



UNIVERSIDADE FEDERAL DO CEARÁ
CAMPUS DE QUIXADÁ
CURSO DE GRADUAÇÃO EM SISTEMAS DE INFORMAÇÃO

JAILSON DE SOUSA BASTOS

**ANÁLISE COMPARATIVA ENTRE O IRIS LOW-CODE CERTIDÃO E OUTRAS
PLATAFORMAS LOW-CODE**

QUIXADÁ
2023

JAILSON DE SOUSA BASTOS

ANÁLISE COMPARATIVA ENTRE O IRIS LOW-CODE CERTIDÃO E OUTRAS
PLATAFORMAS LOW-CODE

Trabalho de Conclusão de Curso apresentado ao
Curso de Graduação em Sistemas de informação
do Campus de Quixadá da Universidade Federal
do Ceará, como requisito parcial à obtenção do
grau de bacharel em Sistemas de informação.

Orientador: Prof. Dr. Davi Romero de
Vasconcelos

QUIXADÁ

2023

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Sistema de Bibliotecas
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

- B329a Bastos, Jailson de Sousa.
Análise comparativa entre o Iris Low-code Certidão e Outras Plataformas Low-Code / Jailson de Sousa Bastos. – 2023.
48 f. : il. color.
- Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Campus de Quixadá, Curso de Sistemas de Informação, Quixadá, 2023.
Orientação: Prof. Dr. Davi Romero de Vasconcelos.
1. Low-Code. 2. Íris low-code certidão. 3. Comparação entre plataformas low-code. I. Título.
CDD 005
-

JAILSON DE SOUSA BASTOS

ANÁLISE COMPARATIVA ENTRE O IRIS LOW-CODE CERTIDÃO E OUTRAS
PLATAFORMAS LOW-CODE

Trabalho de Conclusão de Curso apresentado ao
Curso de Graduação em Sistemas de informação
do Campus de Quixadá da Universidade Federal
do Ceará, como requisito parcial à obtenção do
grau de bacharel em Sistemas de informação.

Aprovada em: ____/____/____

BANCA EXAMINADORA

Prof. Dr. Davi Romero de Vasconcelos (Orientador)
Universidade Federal do Ceará (UFC)

Prof. Dr. Regis Pires Magalhães
Universidade Federal do Ceará (UFC)

Prof. Dr. Bruno Góis Mateus
Universidade Federal do Ceará (UFC)

Dedico este trabalho a minha mãe, a estrela da minha vida que faz brilhar os meus dias, minha esposa por estar ao meu lado a todo momento e minha filha, minha razão de viver.

AGRADECIMENTOS

Inicialmente, agradeço a minha Mãe, Marlene de Sousa Pereira, minha companheira, Maria Naiane Moura Pereira, aos meus sogros, Maria Leonilda e Antônio Vladimir e aos meus amigos, Rene Arnold, Aglaiziane Lima e Samuel Bruno por sempre me incentivarem e me darem forças para que eu pudesse dar o meu melhor e continuar ao longo desses anos buscando e conseguindo conquistas. Mesmo que muitas vezes não estivéssemos próximos, não teria conseguido sem a força que vem de vocês.

Meus sinceros agradecimentos aos meus familiares, Adaline de Sousa Bastos, Dário Gomes, Adailson de Sousa Bastos, Margarete de Sousa Pereira, Marylene de Sousa Pereira e Marta de Sousa Pereira e demais que mesmo distante geograficamente, nunca deixaram de me apoiar e me incentivar para que eu conseguisse alcançar meus objetivos.

Agradeço ao Prof. Dr. Davi Romero de Vasconcelos por ter me aceitado como orientando no Trabalho de Conclusão de Curso (TCC) e pelo apoio e paciência ao longo desse desafio.

Agradeço aos professores Prof. Dr. Regis Pires Magalhães e Prof. Dr. Bruno Góis Mateus por aceitarem participar da banca avaliadora deste trabalho para que ao final da escrita desse trabalho ele contribuísse com um produto que está sendo desenvolvido para melhorar a vida do cidadão Cearense.

Agradeço ao ÍRIS | Laboratório de Inovação e Dados do Governo do Ceará pela oportunidade de contribuir com o *Íris Low-Code Certidão*, que foi desenvolvido no âmbito do projeto Governo Digital do Estado do Ceará (04772314/2020/FUNCAP), financiado pela Fundação Cearense de Apoio ao Desenvolvimento Científico e Tecnológico (FUNCAP).

Também gostaria de agradecer à Universidade Federal do Ceará — Campus de Quixadá, por fornecer os recursos e o ambiente propício para a realização deste trabalho. Agradeço aos professores, pelo conhecimento transmitido e pelo incentivo constante ao aprendizado e à pesquisa.

Por fim, agradeço aos amigos que fiz ao longo da graduação, sem o apoio de vocês a minha trajetória teria sido mais difícil.

“O futuro pertence àqueles que acreditam na beleza de seus sonhos.”

(Eleanor Roosevelt)

RESUMO

Plataformas de desenvolvimento *low-code* são ferramentas usadas para criação de *software* de forma rápida e prática sem exigir muito conhecimento em programação, usando recursos de arrastar e soltar os componentes em tela e configurações simples. Devido a essas facilidades tais ferramentas vêm chamando cada vez mais a atenção do mercado de desenvolvimento e com isso vem possibilitando que empreendedores, mesmo que com baixo investimento, consigam tirar suas ideias do papel. Com o surgimento de vários serviços de emissão de certificados oferecidos pelas secretarias do Estado do Ceará através da carta de serviços do cidadão, surgiu o interesse em investigar formas mais fáceis de lidar com a criação de *templates* de certificado. Pensando nisso, o Projeto Governo Digital, em parceria com o ÍRIS | Laboratório de Inovação e Dados do Governo do Ceará, deu início ao projeto do *Iris Low Code Certidão*. Por ser uma ferramenta que está em constante evolução, surge a necessidade de investigar outras funcionalidades que poderiam melhorar a plataforma do Estado do Ceará. Portanto, o presente trabalho visa realizar uma comparação entre o *Iris Low-Code Certidão* e 3 plataformas de desenvolvimento *low-code* mais usadas neste mercado, gerando uma tabela que serve como catálogo para analisar possíveis melhorias que podem ser implementadas na plataforma do *Iris Low-Code Certidão*.

Palavras-chave: Low-Code; Iris low-code certidão; Comparação entre plataformas low-code

ABSTRACT

Low-code development platforms are tools used to create software for quickly and practically without requiring much knowledge in programming, using drag and drop components into canvas and simple settings. Due to these facilities tools have been drawing more and more attention from the development market and with that has been enabling entrepreneurs, even with low investment, to be able to take your ideas on paper. With the emergence of various certificate issuing services offered by the secretariats of the State of Ceará through the citizen's service charter, interest arose in investigating easier ways to handle the creation of certificate templates. Thinking in this regard, the Digital Government Project, in partnership with ÍRIS | Innovation and Data Laboratory of the Government of Ceará, started the Iris Low Code Certificate project. For being a tool which is constantly evolving, there is a need to investigate other functionalities that could improve the platform of the State of Ceará. Therefore, the present work aims to a comparison between the Iris Low-Code Certificate and 3 low-code development platforms most used in this market, generating a table that serves as a catalog to analyze possible improvements that can be implemented in the Iris Low-Code Certificate platform.

Keywords: Low-Code; Iris low-code certidão; Comparison between low-code platforms

LISTA DE ILUSTRAÇÕES

Figura 1 – Consulta dos dados na plataforma <i>Outsystem</i>	19
Figura 2 – Exemplo de <i>workflow</i> criado na plataforma da <i>Outsystem</i>	22
Figura 3 – Gartner - Quadrante mágico para plataformas de aplicativos <i>low-code</i> corporativos	24
Figura 4 – Representação (<i>Íris Low-code Certidão</i>)	25
Figura 5 – Representação (<i>Zoho Creator</i>)	26
Figura 6 – Representação (<i>Outsystems</i>)	27
Figura 7 – Representação (<i>Mendix Studio</i>)	27
Figura 8 – Representação (<i>Mendix Studio Pro</i>)	28
Figura 9 – Fluxo das atividades que serão executadas	34
Figura 10 – modelo de certidão replicado nas plataformas	42
Figura 11 – <i>Zoho Creator</i> - Tela de relatório de dados	44
Figura 12 – <i>Mendix</i> - Tela de importação de dados	44
Figura 13 – <i>Iris Low-Code Certidão</i> - Opção de configurar os dados com as planilhas do <i>Google</i>	46
Figura 14 – <i>Zoho Creator</i> - Tela para criação de <i>template</i>	47
Figura 15 – <i>Outsystem</i> - Tela para criação de <i>template</i>	47
Figura 16 – <i>Iris Low-Code Certidão</i> - Tela de criação de formulário	48
Figura 17 – <i>Iris Low-Code Certidão</i> - Prévia da tela do formulário	48
Figura 18 – <i>Zoho Creator</i> - Tela para implantação de ambientes de desenvolvimento	49

LISTA DE TABELAS

Tabela 1 – Receita de tecnologias de desenvolvimento <i>low-code</i> (milhões de dólares americanos)	20
Tabela 2 – Principais razões pelo qual as organizações usam <i>low-code</i>	20
Tabela 3 – Por que algumas organizações ainda não estão usando <i>low-code</i> ?	20
Tabela 4 – Tempo de criação do <i>template</i> nas 3 plataformas	43

LISTA DE QUADROS

Quadro 1 – Vantagens e desvantagens de ferramentas low-code	15
Quadro 2 – Critérios de avaliação relacionados a capacidade de execução	23
Quadro 3 – Critérios de avaliação relacionados a visão de mercado	23
Quadro 4 – Grupo de características	30
Quadro 5 – Questões de pesquisa	31
Quadro 6 – Comparação entre os trabalhos relacionados e o proposto.	33
Quadro 7 – Plataformas <i>low-code</i> do quadrante mágico	38
Quadro 8 – Seleção das plataformas do quadrante mágico	40
Quadro 9 – Recursos de banco de dados	43
Quadro 10 – Integração com <i>API</i>	45
Quadro 11 – Outras funcionalidades	46

SUMÁRIO

1	INTRODUÇÃO	14
1.1	Objetivos	16
1.1.1	<i>Objetivo Geral</i>	16
1.1.2	<i>Objetivos Específicos</i>	16
2	FUNDAMENTAÇÃO TEÓRICA	18
2.1	Banco de Dados	18
2.2	<i>API</i>	18
2.3	<i>Low-Code</i>	19
2.4	<i>Workflow</i>	21
2.5	<i>Gartner</i>	21
2.6	Quadrante Mágico da Gartner	22
2.7	Plataformas de desenvolvimento <i>low-code</i>	23
2.7.1	<i>Íris Low-code Certidão</i>	24
2.7.2	<i>Zoho Creator</i>	25
2.7.3	<i>Outsystems</i>	26
2.7.4	<i>Mendix</i>	26
3	TRABALHOS RELACIONADOS	29
3.1	<i>Supporting the understanding and comparison of low-code development platforms</i>	29
3.2	<i>Characteristics and Challenges of Low-Code Development: The Practitioners Perspective</i>	30
3.3	<i>OLP — A RESTful Open Low-Code Platform</i>	32
3.4	Comparação entre trabalhos	32
4	PROCEDIMENTOS METODOLÓGICOS	34
4.1	Levantamento das plataformas de desenvolvimento <i>low-code</i>	34
4.2	Desenvolvimento de uma aplicação simples	34
4.3	Seleção das plataformas de desenvolvimento <i>low-code</i>	35
4.4	Definição dos atributos de comparação	35
4.4.1	<i>Critérios quantitativos</i>	35
4.4.1.1	<i>Tempo para criação do template de certificado</i>	35

4.4.2	<i>Critérios qualitativos</i>	36
4.4.2.1	<i>Recursos de banco de dados</i>	36
4.4.2.2	<i>Integração com API</i>	36
4.4.2.3	<i>Outras funcionalidades</i>	36
4.5	Comparação entre as plataformas	37
4.6	Análise dos resultados obtidos	37
5	EXPERIMENTOS E RESULTADOS	38
5.1	Levantamento das plataformas	38
5.1.1	<i>Seleção das plataformas de desenvolvimento low code</i>	39
5.1.2	<i>Validação das plataformas escolhidas</i>	39
5.1.3	<i>Dificuldades encontradas</i>	40
5.2	Definição dos atributos de comparação	41
5.2.1	<i>Critérios quantitativos</i>	41
5.2.1.1	<i>Tempo para criação do template do certificado</i>	41
5.2.2	<i>Critérios qualitativos</i>	43
5.2.2.1	<i>Recursos de banco de dados</i>	43
5.2.2.2	<i>Integração com API</i>	45
5.2.2.3	<i>Outras funcionalidades</i>	45
5.3	Catálogo com os resultados obtidos	49
5.4	Vídeos dos experimentos	49
6	CONCLUSÕES E TRABALHOS FUTUROS	50
	REFERÊNCIAS	51

1 INTRODUÇÃO

Nos últimos anos, o mercado de tecnologia vem se mostrando cada vez mais em uma crescente ascensão. Mesmo com a pandemia da Coronavírus (COVID-19), este mercado continuou aquecido e a demanda pelo desenvolvimento de *software* cresceu bastante. De acordo com TV e Abdulla (2022), durante a pandemia da COVID-19 as empresas precisaram criar *software* para conectar-se com seus funcionários e clientes para manter os processos do dia a dia.

Conforme o relatório elaborado pela Brasscom (2021), existe a projeção de uma demanda de 797 mil profissionais entre 2021 e 2025. Mesmo com o aumento na demanda no mercado de Tecnologia da Informação (TI), no Brasil, o país vem sofrendo cada vez mais com a falta de mão de obra qualificada e isso vem sendo chamado de apagão de profissionais de TI. Para Moraes (2022), as plataformas de desenvolvimento *low-code* surgiram como resposta a este problema.

De acordo com Sahay *et al.* (2020), as plataformas de desenvolvimento *low-code* tem como principal objetivo lidar com a carência de desenvolvedores qualificados, permitindo que usuários, mesmo com pouca experiência em programar, contribuam com o processo de transformação digital em uma empresa.

Ainda nas palavras de Sahay *et al.* (2020), as plataformas de desenvolvimento de *low-code* são fornecidas através do modelo *platform as a service* (PaaS) e permitem o desenvolvimento e a implantação de aplicativos totalmente funcionais utilizando interfaces gráficas e abstrações visuais que exigem o mínimo ou nenhum código.

Percebe-se que na literatura existe uma confusão na definição entre tecnologia *no-code* e *low-code*. Segundo TV e Abdulla (2022), embora essas tecnologias sejam tratadas de forma diferente, o *no-code* faz parte do *low-code*, com isso, tudo neste trabalho será tratado como *low-code*.

Diferente do que muitos desenvolvedores pensam, essas tecnologias não tornarão o desenvolvimento de *software* tradicional obsoleto. Esse tipo de solução, embora esteja bem evoluída, possui algumas limitações e por conta disso são normalmente utilizadas em situações específicas como, por exemplo, a validação de uma ideia.

No Quadro 1 pode-se verificar algumas vantagens e desvantagens de utilizar as plataformas *low-code*. Para Shridhar (2021), as ferramentas *low-code* possuem vantagens que tornam o desenvolvimento de *software* econômico, por não precisar contratar muitos desenvolvedores, traz mais agilidade ao desenvolvimento por possuir funcionalidades prontas e fácil de adicionar

no projeto, alta gerência de risco por permitir que as empresas façam mudanças rapidamente, devido a alguma mudança que venha surgir ao longo do desenvolvimento do projeto. Por fim, alta produtividade pelo fato da equipe não precisar se preocupar com códigos complexos.

Por outro lado, as desvantagens citadas por Shridhar (2021), vistas no Quadro 1, podem ser um fator considerável na decisão de iniciar um projeto com *low-code*. Como algumas plataformas não permitem que o usuário tenha acesso ao código-fonte ou a algum tipo de *backup* do banco de dados, o desenvolvedor se torna "refém" do fornecedor da ferramenta. Outro fator é que os componentes disponibilizados por algumas ferramentas não permitem customização e isso torna a criação de *layout* limitada. Opções de integrações limitadas também pode ser um fato grave porque isso pode forçar decisões que não são boas para o projeto. Por conta da crescente ascensão das plataformas *low-code*, a escassez de desenvolvedores aptos a trabalhar com tais ferramentas vem diminuindo, mas ainda é pode ser um problema, dependendo da plataforma escolhida.

Quadro 1 – Vantagens e desvantagens de ferramentas low-code

Vantagem	Desvantagem
Econômico	Confiança em terceiros
Agilidade	Customização limitada
Alta gerência de risco	Opções de integração limitadas
Alta produtividade	Escassez de Desenvolvedores

Fonte: Adaptação Shridhar (2021)

Dessa forma, pode-se perceber que as plataformas de desenvolvimento *low-code* tem tornado cada vez mais possível que empreendedores tirem as suas ideias do papel. Manning (2019) explica como foi o processo de desenvolvimento Zurich FaceQuote, um aplicativo para calcular o preço do seguro de vida, concluído em uma semana. Em Dubai foi possível digitalizar 250 serviços em apenas 3 meses (MENDIX, 2021). Devido a casos semelhantes aos apresentados, as ferramentas *low-code* tem chamando atenção e isso incentivou grandes empresas de tecnologia a investirem em suas próprias plataformas de desenvolvimento *low-code* como a Microsoft com o *Power Apps*, a Google com o *AppSheet* e a Amazon com o *Honey Code*.

Com base em uma análise feita na carta de serviços do cidadão, uma aplicação *web* que lista todos os serviços oferecidos ao cidadão pelo Estado do Ceará, foi observado que muitas secretarias disponibilizavam a geração de certificados e certidões. Desta forma, foi investigado como uma ferramenta *low-code* poderia ser utilizada para facilitar a criação de *templates* de certidões. Com isso, surgiu o interesse do Estado em criar sua própria solução *low-code* focada

neste objetivo.

Assim surgiu o *Iris Low Code Certidão*, que teve seu início em agosto de 2020 pelo projeto Governo Digital em parceria com o ÍRIS | Laboratório de Inovação e Dados do Governo do Ceará para Acelerar a Transformação Digital no Estado do Ceará.

Dessa forma, o presente trabalho tem como foco realizar um estudo comparativo entre plataformas de desenvolvimento *low-code* populares no mercado de TI e o *Iris Low Code Certidão*, uma solução *low-code* que ainda vem sendo mantida pelo ÍRIS | Laboratório de Inovação e Dados do Governo do Ceará.

1.1 Objetivos

Nesta seção, são descritos o objetivo geral e objetivos específicos do presente trabalho.

1.1.1 Objetivo Geral

O objetivo do trabalho é realizar um estudo comparativo entre as plataformas de desenvolvimento *low-code* mais populares no mercado de TI e o *Iris Low Code Certidão* com o intuito de identificar critérios relevantes que podem contribuir com a evolução do *Iris Low Code Certidão*.

1.1.2 Objetivos Específicos

1. Selecionar as plataformas de desenvolvimento *low-code* a serem comparadas;
2. Definir os critérios de avaliação que serão usados para a comparação;
3. Comparar as plataformas selecionadas com base nos critérios de avaliação estabelecidos;
4. Identificar melhorias que podem ser aplicadas ao *Iris Low Code Certidão* com base na comparação;

Os próximos capítulos estão organizados da seguinte maneira: No Capítulo 2 apresenta a fundamentação teórica e os conceitos que embasam as abordagens propostas neste trabalho; No Capítulo 3 trata dos trabalhos relacionados, com descrição e comparação de projetos e pesquisas que possuem aspectos similares aos especificados neste trabalho; No Capítulo 4

apresentada a metodologia usada e descrição dos passos seguidos para a realização deste trabalho; No Capítulo 5 são apresentados os resultados obtidos a partir da metodologia usada; Por último, no Capítulo 6 são apresentadas as conclusões a partir dos resultados obtidos e trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são apresentados os conceitos fundamentais utilizados neste trabalho. Na Seção 2.1 é apresentado o conceito de banco de dados; na Seção 2.2, é apresentado o conceito de *API*; na Seção 2.4, é apresentado o conceito de *workflow*; na Seção 2.5, falamos sobre a empresa de consultoria *Gartner*; na Seção 2.6, falamos sobre o quadrante mágico do *Gartner* na Seção 2.3, É apresentado o conceito de *low-code*; na Seção 2.7, é apresentado as plataformas de desenvolvimento *low-code* usadas como objeto de estudo neste trabalho.

2.1 Banco de Dados

Na era atual, é notável o impacto significativo dos bancos de dados nas várias áreas onde a informática está presente. Segundo Elmasri *et al.* (2005) um banco de dados é uma coleção de dados relacionados que representa informações reais como, por exemplo, nomes, endereços e números de celulares. Esses dados são armazenados de forma estruturada e podem ser gerenciados por Sistemas de Gerenciamento de Banco de Dados (SGBD).

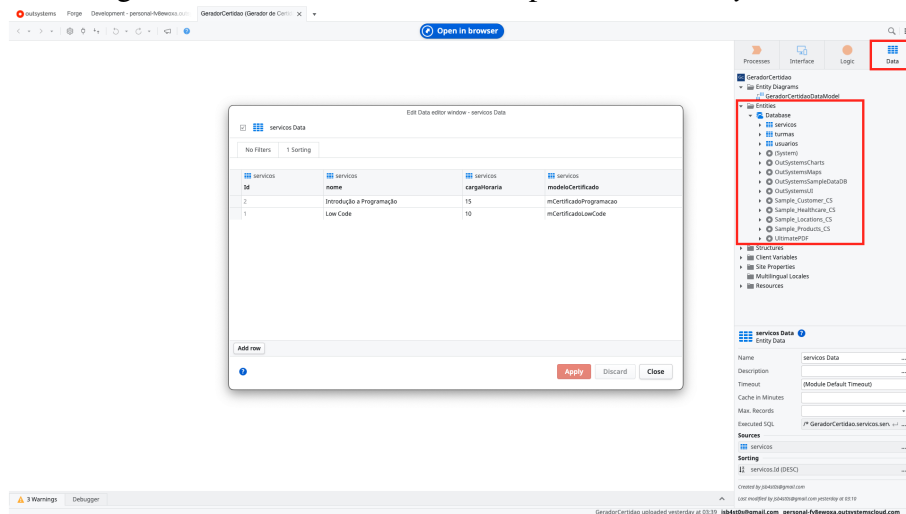
SGBD é um software que possibilita aos usuários criar, manter e gerenciar um banco de dados de forma eficiente e organizada (ELMASRI *et al.*, 2005). Em outras palavras, esse sistema oferece funcionalidades que permitem a criação do banco de dados, das estruturas dos dados, a definição das relações entre elas e a realização de operações de consulta, atualização e recuperação dos dados.

As plataformas *low-code* usadas como objeto de estudo neste trabalho possuem o banco de dados integrado a sua ferramenta e, como a maioria delas não permite acesso direto ao banco de dados, é normalmente disponibilizada uma tela para consulta e manipulação dos dados. Na Figura 1 pode-se visualizar a tela de consulta dos dados cadastrados na aplicação construída na plataforma da *Outsystem*.

2.2 API

Interfaces de programação de aplicativos (*API*), é uma tecnologia que possibilita sistemas diferentes se comuniquem de forma eficiente e segura. Segundo Stylos e Myers (2007), o uso de *APIs* ajuda os programadores a economizar tempo, reutilizando funcionalidades prontas que poderiam ser difíceis de implementar do zero. Com isso, é possível expandir as

Figura 1 – Consulta dos dados na plataforma *Outsystem*



Fonte: Elaborado pelo autor.

funcionalidades de um sistema como, por exemplo, integrando uma consulta de dados de um usuário, integração com mapa, entre outras possibilidades.

Embora o uso de *API* ofereçam benefícios, problemas relacionado ao desempenho devido à utilização da rede para se conectar com *API* de terceiros, complexidade na integração e dependências de terceiros são problemas que devem ser considerados na utilização de *API*.

2.3 Low-Code

Low-code, que na tradução literal significa pouco código, são ferramentas que permitem os desenvolvedores criarem soluções de forma rápida e eficaz com poucas linhas de código. Nas palavras de Alexander (2021b) “*Low-code* é uma abordagem de desenvolvimento de software que permite a entrega de aplicativos mais rapidamente e com o mínimo de codificação manual”.

Segundo Alexander (2021a) “À medida que a pressão sobre a TI para entregar mais e mais rápido aumenta, muitas organizações estão se voltando para o desenvolvimento *low-code* para superar as barreiras impostas pela escassez de desenvolvedores e ferramentas de desenvolvimento tradicionais”. Na Tabela 1 é possível verificar o crescimento da receita de tecnologias de desenvolvimento *low-code* através de pesquisas de mercado do realizado pelo Gartner (2022) nos anos de 2021, 2022 e as projeções para os anos de 2023 e 2024.

Em 2019, a *Outsystem*, empresa que vende solução *low-code*, realizou uma pesquisa com mais de 3300 profissionais de TI, a Tabela 2 é uma adaptação de um dos tópicos discutido nessa pesquisa. O objetivo desse tópico era entender o que estava movendo as empresas a

Tabela 1 – Receita de tecnologias de desenvolvimento *low-code* (milhões de dólares americanos)

	2021	2022	2023	2024
Plataformas de aplicações <i>low-code</i> (LCAP)	\$6,32	\$7,97	\$9,96	\$12,35
Automação de processos de negócios (BPA)	\$2,42	\$2,59	\$2,76	\$2,94
Plataformas de Desenvolvimento Multi-experiência (MDXP)	\$2,08	\$2,51	\$3,00	\$3,56
Automação Robótica de Processos (RPA)	\$2,35	\$2,89	\$3,40	\$3,88
Plataforma de integração como serviço (iPaaS)	\$4,68	\$5,67	\$6,67	\$7,84
Plataforma de desenvolvimento de automação cidadã (CADP)	\$0,55	\$0,73	\$0,95	\$1,23
Outras tecnologias de desenvolvimento <i>low-code</i> (LCD)	\$0,09	\$0,11	\$0,13	\$0,15
Total	\$18,50	\$22,46	\$26,87	\$31,95

Fonte: Adaptação Gartner (2022)

buscarem soluções *low-code*. Como podemos perceber, os pontos que mais tiveram destaques nesse tópico foram acelerar a inovação/transformação digital, aumentar a capacidade de resposta ao negócio e reduzir a dependência de habilidades técnicas difíceis de contratar, o que de fato são os principais objetivos de uma ferramenta *low-code*.

Tabela 2 – Principais razões pelo qual as organizações usam *low-code*

Critério	Valor em %
Acelerar a inovação/transformação digital	66%
Aumentar a capacidade de resposta mercado	66%
Reduzir a dependência de habilidades técnicas difíceis de contratar	45%
Descontinuar sistemas legados	28%
Evitar rotatividade de tecnologia	22%
Permitir que <i>citizen developers</i> melhorem os processos internos	20%
Outros	2%

Fonte: Adaptação Outsystems (2019)

Outro tópico abordado na pesquisa realizada pela *Outsystem* é porque algumas organizações ainda não estão usando soluções *low-code*. A Tabela 3 é uma adaptação das respostas desse tópico no relatório.

Tabela 3 – Por que algumas organizações ainda não estão usando *low-code*?

Critério	Valor em %
Falta de conhecimento sobre as plataformas <i>low-code</i>	47%
Não acreditam que poderiam criar os tipos de aplicativos que precisam	32%
Preocupação com o <i>vendor lock-in</i>	37%
Preocupação com a segurança	25%
Preocupação com a escalabilidade	28%
Outros	10%

Fonte: Adaptação Outsystems (2019)

O motivo pelo qual a falta de conhecimento sobre ferramentas *low-code* está em destaque pode ser pelo fato de que embora essas plataformas já estejam há alguns anos no mercado, elas só ganharam mais destaque devido à grande demanda do mercado de TI nos últimos anos e com isso estão sendo descobertas agora.

2.4 *Workflow*

Workflow são tarefas executadas em sequência ou em paralelo para atingir um objetivo específico. Em plataformas *low-code* os *workflows* são usados para criar fluxos personalizados quando determinadas ações são realizadas.

Segundo Sahay *et al.* (2023), as plataformas de desenvolvimento *low-code* fornecem recursos de modelagem de *workflow* que permite definir processos ou automatizar tarefas eficientemente, agilizando os processos da organização e melhorando a produtividade.

A Figura 2 mostra um exemplo de *workflow* criado na plataforma do *Outsystem*. No exemplo apresentado, o *workflow* é executado quando um usuário tentar emitir um certificado. Inicialmente é feita uma verificação para validar o usuário. Após isso, caso o usuário seja válido, é realizada a ação de download do arquivo. Caso contrário, o usuário é notificado com um erro.

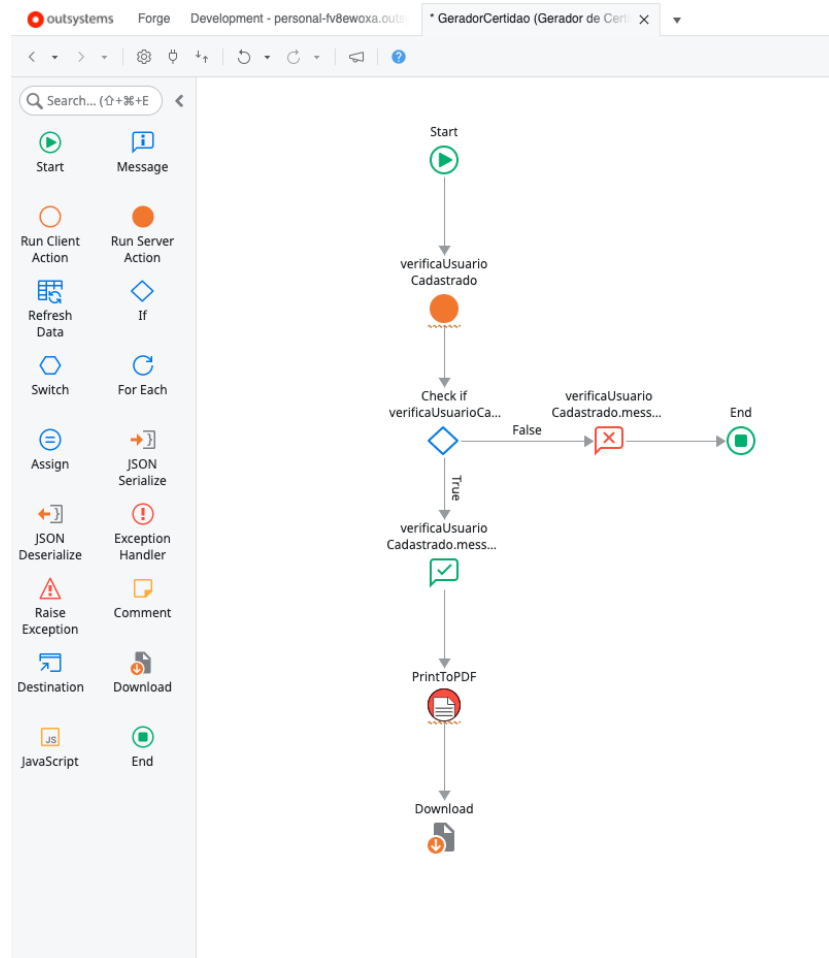
2.5 *Gartner*

Fundada em 1979, o *Gartner* é uma empresa de consultoria e pesquisa especializada em TI conhecida por conta de suas pesquisas, relatórios e o quadrante mágico, que avalia e classifica fornecedores e soluções do mercado de TI.

Segundo O’Leary (2008), o *Gartner* fornece pesquisas e *insights* sobre sistemas para seus clientes visando auxiliar as empresas a compreender como essas ferramentas podem impactar seus negócios e tomar decisões sobre quais tecnologias adotar.

Segunda a ficha corporativa disponível no site do *Gartner*, a empresa conta com mais de 19.500 colaboradores em cerca de 85 escritórios ao redor do mundo. Sua reputação foi construída com base nas metodologias de pesquisa rigorosas e independentes, desenvolvidas por uma equipe global de 2.500 especialistas em pesquisa e consultoria.

Figura 2 – Exemplo de *workflow* criado na plataforma da *Outsystem*



Fonte: Elaborado pelo autor.

2.6 Quadrante Mágico da *Gartner*

Segundo *Gartner* (2023), o quadrante mágico é um estudo que analisa um mercado específico e fornece uma visão sobre os concorrentes nesse mercado. O quadrante mágico usado neste trabalho como fonte de pesquisa para entender quais são as principais empresas nesse mercado é o quadrante mágico para plataformas de aplicativos *low-code* corporativos.

Para avaliar as plataformas, *Vincent et al.* (2022) representa graficamente a posição das empresas com base em critérios como visão de mercado e capacidade de execução. O Quadro 2 apresenta todos os critérios usados para avaliar a capacidade de execução e o Quadro 3 mostra critérios para avaliar a visão de mercado.

Após passar pela avaliação descritos anteriormente, as plataformas *low-code* são separadas em 4 categorias, sendo elas: líderes, desafiantes, visionários e *players* de nicho.

Quadro 2 – Critérios de avaliação relacionados a capacidade de execução

Critério	Definição	Ponderação
Produto/Serviço	Oferta do fornecedor, incluindo qualidade e recursos	Alto
Execução de Vendas/Precificação	Habilidades e suporte em atividades de pré-venda e negociação	Alto
Viabilidade geral	Saúde financeira da organização e sucesso da unidade de negócios	Médio
Capacidade de resposta/registro do mercado	Adaptabilidade e histórico de resposta às mudanças do mercado	Médio
Execução de Marketing	Efetividade dos programas de marketing e promoção da marca	Baixo
Experiência do Cliente	Suporte e programas para o sucesso dos clientes	Médio
Operações	Capacidade da organização em cumprir seus objetivos de forma eficiente	Baixo

Fonte: Adaptação Vincent *et al.* (2022)

Quadro 3 – Critérios de avaliação relacionados a visão de mercado

Critério	Definição	Ponderação
Compreensão do mercado	Capacidade de compreender as necessidades dos compradores	Alto
Estratégia de marketing	Mensagens claras e diferenciadas	Médio
Estratégia de vendas	Abordagem para ampliar o alcance do mercado	Médio
Estratégia de oferta (produto)	Ênfase na diferenciação e funcionalidade dos produtos	Alto
Modelo de negócios	Proposta de negócio sólida	Médio
Estratégia Vertical/Setorial	Direcionamento para atender segmentos específicos	Baixo
Inovação	Uso estratégico de recursos	Alto
Estratégia Geográfica	Direcionamento para atender necessidades em diferentes geografias	Baixo

Fonte: Adaptação Vincent *et al.* (2022)

Vincent *et al.* (2022) define as categorias do quadrante mágico da seguinte forma:

1. Líderes: são bons em executar seus negócios, com uma visão abrangente e com isso conseguem atender vários tipos de negócios.
2. Desafiantes: são bons em executar os seus negócios, porém não possuem uma visão estratégica sobre como entrar em novos mercados.
3. Visionários: possuem ideias inovadoras, mas podem ter dificuldades na execução dessas ideias.
4. *Players* de nicho: são especializados em áreas específicas do mercado ou possuem presença regional.

Na Figura 3 pode-se observar como as empresas avaliadas foram categorizadas no último relatório quadrante mágico disponibilizado 31 de dezembro de 2022. Para auxiliar na tomada de decisão, para cada plataforma do quadrante mágico, o Gartner destaca os principais pontos de forças e cuidados das plataformas avaliadas em seu relatório.

2.7 Plataformas de desenvolvimento *low-code*

De acordo com Bock e Frank (2021), uma plataforma de desenvolvimento *low-code* é um novo ambiente de desenvolvimento de *software* que surgiu não apenas para oferecer

Figura 3 – Gartner - Quadrante mágico para plataformas de aplicativos *low-code* corporativos



Fonte: Vincent *et al.* (2022)

aumento na produtividade, mas também gera novas formas de promover negócios de TI. Segundo Di Ruscio *et al.* (2022) essas plataformas oferecem suporte ao desenvolvimento de aplicativos *web*, *desktop* e dispositivos móveis.

2.7.1 Íris Low-code Certidão

O *Íris Low-code Certidão* é uma plataforma usada pelo Governo do Estado do Ceará que permite aos servidores dos órgãos do Estado criarem formulários e *templates* de documentos visando simplificar a geração de certificados e certidões. A ferramenta está em fase de desenvolvimento no ÍRIS | Laboratório de Inovação e Dados do Governo do Ceará, mas possui uma versão em produção no ambiente da Secretaria de Educação do Estado do Ceará.

O *front-end* do sistema é uma aplicação *web* que consome dados do *back-end* e de outras fontes de dados, como *API* de terceiros ou das planilhas *Google*. A aplicação foi desenvolvida com *Nextjs*, um *framework* que adiciona várias funcionalidades ao *Reactjs* que é uma biblioteca *front-end* do *javascript*.

O *back-end* foi construído com *Nodejs* e o *Parse Server* que é uma plataforma de código aberto que fornece uma estrutura pronta para uso, permitindo gerenciar banco de dados, autenticação de usuários e armazenamento de arquivos. Quanto ao banco de dados, por conta do *Parse Server*, é possível usar *PostgreSQL* ou *MongoDB*. Atualmente o *Íris Low-code Certidão* está em funcionamento no ambiente da Secretaria de Educação do Estado do Ceará com o banco de dados do *PostgreSQL*. Na Figura 4 pode-se observar a tela inicial do *Íris Low-code Certidão*.

Figura 4 – Representação (*Íris Low-code Certidão*)



Fonte: Elaborado pelo autor.

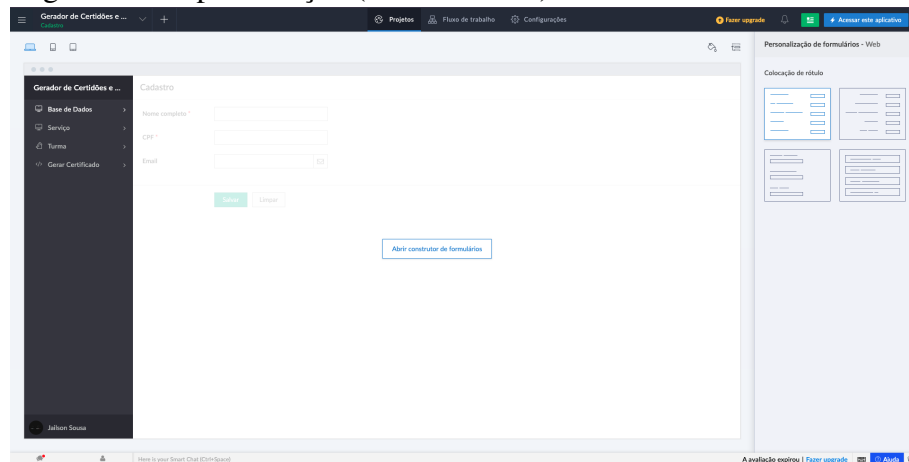
2.7.2 Zoho Creator

Zoho Creator é uma plataforma de desenvolvimento *low-code* disponível na nuvem que permite criar aplicativos *web* e *mobile* de forma visual e intuitiva. Com recursos como integração com serviços de terceiros e automação de processos de negócios, a plataforma atende às necessidades de usuários com diferentes níveis de habilidade técnica.

Durante o estudo da plataforma, foi percebido que a ferramenta possui o plano pago e o gratuito com algumas limitações. Essas são algumas limitações identificadas: (i) uma aplicação por conta; (ii) um usuário por aplicação, (iii) 1.000 registros no banco de dados; (iv) 250 MB de armazenamento, (v) restrição de algumas funcionalidades.

Segundo o Vincent *et al.* (2022), o *Zoho Creator* é uma plataforma madura e atende às necessidades de diversos profissionais no desenvolvimento de aplicativos. Mesmo sendo popular entre as pequenas e médias empresas, o *Zoho Creator* possui recursos corporativos que o tornam uma opção atraente para diversas organizações. Na Figura 5 pode-se observar a tela inicial de um projeto no *Zoho Creator*.

Figura 5 – Representação (Zoho Creator)



Fonte: Elaborado pelo autor.

2.7.3 Outsystems

Outsystems é uma plataforma de desenvolvimento *low-code*, disponível em ambiente *cloud* e *on-premise*, que permite criar soluções *web* e *mobile*. A ferramenta usada pela *Outsystems* para criar as aplicações é a *service studio*, que está disponível para *windows* e *macOs*.

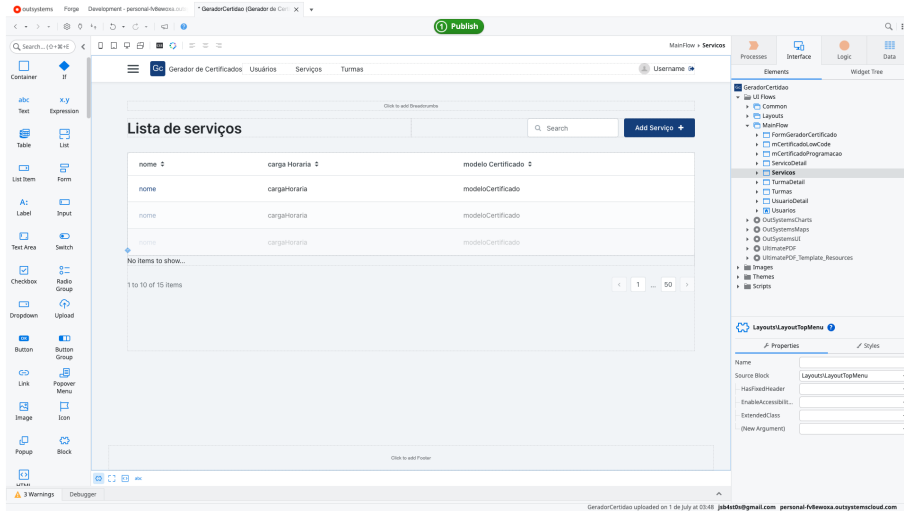
Durante o estudo da plataforma, foi percebido que a ferramenta possui o plano pago e o gratuito com algumas limitações. Essas são algumas limitações identificadas: (i) uma aplicação por conta; (ii) não garante tempo de atividade; (iii) ambiente é excluído após 15 dias de ausência; (iv) 2 GB de armazenamento.

Segundo o Vincent *et al.* (2022), a *OutSystem* é uma líder no mercado de plataformas de desenvolvimento *low-code*. Sua abordagem permite a criação de aplicativos e componentes de qualquer complexidade com facilidade, segurança robusta, e produtividade para os desenvolvedores. Os seus clientes vão desde pequenas a grandes empresas, principalmente nos setores bancário, financeiro, de seguros e de serviços. Na Figura 6 pode-se observar a tela de edição da *OutSystem*

2.7.4 Mendix

Mendix é uma plataforma de desenvolvimento *low-code*, atualmente é uma das principais concorrentes da *Outsystem* nesse mercado. A plataforma possui ambiente na *cloud* e *on-premise* e permite criar soluções *web* e *mobile*.

Figura 6 – Representação (*Outsystems*)

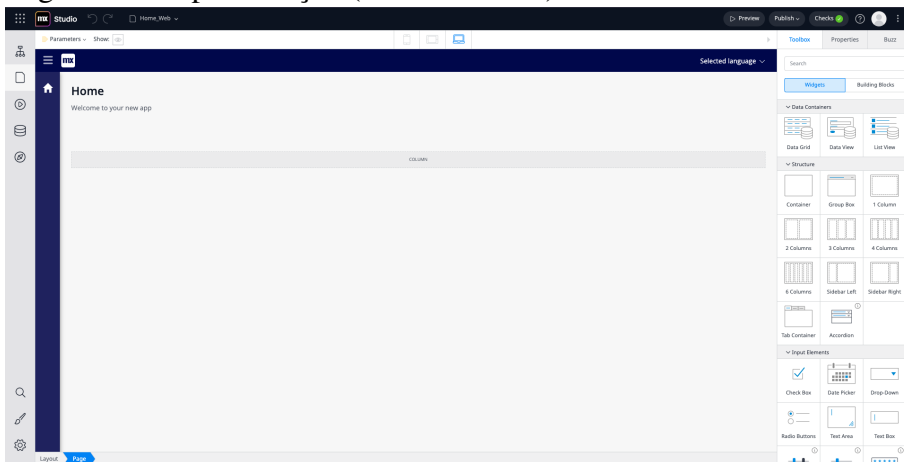


Fonte: Elaborado pelo autor.

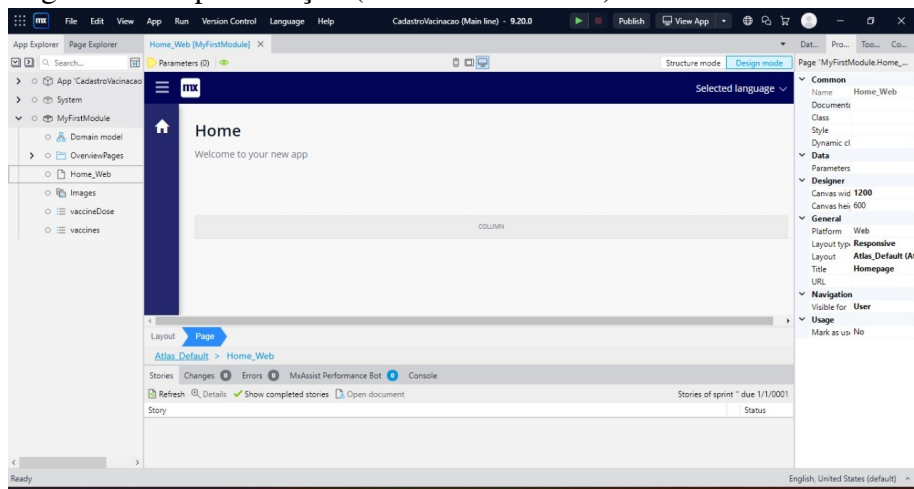
A plataforma possui duas ferramentas de desenvolvimento gratuitas, o *mendix studio* que é uma ferramenta *web* ideal para iniciantes por ser uma versão mais simples e o *mendix studio pro* que é uma ferramenta *desktop* disponível apenas para *windows* e que possibilita uma experiência mais completa. Na Figura 7 pode-se observar a tela inicial do *mendix studio* e na Figura 8 é mostrado a tela inicial do *mendix studio pro*.

Durante o estudo da plataforma, foi percebido que a ferramenta possui o plano pago e o gratuito com algumas limitações. Essas são algumas limitações identificadas: (i) não garante tempo de atividade; (ii) 500 MB de banco de dados; (iii) 1 GB de armazenamento; (iv) modo de suspensão após cerca de uma hora de inatividade;

Figura 7 – Representação (*Mendix Studio*)



Fonte: Elaborado pelo autor.

Figura 8 – Representação (*Mendix Studio Pro*)

Fonte: Elaborado pelo autor.

3 TRABALHOS RELACIONADOS

Neste capítulo, são descritos os trabalhos relacionados ao presente trabalho.

3.1 *Supporting the understanding and comparison of low-code development platforms*

No trabalho Sahay *et al.* (2020), os autores realizaram uma pesquisa visando extrair funcionalidades relevantes de diferentes plataformas de desenvolvimento *low-code* e com isso mapear características que podem auxiliar os desenvolvedores na tomada de decisão sobre qual plataforma melhor atende os requisitos do seu projeto.

As funcionalidades foram coletadas de 8 plataformas *low-code* que foram escolhidas a partir do relatório do Gartner (VINCENT *et al.*, 2019). As plataformas escolhidas para esse estudo foram as seguintes: *OutSystem*, *Mendix*, *Zoho Creator*, *Microsoft PowerApps*, *Google App Maker*, *Kissflow*, *Salesforce App Cloud* e o *Appian*.

Os autores também fornecem uma visão geral sobre as plataformas de desenvolvimento *low-code*, os principais componentes dessas ferramentas e todas as etapas que envolvem o desenvolvimento de aplicações *low-code*. Ao analisar as plataformas *low-code* escolhidas na pesquisa e identificar suas diferenças e semelhanças, os autores criaram uma taxonomia para organizar as funcionalidades. No Quadro 4 são apresentados os detalhes sobre esses grupos.

Por fim, os autores usaram a taxonomia pertencentes a cada grupo apresentado no Quadro 4 para realizar a comparação. Além disso, foi apresentado um breve resumo sobre os recursos e a capacidade de cada plataforma analisada, bem como alguns aspectos adicionais que podem ser considerados na tomada de decisão sobre qual plataforma deve ser escolhida e um breve relato de experiência de uma aplicação de *benchmark* que foi desenvolvida.

Na conclusão, os autores reforçam que foi analisado oito plataformas de desenvolvimento *low-code* para identificar seus pontos em comum e suas diferenças, além de definir um conjunto de características usada para comparar as plataformas e que apresentaram um breve relato de experiência para discutir alguns recursos essenciais de cada plataforma, limitações e desafios identificados durante o desenvolvimento de um aplicativo de *benchmark*. Além disso, os autores destacam que sem a disponibilidade de uma estrutura conceitual apropriada, compreender e comparar várias plataformas pode ser uma tarefa desafiadora.

Como trabalhos futuros, os autores planejam refinar a taxonomia proposta e acrescentar outras plataformas de desenvolvimento *low-code* para obter um conjunto de recursos

Quadro 4 – Grupo de características

Grupo	Descrição
Interface gráfica do usuário	Está relacionado as funcionalidades disponíveis da plataforma para suportar as interações com o usuário.
Suporte a interoperabilidade com serviços externos e fontes de dados	Está relacionado à possibilidade de interagir com serviços externos.
Suporte a segurança	Está relacionado aos aspectos de segurança da aplicação desenvolvida.
Suporte ao desenvolvimento colaborativo	Está relacionado aos modelos de colaboração (por exemplo, online e offline)
Suporte a reutilização	Está relacionado a possibilidade de reutilizar componentes já desenvolvidos.
Suporte à escalabilidade	Está relacionado a possibilidade de permitir que os desenvolvedores escalem suas aplicações.
Mecanismos de especificação de lógica de negócios	Está relacionado aos meios fornecidos para especificar a lógica de negócios do software.
Suporte a implantação	Está relacionado aos mecanismos disponíveis para implantação do aplicativo modelado.

Fonte: Adaptação Sahay *et al.* (2020)

abrangente e validado.

3.2 Characteristics and Challenges of Low-Code Development: The Practitioners Perspective

No trabalho Luo *et al.* (2021), os autores exploram as características e os desafios enfrentado pelo *Low-Code Development* (LCD) na perspectiva dos desenvolvedores de *software*. Os dados analisados neste trabalho foram coletados do *stack overflow* e o *reddit* que são duas comunidades de desenvolvedores.

Na metodologia, os autores estabeleceram o objetivo da pesquisa com base na abordagem *Goal Question Metric* que é uma abordagem de medição de software que relaciona metas, perguntas e métricas para avaliar o progresso e o desempenho de projetos. No Quadro 5 estão as questões de pesquisa levantado neste trabalho. Os dados necessários para responder às questões de pesquisa foram extraídos das comunidades no período de janeiro de 2021 a abril de 2021. Os dados coletados são compostos por opiniões e a experiência dos profissionais que frequentam essas comunidades.

Como resultado, foi extraído e analisado os dados de mais de 301 postagens das comunidades. Todas as questões de pesquisa foram respondidas, além de ter sido criada uma sessão para discutir cada questão de pesquisar suas respostas.

Quadro 5 – Questões de pesquisa

1	Qual é a compreensão de LCD?
2	Quais plataformas estão usando LCD?
3	Quais linguagens de programação são usadas no LCD?
4	Quais são as principais unidades de implementação em LCD?
5	Quais são as tecnologias suportadas usadas no LCD?
6	Que tipos de aplicativos são desenvolvidos pela LCD?
7	Quais são os domínios que usam LCD?
8	Quais são os benefícios do LCD?
9	Quais são as limitações e desafios do LCD?

Fonte: Adaptação Luo *et al.* (2021)

Na conclusão, os autores reforçam que o conceito de desenvolvimento *low-code* não é novo, mas apenas recentemente vem recebendo muita atenção na indústria por ajudar desenvolvedores a entregar aplicativos rapidamente escrevendo pouco código. Além disso, os autores afirmam que as principais conclusões da pesquisa foram as seguintes:

1. Embora não haja uma definição clara sobre o desenvolvimento *low-code*, os usuários costumam descrever essas ferramentas conforme o seu entendimento que, na maioria das vezes, se resume a uma ferramenta que fornece uma interface gráfica que possibilita os usuários arrastarem e soltarem componente e precisam de pouco ou nenhum código.
2. Modelos prontos para uso disponíveis no desenvolvimento *low-code* facilitam o aprendizado e ajudam a acelerar o desenvolvimento.
3. Diferentes plataformas de desenvolvimento *low-code* suportam criar diferentes tipos de aplicações (por exemplo, web ou mobile) e diferentes partes de aplicativos (por exemplo, fluxos de trabalho, integração).
4. O desenvolvimento *low-code* é favorecido nos domínios com a necessidade de automatizar os fluxos de trabalho.
5. Embora as plataformas de desenvolvimento *low-code* acelerem a criação de *software*, os desenvolvedores sofrem por não conseguirem acesso ao código-fonte e pelo vendor lock-in. Além disso, é necessário considerar qual plataforma de desenvolvimento *low-code* é apropriada para o seu projeto.

Como trabalhos futuros, os autores planejam ampliar o estudo aumentando o conjunto de dados e usando outros métodos de coleta, como, por exemplo, um questionário e

entrevista. Além disso, eles pretendem aprofundar-se em outros aspectos das plataformas de desenvolvimento *low-code* como, por exemplo, em qual situação um benefício poderia se tornar uma limitação e como evitar isso.

3.3 OLP — A RESTful Open Low-Code Platform

No trabalho Cruz *et al.* (2021), os autores propõem criar uma plataforma de desenvolvimento *low-code*. Para isso, são apresentados os requisitos funcionais e não funcionais e uma proposta de arquitetura. Além disso, é apresentada uma revisão do estado da arte das plataformas *low-code* e *no-code* e todas as lições aprendidas durante o desenvolvimento do projeto.

Na revisão do estado da arte, os autores abordaram toda a evolução das linguagens de programação tradicionais até chegar na tecnologia *low-code* e *no-code*. Além disso, também é abordado as vantagens e desvantagens das tecnologias *low-code* e *no-code* e suas principais características. Os autores também apresentação os requisitos funcionais e não funcionais e uma proposta de arquitetura para a plataforma proposta pelo trabalho.

Como resultado, os autores apresentaram uma prova de conceito do que foi proposto. Embora o escopo da plataforma desenvolvida se limite a aplicações web e a persistência dos dados e consulta seja por meio de *API REST*, a plataforma consegue disponibilizar todo o código-fonte da aplicação.

Na conclusão, os autores reconheceram a dificuldade de desenvolver uma plataforma *low-code* devido à falta de literatura disponível. Para os autores, a plataforma de desenvolvimento *low-code* proposta nesse trabalho foi construída para entender as dificuldades práticas de criar uma plataforma do zero.

Como trabalhos futuros, os autores propõem que seja melhorada a usabilidade e a automatização de alguns processos e a possibilidade de armazenar dados nativamente. Além disso, outra sugestão dos autores é permitir que usuários escrevam código *HTML* e adicionar suporte a outros tipos de aplicação, como aplicativos móveis.

3.4 Comparação entre trabalhos

Como feito em Sahay *et al.* (2020), o presente trabalho realiza pesquisas nas plataformas de desenvolvimento *low-code* visando identificar as funcionalidades relevantes e as plataformas escolhidas serão do Vincent *et al.* (2022). Entretanto, diferentemente de Sahay *et*

al. (2020) será feito uma análise nas ferramentas de modo a verificar quais serão usadas como objeto estudo neste trabalho.

Diferente do que foi executado em Luo *et al.* (2021), este trabalho não planeja coletar e analisar dados de comunidades de desenvolvedores, contudo será verificado todos os resultados de Luo *et al.* (2021) visando identificar funcionalidades nas plataformas que serão usadas como objeto de estudo neste trabalho que poderiam ser implementadas no Íris *Low-code* Certidão.

Por fim, diferente do que foi executado em Cruz *et al.* (2021), este trabalho não apresenta nenhuma proposta de criação de uma plataforma de desenvolvimento *low-code*, contudo, será analisado os requisitos e a arquitetura proposta no trabalho Cruz *et al.* (2021) visando identificar melhorias para o *Íris Low-Code Certidão*.

No Quadro 6 podem ser observadas características elencadas a partir dos trabalhos relacionados apresentados anteriormente e a sua comparação ao projeto exposto no presente trabalho. Os critérios de comparação estão dispostos da seguinte maneira: (i): quantidade de plataformas comparadas; (ii): definição da lista de plataformas comparadas; (iii): definição dos critérios de avaliação; (iv): propõe melhorias para as plataformas.

Quadro 6 – Comparação entre os trabalhos relacionados e o proposto.

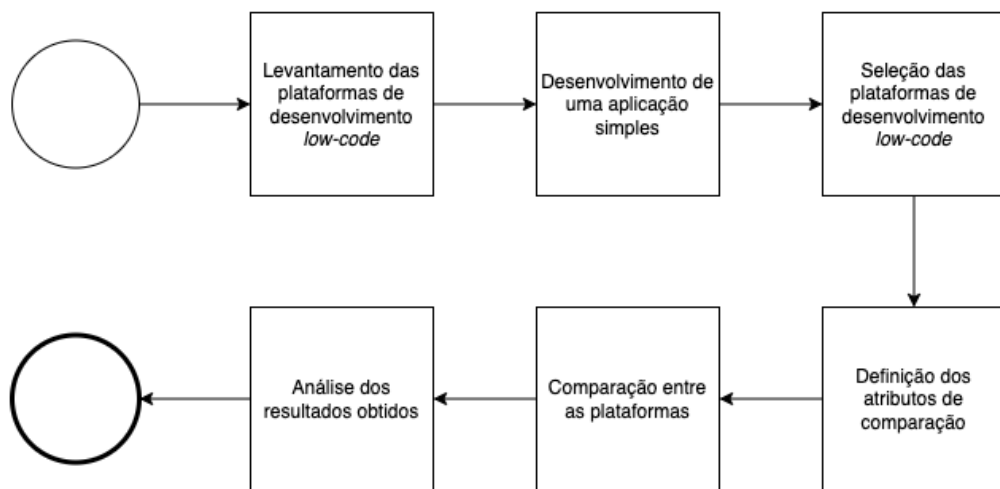
Trabalhos	Quantidade de plataformas comparadas	Definição da lista de plataformas comparadas	Definição dos critérios de avaliação	Propõe melhorias para as plataformas
(SAHAY <i>et al.</i> , 2020)	8	Através do relatório do Gartner e do Forrester	Proposta pelo autor após utilizar as plataformas	Não
(LUO <i>et al.</i> , 2021)	-	Por pesquisa no <i>stackoverflow</i> e no <i>reddit</i>	Através da extração das informações de tópicos nas comunidades de desenvolvedores	Não
(CRUZ <i>et al.</i> , 2021)	-	Não possui lista de plataformas	Proposta pelo autor	Não
Este trabalho	4	relatório do Gartner e o <i>Íris Low-code</i> Certidão	Proposta pelo autor após utilizar as plataformas	Sim

Fonte: Elaborado pelo autor.

4 PROCEDIMENTOS METODOLÓGICOS

Neste capítulo são apresentados os passos a serem realizados para execução deste trabalho. A Figura 9 apresenta os seguintes passos para a execução do trabalho: (i): Levantamento das plataformas de desenvolvimento *low-code*; (ii): Desenvolvimento de uma aplicação simples; (iii): Seleção das plataformas de desenvolvimento *low-code*; (iv): Definição dos atributos de comparação; (v): Comparação entre as plataformas; (vi): Análise dos resultados obtidos;

Figura 9 – Fluxo das atividades que serão executadas



Fonte: Elaborado pelo autor.

4.1 Levantamento das plataformas de desenvolvimento *low-code*

Esta seção descreve como será realizado o levantamento das plataformas de desenvolvimento *low-code*. Nesta etapa, será feito o levantamento das plataformas apresentadas no relatório do quadrante mágico para plataformas de aplicativos *low-code* corporativo publicado por Vincent *et al.* (2022) em 31 dezembro 2022.

O objetivo desse levantamento é conhecer as plataformas apresentadas no relatório e identificar quais não possuem algum impeditivo que impossibilite a plataforma de se tornar um objeto de estudo para este trabalho.

4.2 Desenvolvimento de uma aplicação simples

Nesta etapa será desenvolvida uma aplicação simples, em cada plataforma apresentada no relatório do quadrante mágico do Vincent *et al.* (2022), que simule o processo de emissão

de certificado do *Íris Low-code Certidão* o qual é o objeto de estudo que será comparado com as outras plataformas. Caso a plataforma não possua um plano gratuito, ou seja, encontrado algo que inviabilize o estudo da ferramenta, ela será descartada do estudo. Os requisitos funcionais mínimos que cada aplicação deve cumprir são os seguintes:

1. Cadastrar usuários que poderão solicitar o certificado
2. Cadastrar serviço para emissão de certificado
3. Vincular usuário a um serviço
4. Criar *template* de certificado para usar na emissão do certificado
5. Permitir que o usuário faça a emissão do certificado do serviço vinculado no formato PDF sem passar por uma autenticação

4.3 Seleção das plataformas de desenvolvimento *low-code*

Nesta etapa será selecionado as plataformas que serão objeto de estudo neste trabalho. As plataformas escolhidas serão aquelas que possibilitem desenvolver uma aplicação simples, cumprindo todos os requisitos funcionais descritos na Seção 4.2.

4.4 Definição dos atributos de comparação

Nesta seção são apresentados todos os atributos que servem como guia de comparação das plataformas de desenvolvimento *low-code*. Para melhor organização, os atributos foram separados em dois grupos, os critérios quantitativos e qualitativos.

4.4.1 Critérios quantitativos

4.4.1.1 Tempo para criação do *template* de certificado

Neste fator, é avaliado o tempo necessário para a criação de um *template* de certificado que será usado no momento da emissão. Para cada plataforma será gravado um vídeo contendo todo o processo de criação do *template*.

Embora o tempo necessário para a criação do *template* dependa da experiência e do conhecimento sobre a plataforma, para obter resultados mais imparciais possíveis, será feito um estudo mais aprofundado nas plataformas para definir o caminho mais curto para a realização

dessa medição.

4.4.2 Critérios qualitativos

4.4.2.1 Recursos de banco de dados

Para este atributo é verificado se as plataformas de desenvolvimento *low-code* disponibilizam recursos relacionados ao banco de dados. As funcionalidades verificadas em cada plataforma serão as seguintes:

1. Permite acesso ao banco de dados
2. Possui banco de dados interno
3. Permite conexão com um banco de dados externo
4. Permite importar o banco de dados
5. Permite exportar o banco de dados

4.4.2.2 Integração com API

Para este atributo é verificado se as plataformas de desenvolvimento *low-code* disponibilizam recursos relacionadas a integração com *API*. As funcionalidades verificadas em cada plataforma serão as seguintes:

1. Permite consumir os dados por *API*
2. Permite integrar alguma funcionalidade com uma *API*
3. Permite consumir dados das Planilhas *Google*

4.4.2.3 Outras funcionalidades

Para este atributo é verificado em cada plataformas de desenvolvimento *low-code*, durante o processo de desenvolvimento da aplicação simples descrito na Seção 4.2, se existem funcionalidades relevantes que auxiliaram na criação dos protótipos.

4.5 Comparação entre as plataformas

Nesta etapa, com base nos critérios definidos na Seção 4.4, será criado um catálogo com os resultados obtidos para realizado a comparação entre as plataformas de desenvolvimento *low-code* escolhidas na Seção 4.3 e o *Íris Low-code Certidão*.

4.6 Análise dos resultados obtidos

Por fim, após realizar a comparação entre as plataformas de desenvolvimento *low-code*, o último passo será analisar os resultados e identificar melhorias que podem ser aplicadas ao *Íris Low-code Certidão*.

5 EXPERIMENTOS E RESULTADOS

Neste capítulo são apresentados os resultados obtidos a partir da metodologia adotada no presente trabalho.

5.1 Levantamento das plataformas

Para que os resultados deste trabalho fossem obtidos, foi necessário, como dito anteriormente na Seção 4.1, fazer o levantamento das plataformas de desenvolvimento *low code* que serão comparadas com o *Iris Low-Code Certidão*.

Com isso, como dito na Seção 2.6, foi escolhido o quadrante mágico para plataformas de aplicativo *low-code* corporativos, criado pelo *Gartner*, como fonte de dados para a escolha das plataformas, devido a este relatório possuir informações relevantes e confiáveis do mercado de plataformas *low-code*. A versão usada do relatório do quadrante mágico neste trabalho é a última até a presente data.

Vincent *et al.* (2022) definiu que as plataformas desta versão do relatório são as seguintes: *Unqork*, *ServiceNow*, *Power Apps*, *Salesforce*, *Appian*, *Creatio*, *Astro Zero (Huawei)*, *Quickbase*, *Kintone*, *Newgen*, *YiDA (Alibaba)*, *Retool*, *Pegasystems*, *Zoho Creator*, *Outsystem*, *Mendix*, *Oracle Apex*. No Quadro 7 pode-se observar informações iniciais sobre as plataformas.

Quadro 7 – Plataformas *low-code* do quadrante mágico

Plataforma	Aplicação	Distribuição	Link
Unqork	web e mobile	<i>cloud</i>	https://www.unqork.com
ServiceNow	web e mobile	<i>on-premise e cloud</i>	https://www.servicenow.com
Power Apps	web e mobile	<i>on-premise e cloud</i>	https://powerapps.microsoft.com
Salesforce	web e mobile	<i>cloud</i>	https://www.salesforce.com
Appian	web e mobile	<i>cloud</i>	https://appian.com
Creatio	web e mobile	<i>cloud</i>	https://www.creatio.com
Astro Zero (Huawei)	web e mobile	<i>cloud</i>	https://www.huaweicloud.com/intl/pt-br/product/appcube.html
Quickbase	web e mobile	<i>on-premise e cloud</i>	https://www.quickbase.com
Kintone	web e mobile	<i>cloud</i>	https://www.kintone.com/en-us
Newgen	web e mobile	<i>cloud</i>	https://newgensoft.com
YiDA (Alibaba)	web e mobile	<i>cloud</i>	https://www.alibabacloud.com/product/yida
Retool	web e mobile	<i>cloud</i>	https://retool.com
Pegasystems	web e mobile	<i>cloud</i>	https://www.pega.com/pt-br
Zoho Creator	web e mobile	<i>cloud</i>	https://www.zoho.com/pt-br/creator
Outsystem	web e mobile	<i>on-premise e cloud</i>	https://www.outsystems.com
Mendix	web e mobile	<i>on-premise e cloud</i>	https://www.mendix.com
OracleApex	web e mobile	<i>cloud</i>	https://apex.oracle.com/pt-br

Fonte: Elaborado pelo autor.

Analisando as informações do Quadro 7, é possível observar que todas as ferramentas possibilitam desenvolver aplicações web e mobile, o que pode indicar uma tendência. Por fim,

investigando a forma de distribuição de cada plataforma, foi percebido que este mercado utiliza duas estratégias, *on-premise* e *cloud*. A estratégia *on-premise* é quando as plataformas podem ser executadas no ambiente privado da empresa. Na estratégia em *cloud*, as plataformas são executadas na infraestrutura da empresa que está fornecendo o serviço.

5.1.1 Seleção das plataformas de desenvolvimento low code

A partir do levantamento feito na seção anterior, avaliamos quais plataformas de desenvolvimento *low code* seriam usadas no estudo comparativo, tendo em vista que algumas plataformas poderiam ter algo que impossibilitasse a realização deste estudo.

Em um primeiro momento foi verificado que algumas plataformas não disponibilizavam a versão gratuita, o que impossibilitava de conduzir um estudo mais aprofundado na plataforma e com isso elas não foram incluídas nesta pesquisa. Em outro momento foi percebido que algumas plataformas se concentram em áreas específicas do mercado ou tinham algum impedimento que impossibilitava de criar o protótipo que cumprisse os requisitos funcionais mínimos definidos na Seção 4.2. No Quadro 8 é possível observar a relação das plataformas que possui plano gratuito ou não, e se havia algum impedimento.

Nas plataformas que possuem acesso gratuito, mas não foi liberado o acesso, a liberação do ambiente dependia da aprovação da empresa. Com o *Power Apps*, ao tentar realizar o cadastro, o site da plataforma retornava um erro exigindo um e-mail comercial, ao tentar usar o e-mail de estudante o erro foi que o serviço não estava disponível para a conta corporativa ou de estudante associada.

Com base nas informações apresentadas na Tabela 8, foi escolhido as seguintes plataformas: *Zoho Creator*, *Outsystem* e *Mendix*.

5.1.2 Validação das plataformas escolhidas

Para validar as plataformas escolhidas, em cada uma foi criado um protótipo seguindo os requisitos funcionais mínimos definidos na Seção 4.2 de modo a simular o fluxo de geração de certificado do *Íris Low-code Certidão*.

Ao analisar as funcionalidades disponíveis na plataforma do *Zoho Creator*, *Outsystem* e *Mendix* foi percebido que o processo para criação da aplicação seria semelhante na plataforma da *Outsystem* e na *Mendix*, já na *Zoho Creator* teríamos algumas facilidades.

Com isso decidimos que a criação do protótipo seria realizado apenas na plataforma

Quadro 8 – Seleção das plataformas do quadrante mágico

Plataforma	Possui plano gratuito	Observação
Zoho Creator	Sim	Foi possível cumprir todos os requisitos funcionais
Mendix	Sim	Foi possível cumprir todos os requisitos funcionais
Outsystem	Sim	Foi possível cumprir todos os requisitos funcionais
Creatio	Sim	Não foi possível cumprir todos os requisitos funcionais
Quickbase	Sim	Não foi possível cumprir todos os requisitos funcionais
Kintone	Sim	Não foi possível cumprir todos os requisitos funcionais
Retool	Sim	Não foi possível cumprir todos os requisitos funcionais
Salesforce	Sim	Não foi possível cumprir todos os requisitos funcionais
OracleApex	Sim	Não foi liberado o acesso a plataforma
Pegasystems	Sim	Não foi liberado o acesso a plataforma
Power Apps	Sim	É exigido um e-mail comercial para criar a conta
Unqork	Não	Eliminada por não ter plano gratuito
Appian	Não	Eliminada por não ter plano gratuito
Astro Zero (Huawei)	Não	Eliminada por não ter plano gratuito
Newgen	Não	Eliminada por não ter plano gratuito
YiDA (Alibaba)	Não	Eliminada por não ter plano gratuito
ServiceNow	Não	Eliminada por não ter plano gratuito

Fonte: Elaborado pelo autor.

da *Zoho Creator* e na *Outsystem*, e quanto na *Mendix* seria feito apenas o levantamento das funcionalidades da plataforma para comparar com o *Íris Low-code Certidão*.

Durante os estudos nas plataformas foi percebido que a *Zoho Creator* possuía algumas funcionalidades que facilitavam a criação de certificado. Já na *Outsystem* e na *Mendix* o processo de criar modelos era mais trabalhoso e exigia mais conhecimento técnico.

Embora não fosse possível criar um fluxo semelhante nas plataformas no *Zoho Creator* e na *Outsystem*, o objetivo final de criar um protótipo cumprindo os requisitos funcionais definidos na Seção 4.2 foi alcançado.

5.1.3 Dificuldades encontradas

Para a aplicação ser construída nas plataformas foi necessário estudar cada uma delas para entender suas limitações. Todas as plataformas *low code* usadas neste experimento estavam na sua versão gratuita.

Durante os teste no *Zoho Creator* foi percebido que a funcionalidade que permitia liberar a função de gerar PDF em links públicos não estava disponível, enquanto na *Outsystem* não tivemos problema relacionado a geração de PDF. Uma forma de contornar essa situação foi enviar o PDF diretamente por e-mail do usuário que solicitou a certidão.

Outra dificuldade está relacionada a criação o modelo de certificado na *Outsystem*, que foi mais complexo em comparação ao *Zoho Creator*. Enquanto no *Zoho Creator* contamos

com uma tela própria e simplificada para a criação de modelos, semelhante ao que tem no *Íris Low-code Certidão*. No *Outsystem*, criar um modelo de certificado é semelhante a criar uma tela na aplicação, tornando o processo mais complexo, por ser necessário aplicar estilização na página com CSS para alcançar o resultado desejado.

Outra limitação foi a falta de visualização da estrutura do banco de dados no *Zoho Creator*. com isso, tivemos que lidar com o relacionamento das tabelas por meio de campos específicos nos formulários, o que dificultou a visualização dos relacionamentos das tabelas no banco de dados.

Essas dificuldades mostram a importância de conhecer bem as limitações das ferramentas que estamos usando e o impacto que essas limitações podem gerar na aplicação no futuro. Apesar dos desafios, foram encontradas soluções alternativas para superar as dificuldades e alcançar os objetivos da pesquisa.

5.2 Definição dos atributos de comparação

Nesta seção serão apresentados os resultados dos estudos realizados nas plataformas escolhidas na Seção 5.1.1 com base no que foi definido na metodologia na Seção 4.4. É importante ressaltar que todo estudo foi realizado na versão gratuita das plataformas. Em caso de dúvida sobre alguma funcionalidade, a documentação das plataformas e vídeos no *YouTube* foram usados como fonte de consulta, tendo em vista que o suporte só está disponível para os planos pagos das plataformas.

5.2.1 Critérios quantitativos

5.2.1.1 Tempo para criação do template do certificado

Para realizar o experimento e conseguir os dados necessários para avaliar esse atributo, primeiro foi definido um *template* padrão que seria seguido em todas as plataformas. Como o *Íris Low-code Certidão* possui algumas limitações, o *template* padrão foi definido a partir de um exemplo do *Íris Low-code Certidão*. Na Figura 10 pode-se observar o modelo de certificado replicado na plataforma do *Zoho Creator* e do *Outsystem*.

Para medir o tempo de criação do *template* nas plataformas foi necessário gravar o processo de criação e com isso medir o tempo gasto em cada plataforma. Para o experimento realizado na plataforma da *Outsystem* e do *Íris Low-code*, visto que foram realizados em ambiente

Figura 10 – modelo de certidão replicado nas plataformas



Fonte: Elaborado pelo autor.

local, foi utilizado um computador *desktop* com sistema operacional *MacOS Ventura 13.2.1*, com 16 GB de RAM e um processador Intel i5 de 12ª geração. O experimento realizado com o *Zoho Creator* foi no ambiente da empresa na nuvem.

Na Tabela 4 pode-se observar o tempo necessário para criar o *template* em cada plataforma. É possível perceber que o tempo de criação do *template* no *Zoho Creator* e no *Íris Low-code Certidão* são bem próximos devido aos dois possuírem recursos que facilitam a criação desse tipo de *template*, enquanto no *Outsystem* o tempo chega a ser mais do que 3 vezes o tempo que alcançado no *Íris Low-code Certidão*. Na Seção 5.4 é possível verificar o link da *playlist* com os experimentos.

A complexidade que existe para executar essa tarefa na *Outsystem* é devido à ferramenta não possuir nenhuma funcionalidade que crie, de forma simples, esses *templates*. Os mesmos componentes que usamos para construir telas são usados para criar os *templates*, por um lado isso possibilita construir *templates* mais robustos, contudo, é necessário mais conhecimento técnico para alcançar esse objetivo.

É importante ressaltar que os dados coletados sobre o tempo de criação do *template* depende da experiência e do conhecimento do desenvolvedor sobre a plataforma, que neste cenário é o escritor deste trabalho.

Tabela 4 – Tempo de criação do *template* nas 3 plataformas

Plataformas	Tempo gasto (em minutos)
Outsystem	7:16
Zoho Creator	2:44
Íris Low Code Certidão	2:26

Fonte: Elaborado pelo autor.

5.2.2 Critérios qualitativos

5.2.2.1 Recursos de banco de dados

Para este atributo, foi necessário verificar nas plataformas que estão sendo comparadas se elas possuíam os recurso de banco de dados definidos na Seção 4.4.2.1. No Quadro 9 pode-se observar os resultados das pesquisas nas plataformas.

Quadro 9 – Recursos de banco de dados

Plataformas low-code	Possui banco de dados interno	Permite importar o banco de dados	Permite exportar o banco de dados	Permite acesso ao banco de dados	Permite conexão com banco de dados externo
Outsystem	✓	x	x	x	✓
Mendix	✓	x	x	x	✓
Zoho Creator	✓	x	x	x	✓
Low Code Certidão	✓	✓	✓	✓	✓

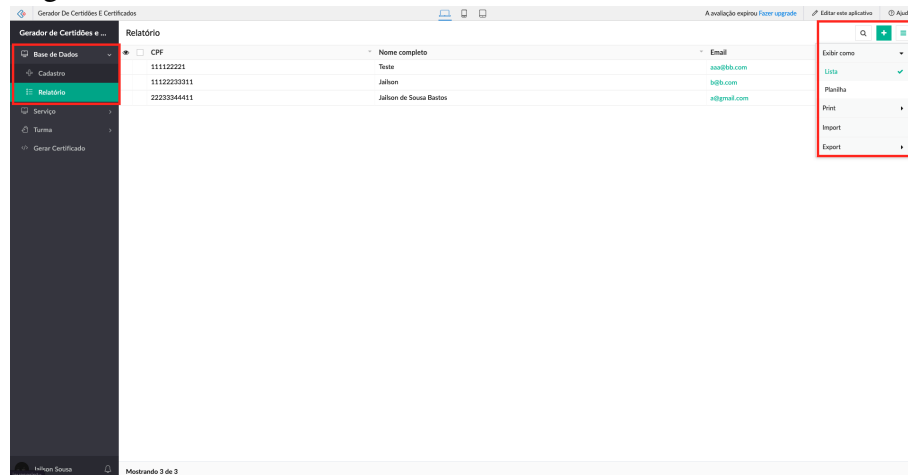
Fonte: Elaborado pelo autor.

Investigando as plataformas foi percebido que todas possuem banco de dados interno. Na *Outsystem* o banco usado é o *SQL Server*, no *Íris Low-code Certidão* existe a possibilidade de usar o PostgreSQL ou o MongoDB, no *Zoho Creator* e no *Mendix* não conseguimos encontrar nenhuma informação sobre qual banco de dados a empresa adota internamente.

Embora todas as plataformas possuam banco de dados interno, apenas no *Íris Low-code Certidão* o acesso está disponível. No *Outsystem* e no *Mendix* conseguimos interagir com os dados para editar, consultar e deletar com algumas telas que a plataforma disponibiliza e nas integrações que realizamos com as telas do sistema que está em desenvolvimento. No *Zoho Creator* a interação com os dados acontece apenas nas telas de formulário ou nas telas de relatório, geradas automaticamente com a tela de formulário. Na Figura 11 pode-se visualizar uma representação da tela de relatórios na plataforma do *Zoho Creator*.

No estudo realizado nas plataformas, foi identificado que apenas no *Íris Low-code Certidão* temos a possibilidade de importar ou exportar o banco de dados. Embora o acesso ao banco de dados não fosse permitido diretamente, investigamos a possibilidade de ter alguma funcionalidade que disponibilizasse esses recursos, mas nada foi encontrado. O *Zoho Creator*

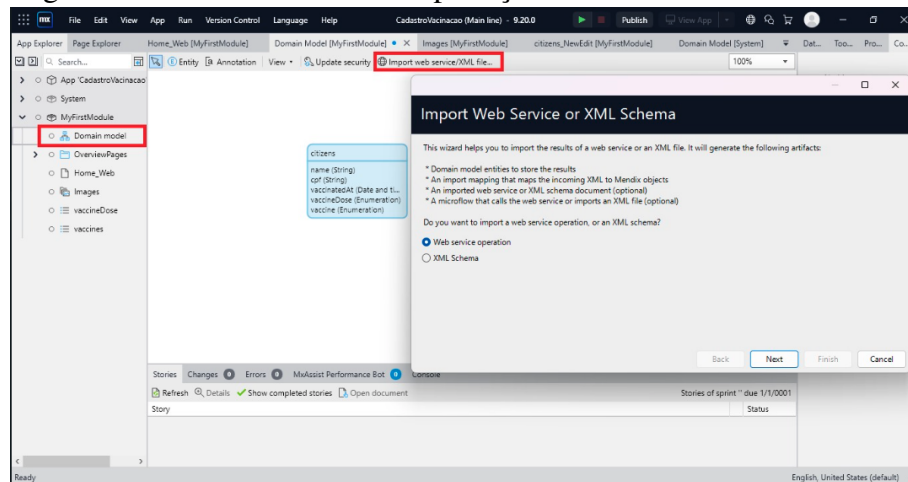
Figura 11 – Zoho Creator - Tela de relatório de dados



Fonte: Elaborado pelo autor.

permite que os dados sejam importados ou exportados individualmente de cada tabela vinculada a um formulário, através das telas de relatório que pode ser visto na Figura 11. No *Outsystem* e no *Mendix* conseguimos importar dados por uma tela da plataforma, mas não foi encontrado nada referente sobre exportar o banco de dados ou apenas os dados. Na Figura 12 pode-se visualizar uma representação da tela de importação dos dados na plataforma do *Mendix*.

Figura 12 – Mendix - Tela de importação de dados



Fonte: Elaborado pelo autor.

Por fim, foi percebido que todas as plataformas permitem que seja usado um banco de dados externo. Com isso, todos os recursos que antes não estavam disponíveis no banco interno das plataformas passam a ser acessíveis.

5.2.2.2 Integração com API

Neste atributo, verificamos nas plataformas se elas possuíam os recurso de integração com *API* que definidos na Seção 4.4.2.2. No Quadro 10 pode-se observar os resultados das pesquisas nas plataformas.

Quadro 10 – Integração com *API*

Plataformas low-code	Permite consumir os dados por API	Permite integrar funcionalidade com uma API	Permite consumir dados da Planilhas Google
Outsystem	✓	✓	x
Mendix	✓	✓	x
Zoho Creator	✓	✓	x
Low Code Certidão	✓	x	✓

Fonte: Elaborado pelo autor.

Na funcionalidade permite consumir os dados por *API*, todas as plataformas possuem esse recurso, contudo no caso do *Zoho Creator* e do *Mendix*, é necessário manualmente acionar essa função para atualizar os dados. No caso do *Íris Low-code Certidão* e do *Outsystem*, basta configurar esse recurso que os dados serão consumidos da *API* sempre que necessário.

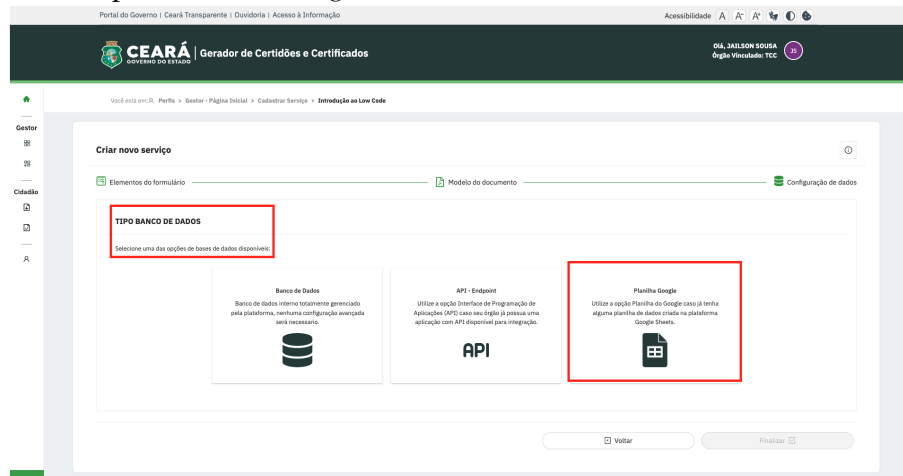
Na funcionalidade permite integrar funcionalidade com uma *API*, apenas o *Íris Low-code Certidão* não disponibiliza esse recurso. No caso das outras plataformas que foram avaliadas, normalmente esse recurso é usado durante a criação de *workflow* que é posteriormente atrelado a algum evento de botão ou algum *input*.

Na funcionalidade permitir consumir dados da *Planilhas Google*, apenas o *Íris Low-code Certidão* possui essa integração. No *Zoho Creator* identificado que conseguimos importar dados de planilha que estão no *Google Drive* de uma vez, enquanto no *Íris Low-code Certidão*, a cada novo dado inserido em uma planilha associada com um serviço já é reconhecido no sistema. Na Figura 13 pode-se visualizar uma representação da configuração de dados de um serviço na plataforma do *Íris Low-code Certidão* com as planilhas do *Google*.

5.2.2.3 Outras funcionalidades

Para realizar a análise neste atributo, como foi estabelecido na Seção 4.4.2.3, documentamos as funcionalidades presentes nas plataformas *Outsystem* e no *Zoho Creator* que foram

Figura 13 – *Iris Low-Code Certidão* - Opção de configurar os dados com as planilhas do *Google*



Fonte: Elaborado pelo autor.

usadas no processo de criar a aplicação definida na Seção 4.2. Com a *Mendix*, embora não tenha sido feito o desenvolvimento da aplicação, investigamos se a plataforma tinha funcionalidades semelhantes às encontradas nas outras plataformas. No Quadro 11 pode-se observar a relação das plataformas com este atributo.

Quadro 11 – Outras funcionalidades

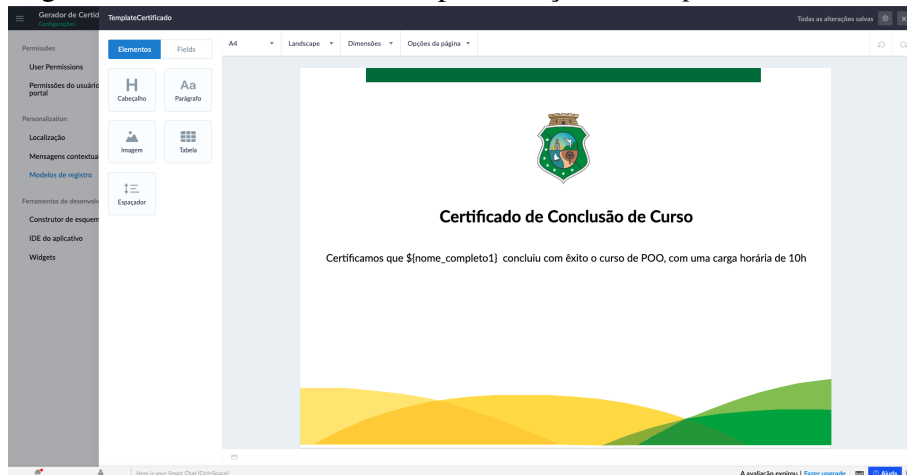
Plataformas low-code	Permite a criação de template de PDF personalizado	Permite a criação de workflow	Permite a criação de cron job	Permite que as telas dos formulário sejam geradas dinamicamente	Permite incorporar telas do sistema em outros sistemas	Permite o envio de email	Permite a implantação de ambientes de desenvolvimento dentro do sistema
Outsystem	✓	✓	✓	X	✓	✓	✓
Mendix	✓	✓	✓	X	X	✓	✓
Zoho Creator	✓	✓	✓	X	✓	✓	✓
Low Code Certidão	✓	X	X	✓	X	X	X

Fonte: Elaborado pelo autor.

Na funcionalidade, permitir a criação de template de PDF personalizado, todas as plataformas conseguem realizar essa tarefa, contudo, no caso da *Outsystem* e do *Mendix*, elas não possuem nenhuma funcionalidade que facilite a criação de *templates*, tornando o processo complexo. Na Seção 5.4 é possível verificar a *playlist* com o video do processo de *template* com na *Outsystem*. Na Figura 14 pode-se visualizar a tela de criação de *template* do da plataforma *Zoho Creator*, enquanto na Figura 15 pode-se visualizar a tela de criação de *template* da *Outsystem*.

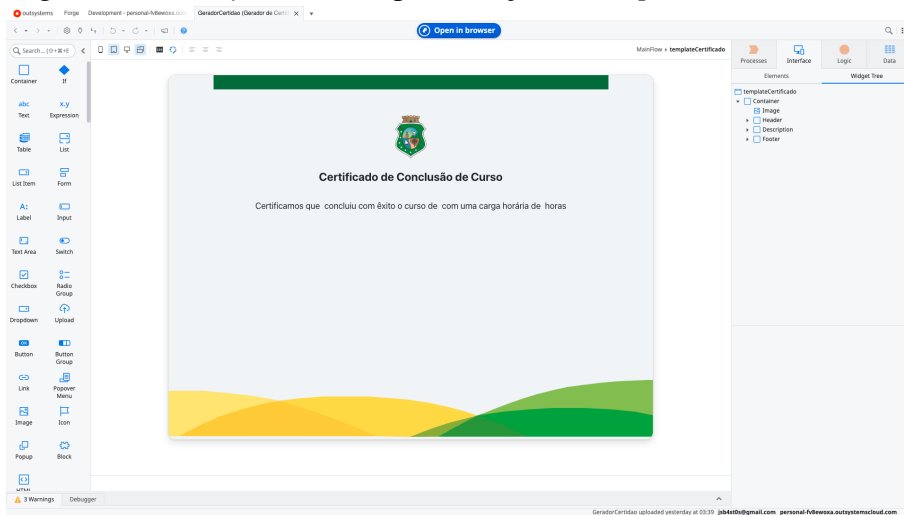
Nas funcionalidades, permitir a criação de *workflow* e *cron job*, apenas o *Iris Low-code Certidão* não tem suporte a essa funcionalidade. Todas as plataformas analisadas neste trabalho que possuem algum recurso visual que facilita a criação de *workflow*, embora seja necessário ter conhecimento técnico em programação. No Caso do *cron job*, ele é um recurso

Figura 14 – *Zoho Creator* - Tela para criação de *template*



Fonte: Elaborado pelo autor.

Figura 15 – *Outsystem* - Tela para criação de *template*



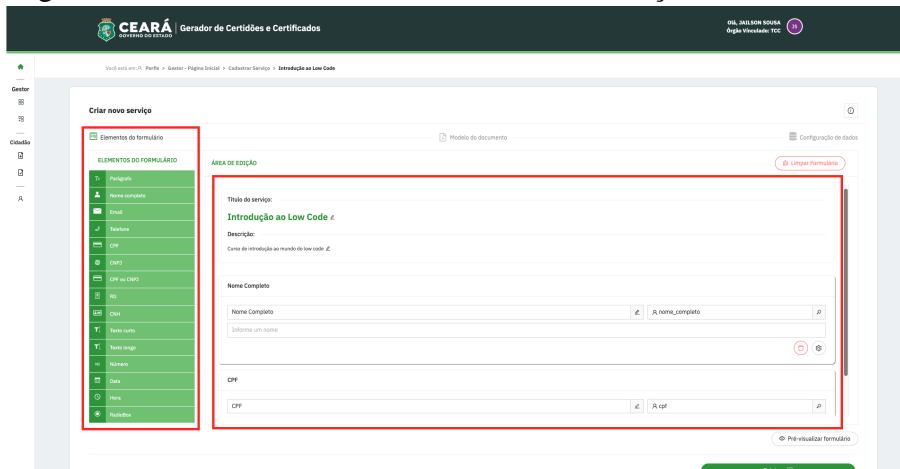
Fonte: Elaborado pelo autor.

que pode ser usado em conjunto com o *workflow* para programar que determinadas ações sejam realizadas automaticamente sem a necessidade que algum usuário precise executar a ação manualmente.

Na funcionalidade, permitir que as telas dos formulários sejam geradas dinamicamente, apenas o *Íris Low-code Certidão* tem suporte a essa funcionalidade. Na *Outsystem*, por não ter suporte isso, a estratégia que pode ser usada, por exemplo, na tela de emissão de certificado, seria criar uma tela genérica com um campo para selecionar o serviço que deseja emitir o certificado. No *Zoho Creator* a única estratégica identificada seria criar uma tela de emissão de certificado para cada serviço, porque nessa plataforma não existe uma forma de escolher dinamicamente um *template* de PDF na chamada do *workflow* para emitir o certificado. Na Figura 16 pode-se visualizar a tela de criação de formulário no *Íris Low-code Certidão*,

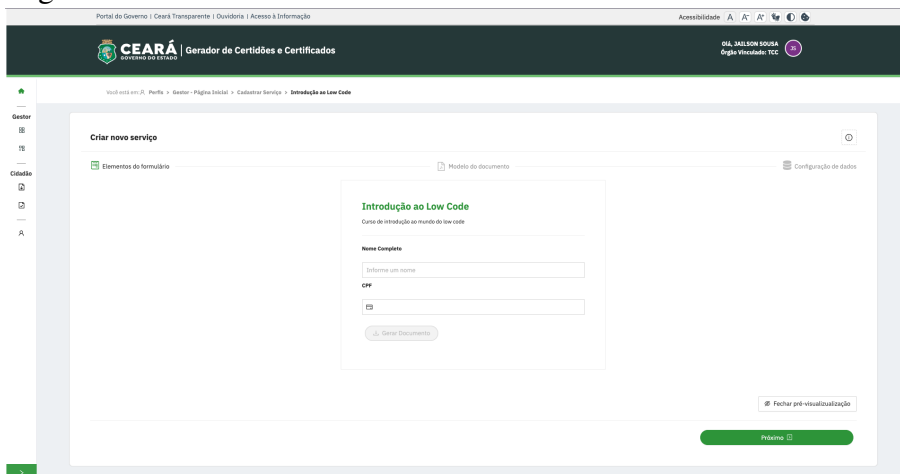
enquanto na Figura 17 pode-se visualizar a prévia do formulário que está sendo criado.

Figura 16 – *Iris Low-Code Certidão* - Tela de criação de formulário



Fonte: Elaborado pelo autor.

Figura 17 – *Iris Low-Code Certidão* - Prévia da tela do formulário



Fonte: Elaborado pelo autor.

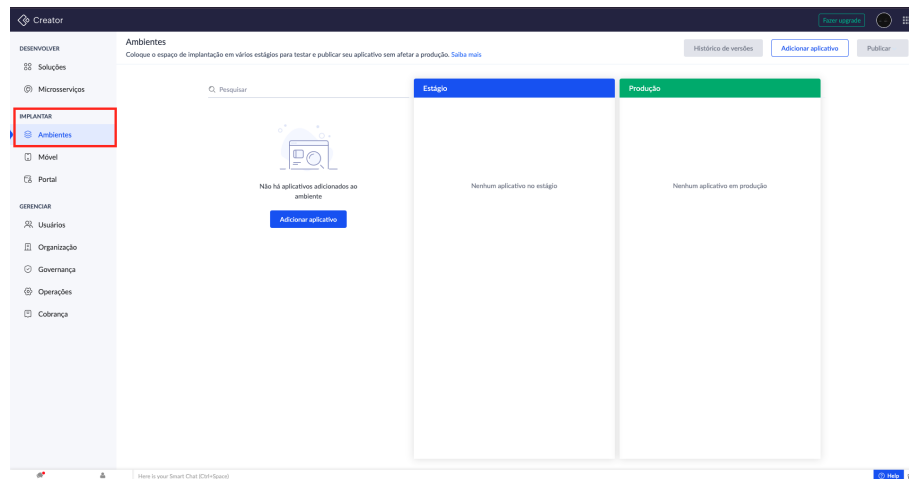
Na funcionalidade, permitir incorporar telas do sistema em outros sistemas, esse recurso foi visto apenas na plataforma do *Zoho Creator* e do *Outsystem*. O que torna esse recurso interessante é o fato de conseguir gerenciar quais recursos estarão disponíveis na tela pública. No *Zoho Creator*, por exemplo, pode-se liberar que em uma tela de relatório pública seja possível visualizar os dados e exportar os dados apenas no formato CSV.

O recurso de permitir o envio de e-mail é útil por ser mais uma alternativa para o usuário receber o seu documento. Nesta funcionalidade, apenas o *Íris Low-code Certidão* não possui este recurso.

E por fim, a funcionalidade, permitir a implantação de ambientes de desenvolvimento no sistema, está disponível em todas as plataformas, exceto no *Íris Low-code Certidão*. Esse

recurso é muito útil porque possibilita que ambientes de desenvolvimento diferentes, por exemplo, homologação e produção, sejam configurados no próprio sistema. Embora isso seja possível ser feito configurando a infraestrutura do projeto, ter isso ao nível de sistema seria mais prático para quem usa o sistema gerenciar essa rotina. Na Figura 18 pode-se visualizar a tela de implantação de ambientes da plataforma *Zoho Creator*

Figura 18 – *Zoho Creator* - Tela para implantação de ambientes de desenvolvimento



Fonte: Elaborado pelo autor.

5.3 Catálogo com os resultados obtidos

Nesta seção está disponibilizado o link com todos os resultados obtidos neste trabalho reunidos em apenas um local em formato de tabela. A tabela disponibilizada no link reúne todos os dados anteriores disponibilizados neste trabalho. O link consistem em: https://docs.google.com/spreadsheets/d/18rhWWx_MKkN703hQ_ua1y5vq114ouayvhS3C6AjMPIM/edit?usp=sharing

5.4 Vídeos dos experimentos

Nesta seção está disponibilizado o link da playlist do *YouTube* com as demonstrações realizadas na plataforma do *Zoho Creator*, *Outsystem* e *Íris Low-code Certidão*. O link consistem em: <https://www.youtube.com/playlist?list=PLymssG-ZL5YtyaFwNHera875ijIIFkLV>

6 CONCLUSÕES E TRABALHOS FUTUROS

Com base nos resultados obtidos a partir do levantamento das plataformas apresentado na Subseção 5.1, pode-se perceber que as grandes plataformas de *low-code* do mercado, segundo o Vincent *et al.* (2022), permitem desenvolver aplicações *web* e *mobile*. Com isso, seria importante o *Íris Low-code Certidão* começar refletir sobre a construção de uma plataforma *mobile*.

Mesmo que o *Íris Low-code Certidão* tenha conseguido o menor tempo para criação do *template* de certificado como pode ser visto na Subseção 5.2.1.1, pode-se inferir que isso aconteceu devido a ele ser uma ferramenta feita especificamente para isso, o que torna mais rápido a criação de *template* quando existe uma estrutura pronta. Contudo, tornar a criação de *template* mais flexível no *Íris Low-code Certidão*, assim como é feito no *Zoho Creator*, seria mais adequado por trazer mais possibilidades de personalização ao mesmo tempo que mantém o processo de forma simples.

A partir dos resultados apresentados na Subseção 5.2.2, Embora o *Íris Low-code Certidão* tenha apresentado bons resultados no Quadro 9, pode-se inferir que os resultados apresentados no Quadro 10 e no Quadro 11 mostram que existem alguns pontos de melhorias que poderiam ser adicionados na plataforma do *Íris Low-code Certidão* no futuro. Permitir integrar novas funcionalidades através da integração com *API* e a criação de *workflow* são recursos que permitem mais flexibilidade para os usuários da plataforma e com isso os desenvolvedores da plataforma podem focar em outras demandas mais importantes.

Para realização de trabalhos futuros é válido avaliar quais das funcionalidades identificadas na Seção 5.2 poderiam ser implementadas no *Íris Low-code Certidão* com o intuito trazer melhorias para a plataforma com base no que as grandes plataformas de desenvolvimento *low-code* apresentadas por Vincent *et al.* (2022) no último relatório do quadrante mágico para plataformas de aplicativos *low-code* corporativos até a data deste trabalho. Além disso, seria válido realizar um novo levantamento de plataformas que sejam mais alinhadas com a proposta do *Íris Low-code Certidão* e com isso identificar novas funcionalidades que poderiam agregar a plataforma.

REFERÊNCIAS

- ALEXANDER, F. **What Can You Build With Low-Code?** 2021. Disponível em: <https://www.outsystems.com/blog/posts/what-can-you-build-with-low-code/>. Acesso em: 20 ago. 2022.
- ALEXANDER, F. **What Is Low-Code?** 2021. Disponível em: <https://www.outsystems.com/blog/posts/what-is-low-code/>. Acesso em: 20 ago. 2022.
- BOCK, A. C.; FRANK, U. Low-code platform. **Business Information Systems Engineering**, Springer Fachmedien Wiesbaden, Wiesbaden, v. 63, n. 6, p. 733–740, 2021. ISSN 2363-7005.
- BRASSCOM. **Demanda de Talentos em TIC e Estratégia e TCEM**. 2021. Disponível em: <https://brasscom.org.br/pdfs/demanda-de-talentos-em-tic-e-estrategia-tcem/>. Acesso em: 10 set. 2022.
- CRUZ, M. A. A. da; PAULA, H. T. L. de; CAPUTO, B. P. G.; MAFRA, S. B.; LORENZ, P.; RODRIGUES, J. J. P. C. Olp—a restful open low-code platform. **Future Internet**, v. 13, n. 10, 2021. ISSN 1999-5903. Disponível em: <https://www.mdpi.com/1999-5903/13/10/249>.
- DI RUSCIO, D.; KOLOVOS, D.; LARA, J. de; PIERANTONIO, A.; TISI, M.; WIMMER, M. Low-code development and model-driven engineering: Two sides of the same coin? **Software and systems modeling**, Springer Berlin Heidelberg, Berlin/Heidelberg, v. 21, n. 2, p. 437–446, 2022. ISSN 1619-1366.
- ELMASRI, R.; NAVATHE, S. B.; PINHEIRO, M. G. *et al.* **Sistemas de banco de dados**. [S. l.]: Pearson Addison Wesley São Paulo, 2005.
- GARTNER. **Gartner Forecasts Worldwide Low-Code Development Technologies Market**. 2022. Disponível em: <https://www.gartner.com/en/newsroom/press-releases/2022-12-13-gartner-forecasts-worldwide-low-code-development-technologies>. Acesso em: 20 jun. 2023.
- GARTNER. **Magic Quadrant research methodology**. 2023. Disponível em: <https://www.gartner.com/en/research/methodologies/magic-quadrants-research>. Acesso em: 8 jul. 2023.
- LUO, Y.; LIANG, P.; WANG, C.; SHAHIN, M.; ZHAN, J. Characteristics and challenges of low-code development. In: **Proceedings of the 15th ACM / IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)**. ACM, 2021. Disponível em: <https://doi.org/10.1145%2F3475716.3475782>.
- MANNING, M. **Zurich FaceQuote Goes from Idea to App in Weeks**. 2019. Disponível em: <https://www.mendix.com/blog/zurich-facequote-goes-idea-app-weeks/>. Acesso em: 15 out. 2022.
- MENDIX. **Dubai Drives Digitization**. 2021. Disponível em: <https://www.mendix.com/customer-stories/dubai-municipality/>. Acesso em: 15 out. 2022.
- MORAES, P. H. M. M. **Aplicação de ferramentas low-code para melhoria e automação de processos em uma empresa de contabilidade**. 2022. Disponível em: <http://www.repositorio.ufc.br/handle/riufc/65241>.

O'LEARY, D. E. Gartner's hype cycle and information system research issues. **International journal of accounting information systems**, Elsevier Inc, New York, v. 9, n. 4, p. 240–252, 2008. ISSN 1467-0895.

OUTSYSTEMS. **Top Application Development Trends in 2019**. 2019. Disponível em: <https://www.outsystems.com/1/state-app-development-trends/>. Acesso em: 15 out. 2022.

SAHAY, A.; INDAMUTSA, A.; RUSCIO, D. D.; PIERANTONIO, A. Supporting the understanding and comparison of low-code development platforms. In: **2020 46th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)**. [S. l.: s. n.], 2020. p. 171–178.

SAHAY, A.; RUSCIO, D. D.; IOVINO, L.; PIERANTONIO, A. Analyzing business process management capabilities of low-code development platforms. **Software, practice experience**, Wiley Subscription Services, Inc, Bognor Regis, v. 53, n. 4, p. 1036–1060, 2023. ISSN 0038-0644.

SHRIDHAR, S. Analysis of low code-no code development platforms in comparison with traditional development methodologies. **International Journal for Research in Applied Science and Engineering Technology**, v. 9, n. 12, p. 508–513, 2021. ISSN 2321-9653.

STYLOS, J.; MYERS, B. Mapping the space of api design decisions. In: **IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC 2007)**. [S. l.: s. n.], 2007. p. 50–60.

TV, L. C.; ABDULLA, M. S. **A survey of Low-Code/No-Code software development tools with an application**. 2022. Disponível em: https://iimk.ac.in/uploads/publications/IIMK_WPS_524_ITS_2022_08_Upload_File.pdf.

VINCENT, P.; IJIMA, K.; DRIVER, M.; WONG, J.; NATIS, Y. **Magic quadrant for enterprise low-code application platforms**. 2019. Acesso em: 10 nov. 2022.

VINCENT, P.; IJIMA, K.; LEOW, A.; WEST, M.; MATVITSKY, O. **Magic quadrant for enterprise low-code application platforms**. 2022. Disponível em: <https://www.gartner.com/doc/reprints?id=1-2C84AIFS&ct=230110&st=sb>. Acesso em: 20 jun. 2023.