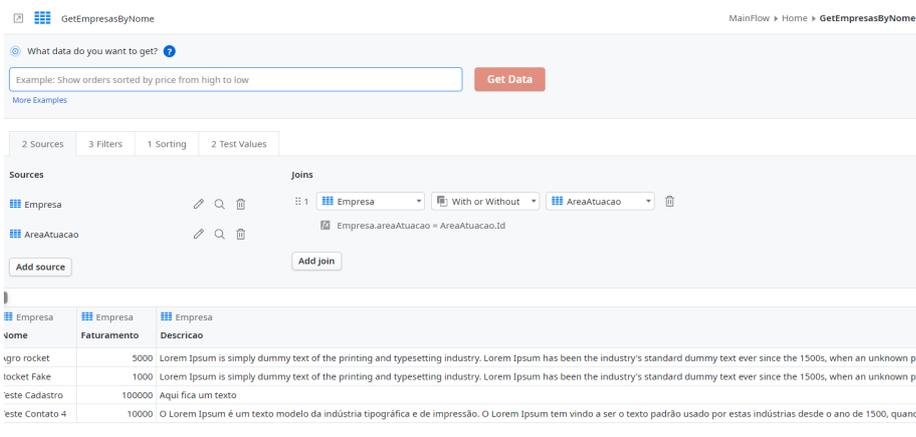


Fonte: Autor (2024)

Na parte direita da Figura 31, é apresentada a estrutura de pastas da interface, organizada por telas. Dentro de cada tela, são exibidas as variáveis locais (representadas por quadrados amarelos), as consultas no banco de dados (símbolo da aba de dados) e as ações (bolas amarelas).

A Figura 32 detalha o funcionamento das queries na plataforma OutSystems. Essas queries simplificam a estrutura de um SELECT em SQL, facilitando a compreensão dos dados que estão sendo recuperados, principalmente porque aceitam filtros e retornam os valores da query abaixo.

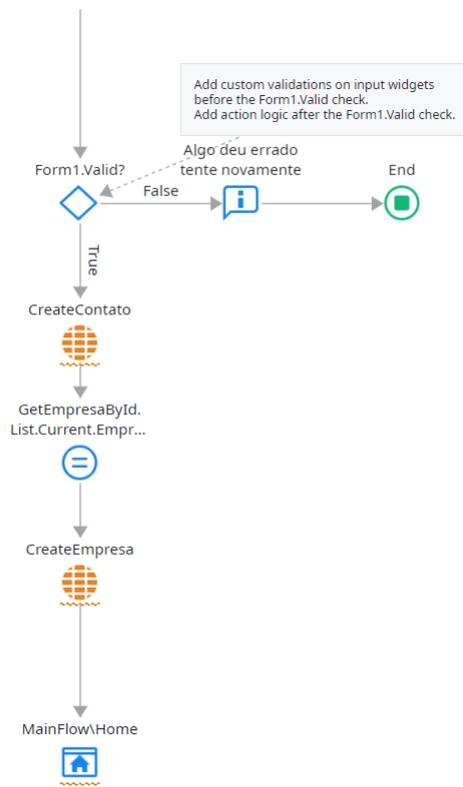
Figura 32 – Query usada para coletar as empresas de empresas



Fonte: Autor (2024)

A seção de ações apresenta um fluxograma simplificado que simula o funcionamento de um algoritmo, considerando a estrutura do banco de dados. Durante o cadastro, foi necessário criar o contato e estabelecer o relacionamento, conforme mostrado na Figura 33.

Figura 33 – Ação criada para o cadastro de uma empresa



Fonte: Autor (2024)

### 5.2.2 Telas desenvolvidas

A página inicial apresenta semelhanças com a desenvolvida utilizando a plataforma Adalo, oferecendo recursos de filtragem por área de atuação e busca por nome, além de *cards* contendo o nome da empresa, logotipo e área de atuação. É perceptível que os componentes estão mais alinhados, embora as imagens possam não ter sido renderizadas da melhor maneira. As regras de exibição de apenas empresas ativas são mantidas na listagem.

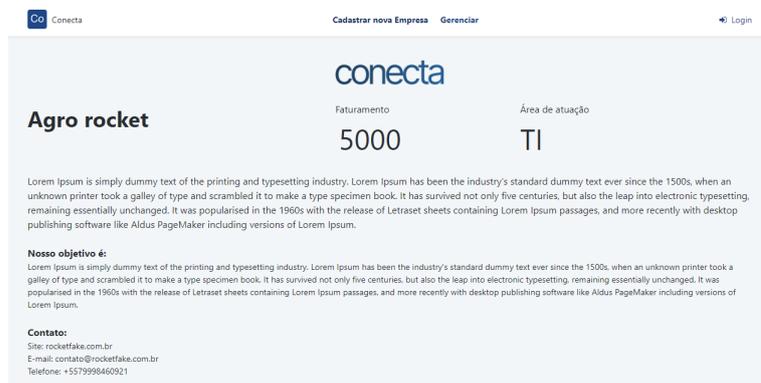
Figura 34 – Página inicial desenvolvida utilizando OutSystems



Fonte: Autor (2024)

A tela de detalhes segue uma estrutura semelhante, com a única diferença sendo a ausência do banner ao fundo, o que não afeta o propósito do produto, embora possa impactar visualmente de forma menos agradável. Devido à estrutura de posicionamento de blocos, o texto se ajusta ao tamanho do conteúdo de maneira adequada.

Figura 35 – Tela de detalhes desenvolvida utilizando OutSystems



Fonte: Autor (2024)

A tela de cadastro mantém a mesma estrutura sem grandes modificações, consistindo em um formulário contendo todos os campos pertinentes à empresa, incluindo informações de contato e área de atuação.

Figura 36 – Tela de cadastro desenvolvida utilizando OutSystems

A screenshot of a registration form titled 'Cadastre sua empresa gratuitamente'. Below the title is a subtitle: 'Entre para nossa comunidade e ganhe investimento e novos clientes para seu negócio'. The form contains several input fields: 'Nome \*', 'Logotipo' (with a file upload icon), 'Faturamento', 'Descrição \*', 'Objetivo \*', 'Área de atuação' (a dropdown menu with the text 'Selecione a área de atuação da sua empresa'), 'Telefone \*', and 'Email \*'. Each field is followed by a small asterisk indicating it is a required field.

Fonte: Autor (2024)

Por fim, a tela de aprovação das empresas cadastradas exibe informações básicas da empresa e apresenta um switch para ativação ou desativação, como demonstrado no cabeçalho, onde é necessário autenticação do usuário para acessar essa página.

Figura 37 – Tela de gerenciamento desenvolvida utilizando OutSystems

Nome	Faturamento	Descricao	Objetivo	Ativo	Area de Atuação	Contato
Agro rocket	\$5.000.00	Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.	Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.	<input checked="" type="checkbox"/>	TI	+5579998460921
Rocket Fake	\$1.000.00	Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining	Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining	<input type="checkbox"/>	Agro	+5579998460921

Fonte: Autor (2024)

### 5.2.3 Conclusão

A ferramenta OutSystems demonstrou um desempenho notável, sendo necessárias apenas 7 horas para desenvolvimento. Apesar de apresentar um atrito inicial e exigir habilidades de desenvolvimento para utilização efetiva, destaca-se pela sua organização e suporte para fluxos mais complexos. Caso haja evolução do produto, não haveria impedimentos para continuar o desenvolvimento utilizando essa plataforma.

Um ponto a ser considerado é que os componentes não são estilizados por padrão, o que pode resultar em uma interface final relativamente simples e pouco atrativa visualmente. Nesse sentido, é necessário um trabalho adicional com CSS, o qual pode não ser trivial para alguns usuários.

## 5.3 Solução do MVP usando Wix

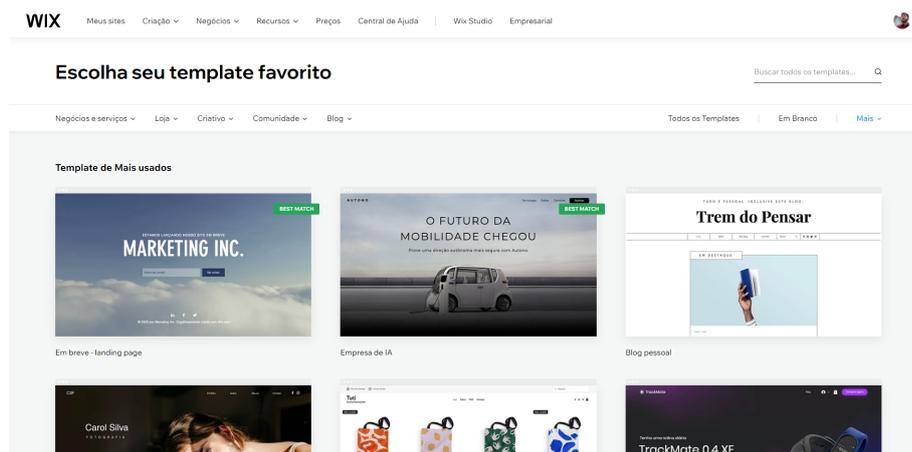
A plataforma Wix é reconhecida como uma das mais populares para a criação de *sites* gratuitos, o que justifica sua inclusão, apesar das limitações amplamente reconhecidas. Sua facilidade de uso e rapidez na criação de *sites* a tornam uma escolha atrativa para indivíduos com conhecimento limitado em desenvolvimento *web*.

Recentemente, o Wix tem investido em recursos de inteligência artificial (IA) para a criação de *sites*, como evidenciado por meio de um *chatbot* em fase beta. No entanto, até o momento, essa funcionalidade não produziu resultados para o nosso sistema. Embora exista a expectativa de que essa iniciativa possa acelerar ainda mais o processo de desenvolvimento de soluções utilizando o Wix, é importante ressaltar que não resolverá necessariamente os problemas identificados anteriormente.

### 5.3.1 Desenvolvimento

O desenvolvimento no Wix iniciou-se de forma distinta, uma vez que foi necessário selecionar um template para iniciar o processo de construção, conforme ilustrado na Figura 38. Contudo, mesmo com essa abordagem, o processo seguiu de maneira similar às outras plataformas, com todas as etapas realizadas utilizando os componentes fornecidos pelo sistema.

Figura 38 – Listagem de templates do Wix



Fonte: Autor (2024)

A parte relativa aos dados foi inicialmente abordada de forma semelhante às demais plataformas. Entretanto, ao se deparar com as limitações das ferramentas de gerenciamento de banco de dados e formulários, foi necessário reestruturar a abordagem, consolidando as opções de contato na tabela de empresas e removendo a possibilidade de *upload* de logotipos e *banners*, uma vez que tal funcionalidade não estava disponível nos formulários do plano gratuito.

Figura 39 – Tabela de empresas criada no Wix

Empresa Mais ações + Adicionar item

Visualização padrão + Nova visualização

Visualização padrão Tabela Atualizar ordem Gerenciar campos Ordenar Filtrar Buscar

#	Nome	Objetivo *	Descrição *	# Faturamento	Ativo *	Área de atuação	Site *	Telefone *	E-mail
	Rocket Fake	Teste	Teste	100	✓	Agro	http://fake.com	79998460921	contato@fake.com

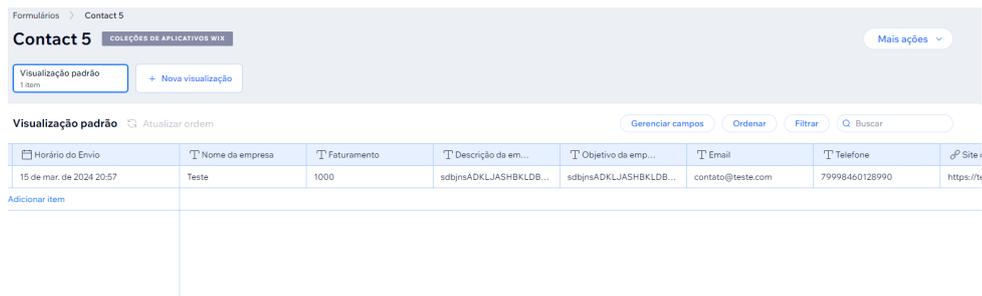
[Adicionar item](#)

Fonte: Autor (2024)

Os componentes, organizados por categorias, ofereciam diversos exemplos pré-estilizados, embora fosse possível aplicar estilizações adicionais. As conexões eram estabelecidas diretamente nos componentes, proporcionando liberdade para acessar as informações da tabela e seus relacionamentos. Para a criação da tela de detalhes, o próprio sistema configurou as informações a serem enviadas e sugeriu uma estrutura de apresentação.

O formulário não possuía a capacidade de criar novos itens, funcionando apenas para enviar e-mails e salvar as respostas em uma tabela de registros. Alguém precisaria inserir manualmente essas respostas na tabela. Embora esse comportamento estivesse previsto na modelagem, a limitação quanto ao upload de imagens tornou o fluxo de cadastro de uma nova empresa incompleto, como exemplificado na Figura 40.

Figura 40 – Tabela de repostas do formulário no Wix



Horário do Envio	Nome da empresa	Faturamento	Descrição da em...	Objetivo da emp...	Email	Telefone	Site
15 de mar. de 2024 20:57	Teste	1000	sdbjnsADKLJASHBKLD...	sdbjnsADKLJASHBKLD...	contato@teste.com	79998460128990	https://t...

Fonte: Autor (2024)

### 5.3.2 Telas desenvolvidas

O sistema foi estruturado em duas telas distintas, a saber: a página inicial, que compreende o formulário de cadastro e a listagem das empresas. A Figura 41 ilustra a listagem das empresas, enquanto a Figura 42 apresenta o formulário de cadastro.

Figura 41 – Listagem de empresas desenvolvida com Wix

**A plataforma que junta todas  
empresas que você precisa**

Empresas cadastradas



Fonte: Autor (2024)

Figura 42 – Formulário de cadastro de empresas desenvolvido com Wix

Cadastre sua empresa  
gratuitamente

Entre na nossa comunidade e ganhe novos investimentos e clientes

Nome da empresa *	Faturamento *
<input type="text"/>	<input type="text"/>
Descrição da empresa *	
<input type="text"/>	
Objetivo da empresa *	
<input type="text"/>	
Email *	Telefone *
<input type="text"/>	<input type="text"/>
Site da empresa *	
<input type="text"/>	
Escolha a área de atuação	
<input type="text"/>	

Fonte: Autor (2024)

Por fim, foi desenvolvida a tela de detalhes da empresa, seguindo a mesma linha visual adotada em outras plataformas, mas adaptando-se às limitações relacionadas às imagens.

Figura 43 – Tela de detalhes das empresas desenvolvida com Wix

< Back

Rocket Fake

Faturamento	Faturamento
100	Agro
Teste	
Teste	

Contatos

- <http://fake.com>
- 79998460921
- contato@fake.com

Fonte: Autor (2024)

### 5.3.3 Conclusão

Com essa plataforma, não foi possível alcançar todos os objetivos devido às limitações impostas pelo plano e pela própria plataforma. No entanto, o tempo de desenvolvimento foi notavelmente reduzido, concluído em apenas uma hora. Dessa forma, caso o escopo do projeto fosse reduzido, essa plataforma teria se destacado como a mais rápida em termos de implementação.

As limitações impostas afetaram diretamente a qualidade do produto, tornando-o insuficiente para passar pelo processo de validação. No entanto, seria viável explorar o uso de ferramentas de terceiros para mitigar algumas das lacunas identificadas, como a possibilidade de incorporar um formulário externo para contornar as limitações do sistema.

## 5.4 Comparação das ferramentas analisadas

Ao usar as três ferramentas conseguimos ver que cada uma se mostrou ser focada em públicos diferentes, sendo necessários níveis de conhecimento diferentes de engenharia de *software*. A OutSystem se mostrou a mais interessante para nível de escala já que sua organização e ferramentas permitem integrações e fluxos complexas, já a Adalo se mostrou mais forte para um público que não possui conhecimento vasto de desenvolvimento, mas possui uma noção de sistemas, essa ferramenta consegue gerar software completos sem uso de nenhum código, além de que apresenta a interface mais amigável. A ferramenta Wix se mostrou insuficiente para o nosso MVP, mas por ser uma ferramenta muito consolidada na construção de sites, não se pode descartar o seu uso, porém apresenta um escopo mais limitado de opções e é necessário pensar com cuidado se realmente vai ser suficiente para seu caso.

Na Tabela 6 podemos ver uma comparação mais direta entre as ferramentas, sendo dividido em tempo de desenvolvimento (estimado em horas), prós e contras do uso das ferramentas para o caso específico de MVP especificado.

Tabela 6 – Comparativo entre as ferramentas.

Nome	Tempo de desenvolvimento	Prós	Contras	Custo estimado
OutSystems	7 horas	Plataforma é robusta e suas funções são pensadas para desenvolvimento de sistemas complexos, sua organização ajuda bastante quando o projeto cresce, usa termos comuns do desenvolvimento Web, aplicando boas práticas de nome de funções criada automaticamente.	Exige um conhecimento mínimo de desenvolvimento, precisa baixar um software para usar o editor, não possui uma interface muito amigável.	209.588 reais por ano.
Adalo	6 horas	A plataforma Adalo apresenta uma interface simples e intuitiva, além de não exigir que o usuário tenha conhecimento prévio de desenvolvimento, a plataforma possui conteúdo informativo no próprio editor e também em seu <i>blog</i> .	A ferramenta deixa muito livre a adição de componentes, o que dificulta o uso de textos dinâmicos, deixando o produto final com aspecto de desalinhado, a necessidade de pagar para publicar também apresenta um ponto negativo.	36 dólares por mês.
Wix	1 hora	A plataforma oferece diversos <i>templates</i> e no futuro pretende usar IA para criação de <i>sites</i> , o que facilita bastante o desenvolvimento de <i>sites</i> .	Os recursos são bem limitados e no geral são focados em <i>sites</i> institucionais e lojas virtuais, para o nosso caso não se adequou bem, além de que o editor é bastante pesado o que causava travamentos constantes durante o desenvolvimento.	A solução atual custo zero, mas poderia ser melhorada com plano de 13 reais por mês.

Fonte: Autor (2023).

Em resumo, podemos dizer que a ferramenta OutSystems apresenta maior sinergia com o produto proposto, já que o mesmo conseguiria ser evoluído com facilidade e seu uso de CSS ajuda a trabalhar com textos dinâmicos, essencial para as páginas de detalhes da empresa. A ferramenta Adalo se mostrou a melhor em casos genéricos, trazendo praticidade e facilidade para o desenvolvimento, por mais que tenha seus problemas poderiam ser mitigados, para um caso onde o MVP visa apenas validar, sem esperança de evoluir o software da mesma forma poderia ser uma opção mais interessante que a OutSystems. Por último, como falado anteriormente, a Wix apresenta uma sólida popularidade, mas apenas tem uso para caso mais genéricos e *sites*.

Outro problema é relacionado ao custo das ferramentas, por mais que as ferramentas tenham cumprido seu papel em fornecer celeridade ao desenvolvimento, ainda seriam um grande dificuldades para pequenas empresas usarem essas ferramentas.

## 5.5 Opção alternativa

Por mais que não seja o foco do estudo, visto que as soluções apresentadas necessitariam um alto investimento, uma oportunidade que vemos é o uso de IAs como ChatGPT para a criação dos códigos.

Em um experimento foi usado o seguinte *prompt* para tentar gerar o mesmo produto "Crie um *site* para gerar uma pagina com uma lista de empresas, a lista deve conter o nome da empresa, área de atuação e logo. Ao clicar na empresa deve ser encaminhado para uma página com os detalhes da empresa que são: objetivo, descrição, faturamento e contato". Como resultado, foi gerado as telas mostradas nas figuras 44 e 45. O código gerado não apresentou falhas, porém deixou tudo estático.

Figura 44 – Home gerado pelo chatgpt



Fonte: Autor (2024)

Figura 45 – Home gerado pelo chatgpt



Fonte: Autor (2024)

Nesse caso poderia ser usado uma ferramenta para hospedagem gratuita e poderia se obter um *site* em pouco tempo e sem custo, os problemas são a necessidade de avaliação para caso que necessitasse modificações no código, e também as ferramentas de IA não conseguem criar *sites* com interfaces mais complexas, ocasionando em um *site* sem um visual agradável.

# 6

## Considerações Finais e Trabalhos Futuros

Este estudo teve como propósito fornecer uma alternativa para o desenvolvimento de *MVP* de maneira ágil e acessível, realizando um levantamento das ferramentas disponíveis e uma análise concisa de cada uma, seguida por testes práticos nas ferramentas OutSystems, Adalo e Wix. A escolha dessas ferramentas não se deu pela sua excelência absoluta, mas sim pela presença de especificidades relevantes para embasar uma decisão informada.

Inicialmente, o escopo do estudo abrangeu o mapeamento de 17 ferramentas, das quais 10 foram submetidas a testes práticos e 3 foram selecionadas para o desenvolvimento de uma solução real. Dentro desse contexto, foram consideradas ferramentas tanto de baixo código quanto sem código, oferecendo flexibilidade de escolha conforme o perfil do desenvolvedor.

A modelagem do *MVP* foi baseada para testar a capacidade de gerenciamento de tabelas de dados e uso de conteúdos dinâmicos, apresentando um fluxo simples, porém usando ferramentas de visualização, pesquisa, criação e atualização. O *MVP* ainda foi pensando para não fugir do tema de empresas, já que o produto proposto também visa ajudar empresas em estados iniciais.

O desenvolvimento das aplicações foram feitos sem uso de *templates*, considerando que os usuários poderiam querer personalizar o seu sistema. Também o uso de *templates* ofuscaria como seriam realizadas as integrações, sendo critério importante para aplicações que necessitassem de maiores configurações.

No que tange ao desenvolvimento das aplicações, duas das três ferramentas demonstraram capacidade satisfatória para suportar a solução proposta, mais detalhes na seção 5.4. Entre elas, a Adalo se destacou por sua fidelidade aos requisitos mapeados, enquanto o OutSystems apresentou maior escalabilidade, o que se mostra relevante para projetos com potencial de crescimento.

Ao final, temos que as ferramentas no-code e low-code realmente apresentam potencial de reduzir o investimento inicial de *MVPs* e também agilizar o seu desenvolvimento, considerando metodologias mais agressivas poderiam lançar um sistema em apenas um dia. Reduzindo os

riscos em caso de sistemas para situações emergenciais, como foi o caso da pandemia da COVID-19. O único problema é relacionado a seu custo, que pode ser um problema grave a depender do estágio do produto.

Como perspectivas futuras, sugere-se a avaliação de novas plataformas visando explorar novas possibilidades. Além disso, seria proveitoso incorporar critérios adicionais de avaliação, como usabilidade e acessibilidade. Outro ponto de evolução deste estudo consiste em submeter a plataforma OutSystems a um estresse mais intenso, ampliando sua escala e complexidade, a fim de compreender seus limites e capacidades em cenários mais desafiadores. Outro ponto a se explorar seria o uso de IAs para criação desses produtos.

# Referências

- A, R. R. W.; S, L. *Prevention and Financial-Incentive mHealth Program Using a “No-Code” Mobile App Builder: Development and Usability Study*. e43823. [s.n.], 2023. v. 7. Disponível em: <<https://forurmativ.jmir.org/2023/1/e43823>>. Citado 2 vezes nas páginas 22 e 36.
- ABES. *Estudo Mercado Brasileiro de Software – Panorama e Tendências 2023*. [S.l.], 2023. Citado na página 11.
- APPIAN. 2023. Disponível em: <<https://appian.com/products/platform/overview.html>>. Acesso em: 04/10/2023. Citado 3 vezes nas páginas 30, 31 e 32.
- JOHNSON, W. Manage your organization as a portfolio of learning curves. *Havard Business Review*, 2022. Citado na página 16.
- LEAL, V. C. Desenvolvimento de um aplicativo em plataforma low code integrado a business intelligence para vistorias de unidades residenciais. 2023. Citado 2 vezes nas páginas 22 e 30.
- LEVITT, T. Exploit the product life cycle. *Havard Business Review*, 1965. Citado 2 vezes nas páginas 11 e 15.
- MATEO, F. W.; REDCHUK, A. Process optimization in the steel industry using machine learning adopting an artificial intelligence low code platform. In: *X Jornadas de Cloud Computing, Big Data & Emerging Topics (La Plata, 2022)*. [S.l.: s.n.], 2022. Citado na página 22.
- OUTSYSTEM. 2023. Disponível em: <<https://www.outsystems.com/pt-br/low-code-platform/development-deployment/>>. Acesso em: 03/10/2023. Citado 2 vezes nas páginas 24 e 25.
- POWERAPPS. 2023. Disponível em: <<https://powerapps.microsoft.com/pt-br/guidedtourof-power-platform/power-apps/2/1>>. Acesso em: 04/10/2023. Citado na página 30.
- RICHARDSON, C.; RYMER, J. R. The forrester wave™: low-code development platforms, q2 2016. *Forrester, Washington DC*, 2016. Citado na página 18.
- RICHARDSON, C.; RYMER, J. R. Vendor landscape: The fractured, fertile terrain of low-code application platforms. *Forrester, Janeiro*, p. 12, 2016. Citado 3 vezes nas páginas 22, 23 e 35.
- RIES, E. *Startup Enxuta*. [S.l.]: Editora Leya, 2012. Citado 3 vezes nas páginas 16, 17 e 18.
- RUSCIO, D. D. et al. Low-code development and model-driven engineering: Two sides of the same coin? *Software and Systems Modeling*, Springer, v. 21, n. 2, p. 437–446, 2022. Citado 2 vezes nas páginas 11 e 18.
- S LA H, W. A. L.; RE, R. A “no-code” app design platform for mobile health research: Development and usability study. *JMIR Form Res* 2022, v. 6, 2022. Citado 2 vezes nas páginas 22 e 36.
- WORDPRESS. 2023. Disponível em: <<https://br.wordpress.org/>>. Acesso em: 04/10/2023. Citado na página 35.