



**Politécnico
Castelo Branco**

Escola Superior
de Tecnologia

Low-Code – A nova realidade da programação

Projeto I

Eduardo Jonas Tomás Alves, 20220429

Nuno Alexandre Salvado Baptista, 20220423

Orientador

Professor Doutor Pedro Nuno Moreira da Silva

Trabalho de Projeto apresentado à Escola Superior de Tecnologia do Instituto Politécnico de Castelo Branco realizada sob a orientação científica do Professor Doutor Pedro Nuno Moreira da Silva, do Instituto Politécnico de Castelo Branco.

Janeiro 2025

Composição do júri

Presidente do júri

Professora Doutora, Arminda da Conceição dos Santos Guerra e Lopes

Professor Coordenador, Escola Superior de Tecnologia

Vogais

Professor Doutor, Eduardo Sabina dos Santos Valente

Professor Adjunto, Escola Superior de Tecnologia

Professor Doutor, Pedro Nuno Moreira da Silva

Professor Adjunto, Escola Superior de Tecnologia

Agradecimentos

Gostaríamos de expressar a nossa mais sincera gratidão a todas as pessoas e empresas que tornaram possível a realização deste projeto.

Agradecemos ao nosso orientador e professores, cujo conhecimento, paciência e orientação foram essenciais ao longo de todas as etapas deste trabalho.

À Escola Superior de Tecnologia, pelos recursos disponibilizados que proporcionaram um ambiente de estudo e investigação adequado.

Aos nossos colegas de curso que nos ajudaram com opiniões, pela colaboração e partilha de conhecimentos, que enriqueceram as discussões e ideias que surgiram ao longo deste projeto.

Um agradecimento especial aos participantes e colaboradores do jornal *Canal Albicastrense*, cuja disponibilidade e informações foram fundamentais para o desenvolvimento e validação do protótipo.

A todos, o nosso sincero obrigado.

Resumo

O projeto foca no planeamento e implementação de uma aplicação de notícias digitais para a região da Beira Baixa, utilizando tecnologia Low-Code. Com a crescente adoção dessas plataformas como solução ágil e acessível para o desenvolvimento de *software*, o objetivo é analisar o seu impacto, identificar tendências e desafios, desenvolver um protótipo e testar a sua utilidade num contexto real.

O planeamento seguiu etapas estruturadas, incluindo revisão bibliográfica, análise de estudos e ferramentas Low-Code, desenvolvimento do protótipo e redação do relatório. A diferenciação entre Low-Code e No-Code é destacada, assim como as vantagens, desafios e aplicações empresariais dessas tecnologias, com análise de plataformas populares como *OutSystems*, *Mendix* e *Microsoft PowerApps*. Para garantir fundamentação teórica, foi realizada uma revisão sistemática da literatura com base na metodologia *PRISMA*, permitindo a seleção e análise de estudos científicos relevantes.

O desenvolvimento do protótipo tem como foco a criação de uma aplicação para o jornal digital *Canal Albicastrense*, permitindo que jornalistas publiquem e gerenciem notícias de forma simplificada, enquanto os leitores podem aceder a conteúdos, filtrar informações e receber notificações personalizadas. São detalhados os requisitos funcionais e técnicos da aplicação, além do *design* da *interface*, utilizando *wireframes* e um protótipo interativo desenvolvido no *Figma*.

O projeto enfatiza a importância do Low-Code na democratização do desenvolvimento de *software*, promovendo inovação, acessibilidade e maior eficiência no processo de criação de aplicações. Procura-se não apenas validar as capacidades destas plataformas, mas também evidenciar o seu impacto na transformação digital e no desenvolvimento tecnológico de forma ágil e colaborativa.

Palavras chave

Low-Code, Impacto do Low-Code, Low-Code contra No-Code, Plataformas Low-Code

Abstract

In the current context of digital transformation, the demand for agile and accessible solutions has driven the widespread adoption of Low-Code platforms. This project explores the planning and implementation of a digital news application for the Beira Baixa region, using Low-Code technology to streamline development and enhance accessibility. The study aims to analyze the impact of Low-Code, identify trends and challenges, develop a prototype, and assess its practical effectiveness.

The project follows a structured methodology, including a systematic literature review based on the PRISMA framework, an analysis of related works, and the evaluation of popular Low-Code platforms such as OutSystems, Mendix, and Microsoft PowerApps. The prototype was designed for the Canal Albicastrense digital newspaper, allowing journalists to create and manage news articles efficiently while enabling readers to access, filter, and receive personalized notifications. The system's functional and technical requirements were defined, and the user interface was developed using wireframes and an interactive prototype in Figma.

By demonstrating the capabilities of Low-Code platforms, this project highlights their role in democratizing software development, fostering innovation, and improving efficiency. The findings suggest that Low-Code technologies significantly accelerate the software development process, reduce costs, and bridge the gap between technical and non-technical professionals, making digital solutions more accessible and collaborative.

Keywords

Low-Code, Impact of Low-Code, Low-Code vs. No-Code, Low-Code Plataforms

Índice geral

1. Introdução	1
1.1. Enquadramento e motivação do Projeto	2
1.1.1. Enquadramento	2
1.1.2. Motivação	2
1.2. Objetivos	3
1.3. Planeamento do Projeto	3
2. Estado da Arte e Análise de Trabalhos Relacionados	5
2.1. Estado da Arte	5
2.1.1. Tecnologias Low-Code	6
2.1.2. Tecnologias No-Code	9
2.1.3. Diferença entre Low-Code e No-Code	10
2.1.4. Impacto do Low-Code no desenvolvimento de Software	13
3. Descrição detalhada da metodologia adotada	15
3.1. Análise de Trabalhos Relacionados	15
3.1.1. Estudo e análise de trabalhos relacionados	15
3.1.2. Questões de pesquisa	16
3.1.3. Critérios de inclusão	16
3.2. Estratégia de pesquisa	16
3.3. Resultados	18
3.4. Extração e análise dos dados	19
3.5. Discussão	22
3.6. Protótipo	33
3.6.1. Diagrama de Contexto	37
3.6.2. Diagrama de Serviços	41
3.6.3. Cenários de Contexto	47
3.6.4. Biblioteca de Estilos <i>UI</i>	78
4. Conclusões e Trabalho Futuro	89
5. Referências	90

Índice de figuras

Figura 1 - Esquema com os logótipos das principais plataformas Low-Code	7
Figura 2 - Pesquisa na base de dados IEEE Xplore	17
Figura 3 - Pesquisa na base de dados ACM Digital	17
Figura 4 - Pesquisa na base de dados Scopus	17
Figura 5 - Fluxograma	18
Figura 6 - Nome da empresa	22
Figura 7 - Há quanto tempo a sua empresa utiliza tecnologias Low-Code?	22
Figura 8 - Que plataforma(as) Low-Code são utilizadas na sua empresa?	23
Figura 9 - Qual(ais) as principais finalidades na utilização?	23
Figura 10 - A sua empresa oferece formação ou workshops para os colaboradores sobre tecnologias Low-Code?	24
Figura 11 - Em que medida a plataforma Low-Code responde às necessidades específicas da empresa?	24
Figura 12 - Quais funcionalidades de personalização sente falta?	24
Figura 13 - Como avalia a facilidade de integração da plataforma com outros sistemas?	25
Figura 14 - Enfrentou limitações técnicas ao integrar com outros sistemas? ..	25
Figura 15 - Quais foram as ferramentas ou sistemas que conseguiu integrar com a plataforma?	25
Figura 16 - Qual o nível de dependência do suporte do fornecedor?	26
Figura 17 - A documentação fornecida pelo fornecedor é suficiente para uso independente?	26
Figura 18 - A plataforma atende aos requisitos de conformidade da sua organização?	26
Figura 19 - Como avalia as funcionalidades de segurança da plataforma?	27
Figura 20 - Já enfrentou preocupações relacionadas à privacidade de dados ao usar a plataforma?	27
Figura 21 - A plataforma suporta o crescimento da sua organização?	27
Figura 22 - Já enfrentou problemas de desempenho com a plataforma em cenários de alto uso?	28
Figura 23 - Quais habilidades são mais necessárias para trabalhar com a plataforma?	28
Figura 24 - Das vantagens descritas em baixo relativamente ao uso de uma plataforma Low-Code escolha 3 que considere que se adaptem mais à sua organização	29
Figura 25 - A sua equipa precisou de treino específico para usar a plataforma?	29
Figura 26 - Como avalia a colaboração entre equipas usando a plataforma? ..	30
Figura 27 - Considera importante a preparação específica para a utilização de plataformas Low-Code?	30

Figura 28 - Se respondeu sim escolha a opção que considera mais relevante	30
Figura 29 - Embora a formação tradicional dos currículos das licenciaturas na área de informática ofereça bases para aprender Low-Code rapidamente, considera que seria útil	31
Figura 30 - Deseja partilhar mais algum comentário ou sugestão sobre o uso de tecnologias Low-Code na empresa?	31
Figura 31 – Caracterização dos atores.....	34
Figura 32 - Diagrama de Contexto.....	38
Figura 33 - Estatísticas Meta do Público do Canal Albicastrense (29 de janeiro de 2025 às 19:12)	39
Figura 34 - Caracterização da <i>persona</i> Eduardo.....	40
Figura 35 - Caracterização da <i>persona</i> Nuno	40
Figura 36 - Caracterização da <i>persona</i> Inês.....	41
Figura 37 - Diagrama de Serviços do Visitante	42
Figura 38 - Mapeamento dos Data Objects nos wireframes do Visitante.....	42
Figura 39 - Diagrama de Serviços do Jornalista	43
Figura 40 - Mapeamento dos Data Objects nos wireframes do Jornalista.....	44
Figura 41 - Diagrama de Serviços do Administrador	45
Figura 42 - Mapeamento dos Data Objects nos wireframes do Administrador.....	46
Figura 43 - <i>Wireframe</i> de Boas-Vindas à aplicação.....	48
Figura 44 - <i>Wireframe</i> da Página Principal.....	48
Figura 45 - <i>Wireframe</i> do destaque Cultura	49
Figura 46 - <i>Wireframes</i> da pequena visualização de uma Notícia	49
Figura 47 - <i>Wireframes</i> de visualização da Notícia completa.....	50
Figura 48 - <i>Wireframes</i> de Notícias com áudio e citações.....	50
Figura 49 - <i>Wireframes</i> de pesquisa de Notícias.....	51
Figura 50 - <i>Wireframe</i> de perfil com sessão iniciada	52
Figura 51 - <i>Wireframe</i> de perfil sem sessão iniciada	52
Figura 52 - <i>Wireframe</i> da Página Login.....	53
Figura 53 - <i>Wireframe</i> da Página Criar Conta.....	53
Figura 54 - <i>Wireframe</i> de Editar Dados Pessoais	54
Figura 55 - <i>Wireframe</i> de Notícias Guardadas	55
Figura 56 - <i>Wireframe</i> da Página Definições	55
Figura 57 - <i>Wireframe</i> da Página Segurança	56
Figura 58 - <i>Wireframe</i> da Página de Seleção de Linguagem.....	56
Figura 59 - <i>Wireframe</i> sobre a aplicação.....	57
Figura 60 - Início de sessão administrador	57
Figura 61 - <i>Wireframe</i> de perfil administrador.....	58
Figura 62 - <i>Wireframe</i> da Página de Definições de Administrador	58
Figura 63 - <i>Wireframe</i> de Criar Notícia.....	59
Figura 64 - <i>Wireframe</i> de Selecionar categoria da Notícia	59
Figura 65 - <i>Wireframe</i> de Publicação de Notícia	60

Figura 66- <i>Wireframe</i> da Página de Minhas Notícias	60
Figura 67 - <i>Wireframe</i> da Visualização da Notícia selecionada para leitura ou edição.....	61
Figura 68- <i>Wireframe</i> de Edição de Notícia Administrador	62
Figura 69- <i>Wireframe</i> de Edição do título e texto de uma Notícia Administrador	62
Figura 70- <i>Wireframe</i> de Edição e Adição de Conteúdos Multimédia Administrador	63
Figura 71- <i>Wireframe</i> da Edição da Categoria.....	63
Figura 72- <i>Wireframe</i> da Página Aprovar Notícias.....	64
Figura 73- <i>Wireframe</i> de Pré-Visualização da Notícia para aprovação	64
Figura 74- <i>Wireframe</i> da Notícia completa para aprovação	65
Figura 75 - <i>Wireframe</i> Página de Edição da Notícia por aprovar.....	66
Figura 76- <i>Wireframe</i> da Edição do título e texto da Notícia por aprovar	66
Figura 77- <i>Wireframe</i> de Edição e Adição Edição de conteúdos multimédia da Notícia por aprovar	67
Figura 78 - <i>Wireframe</i> da Seleção de Categoria.....	67
Figura 79 - <i>Wireframe</i> da Página de Sucesso.....	68
Figura 80 - <i>Wireframe</i> da Página de Visualizar Equipa	68
Figura 81 - <i>Wireframe</i> da Página Gerir Equipa.....	69
Figura 82 - <i>Wireframe</i> de Adicionar Membro	69
Figura 83 - <i>Wireframe</i> do Perfil Jornalista	70
Figura 84 - <i>Wireframe</i> de Editar Dados Pessoais Jornalista.....	71
Figura 85- <i>Wireframe</i> da Página Definições.....	71
Figura 86 - <i>Wireframe</i> de Nova Notícia	72
Figura 87 - <i>Wireframe</i> de Selecionar Categoria	72
Figura 88 - <i>Wireframe</i> da página de Sucesso	73
Figura 89 - <i>Wireframe</i> da Página Minhas Notícias.....	74
Figura 90 - <i>Wireframe</i> de Pré-Visualizar Notícia para edição ou leitura.....	74
Figura 91 - <i>Wireframe</i> de Edição da Notícia Jornalista.....	75
Figura 92 - <i>Wireframe</i> de Edição de título e texto de uma Notícia Jornalista ..	75
Figura 93 - <i>Wireframe</i> de Edição ou Adição de Conteúdos Multimédia Jornalista	76
Figura 94 - <i>Wireframe</i> de Seleção de Categoria.....	76
Figura 95 – <i>Wireframe</i> da Página de Sucesso.....	77
Figura 96 - <i>Wireframe</i> da Página de Visualização da Equipa	77
Figura 97 - As diferentes áreas da <i>interface</i>	78
Figura 98 - Comparação entre as áreas de possível alcance com 8 <i>interfaces</i>	79
Figura 99 - Tipografia <i>Inter</i>	79
Figura 100 - Exemplo de <i>wireframes</i> com o tipo de letra <i>Inter</i>	80
Figura 101 - Tipografia <i>Glegoo</i>	80
Figura 102 - Exemplo de <i>wireframes</i> com o tipo de letra <i>Glegoo</i>	81

Figura 103 - Tipografia <i>Istok Web</i>	81
Figura 104 - Exemplo de <i>wireframes</i> com o tipo de letra <i>Istok Web</i>	82
Figura 105 - Tipografia <i>Poppins</i>	82
Figura 106 - Exemplo de <i>wireframes</i> com o tipo de letra <i>Poppins</i>	83
Figura 107 - Cores (1).....	83
Figura 108 - Logotipo do <i>Canal Albicastrense</i>	83
Figura 109 - Cores (2).....	84
Figura 110 - <i>Wireframe</i> da Pesquisa de Notícias.....	84
Figura 111 - Imagem utilizada numa notícia real do <i>Canal Albicastrense</i> referente a educação	84
Figura 112 - Cores (3).....	85
Figura 113 - <i>Wireframes</i> de gestão da equipa.....	85
Figura 114 - Icnografia utilizada.....	85
Figura 115 - Tipos de <i>Tab Bars</i> utilizados.....	86
Figura 116 - Botões	86
Figura 117 - <i>Tabs</i> utilizadas.....	87

Lista de tabelas

Tabela 1 - Distribuição de tarefas.....	4
Tabela 2 - Resumo das diferenças entre Low-Code e No-Code.....	12
Tabela 3 - Estudos científicos analisados.....	19
Tabela 4 - Descrição dos atores.....	34
Tabela 5 - Cenários	47

Lista de abreviaturas, siglas e acrónimos

API – Application Programming Interface

AWS – Amazon Web Services

BI – Business Intelligence

CRM – Customer Relationship Management

ERP – Enterprise Resource Planning

IA – Inteligência Artificial

IEEE – Institute of Electrical and Electronics Engineers

IxD – Interaction Design (Design de Interação)

LCD – Low-Code Development

MDE – Model-Driven Engineering

MVC – Model-View-Controller

No-Code – Desenvolvimento sem Código

PRISMA – Preferred Reporting Items for Systematic Reviews and Meta-Analyses

PBNL – Programming by Natural Language

PBD – Programming by Demonstration

RPA – Robotic Process Automation

RT – Requisitos Técnicos

RNF – Requisitos Não Funcionais

RF – Requisitos Funcionais

SEO – Search Engine Optimization

TI – Tecnologia da Informação

UI – User Interface (Interface do Utilizador)

UCD – User-Centered Design (Design Centrado no Utilizador)

VPL – Visual Programming Language

1. Introdução

No contexto atual de transformação digital, a necessidade de desenvolver soluções ágeis e acessíveis tem levado ao crescimento exponencial do uso de plataformas Low-Code. Essas ferramentas oferecem um meio eficiente de criar aplicações complexas com menor esforço de programação, democratizando o desenvolvimento e permitindo que profissionais com diferentes níveis de qualificação contribuam ativamente na construção de *software*. [1] [2]

Este relatório explora o processo de planejamento e implementação de um projeto focado em Low-Code, destacando a sua aplicação como metodologia central para simplificar e acelerar o ciclo de desenvolvimento. Utilizando o paradigma Low-Code, o projeto procura, por um lado, evidenciar como estas plataformas podem transformar ideias em produtos funcionais de maneira ágil, e por outro reduzir custos, otimizando recursos e aumentando a produtividade da equipa.

A estrutura do documento reflete uma abordagem composta por duas fases: inicia-se com o **(i) enquadramento e a motivação do projeto**, passando pela análise de estudos e ferramentas relacionadas ao Low-Code, e culmina na **(ii) descrição detalhada da metodologia adotada**. Este percurso assegura que todas as etapas, desde a conceção até à prototipagem e validação, sejam documentadas de forma clara e precisa, enfatizando a importância estratégica do Low-Code no contexto do desenvolvimento de *software* moderno.

Com este projeto, procura-se não apenas demonstrar as capacidades técnicas das plataformas Low-Code, mas também refletir sobre o impacto dessa abordagem na redução de barreiras ao desenvolvimento tecnológico, promovendo a inovação colaborativa e a acessibilidade.

1.1. Enquadramento e motivação do Projeto

1.1.1. Enquadramento

Este projeto, inserido no Projeto Final de Curso da licenciatura em Engenharia Informática da Escola Superior de Tecnologia, visa aplicar os conhecimentos teóricos e práticos adquiridos ao longo do curso, desenvolvendo uma solução inovadora que atenda um problema que afeta tantas pessoas, a necessidade da criação de uma aplicação que solucione as dificuldades sendo este a falta de informação e divulgação da mesma na região da Beira Baixa.

O projeto está alinhado com as competências essenciais do curso, como desenvolvimento de *software*, engenharia de requisitos e gestão de projetos.

O projeto insere-se no contexto atual da crescente procura por soluções tecnológicas personalizadas e eficientes, explorando as possibilidades oferecidas pelas ferramentas de desenvolvimento Low-Code.

1.1.2. Motivação

Pretende-se explorar o processo de planeamento e implementação focado em tecnologia Low-Code e à posteriori implementar e testar a sua eficácia através do desenvolvimento de uma aplicação para um jornal de notícias digital, uma vez que existem poucas alternativas conhecidas para um problema que afeta tantas pessoas, surgindo a necessidade da criação de uma aplicação que solucionasse as dificuldades sendo esse a falta de informação e divulgação da mesma na região da Beira Baixa.

Por outro lado, a motivação prende-se também, pelo desejo de aplicar os conhecimentos adquiridos ao longo da nossa licenciatura, nomeadamente, no que diz respeito ao desenvolvimento ágil de aplicações, uma abordagem cada vez mais alinhada com as tendências do mercado tecnológico atual. Assim, esperamos que a nossa experiência acumulada no que diz respeito à análise de requisitos e desenvolvimento de protótipos seja uma mais-valia para o desenvolvimento do projeto.

1.2. Objetivos

Com este projeto, procura-se não apenas demonstrar as capacidades técnicas das plataformas Low-Code, estudar e refletir sobre o impacto desta abordagem, promovendo a inovação colaborativa e a acessibilidade.

A definição de objetivos permite que sejam traçadas linhas orientadoras sobre a metodologia e pesquisa, de modo a ajudar na elaboração do projeto.

O objetivo principal deste projeto é explorar a nova realidade de programação usando as tecnologias Low-Code, analisando como estas ferramentas transformaram a forma de programar e quais as competências e o perfil indicado dos novos profissionais na utilização destas novas tecnologias.

Como objetivos específicos temos:

- Analisar os artigos selecionados de forma profunda para compreensão de informações e passos importantes para a realização do projeto.
- Identificar tendências e desafios da utilização de plataformas Low-Code.
- Investigar/analisar o impacto e os benefícios envolvidos no desenvolvimento de software numa perspetiva empresarial.
- Traçar o perfil dos profissionais envolvidos.
- Desenvolvimento de um protótipo de teste atual, simples e intuitivo.
- Testar se em contexto prático, a utilidade do sistema.

1.3. Planeamento do Projeto

Como referido anteriormente, o objetivo deste projeto passa por investigar a nova realidade da programação utilizando as tecnologias Low-Code, analisando as suas aplicações numa perspetiva empresarial. O planeamento para este projeto seguiu uma estrutura organizada em etapas de forma a garantir a clareza e qualidade do estudo.

Assim, foram definidas as seguintes tarefas:

- Tarefa 1 – Revisão bibliográfica
- Tarefa 2 - Estudo e análise de trabalhos relacionados
- Tarefa 3 – Análise dos dados e interpretações
- Tarefa 4 – Desenvolvimento da Ferramenta de teste (Prototipagem)
- Tarefa 5 - Escrita e revisão do relatório

Tabela 1 - Distribuição de tarefas

	Setembro 2024	Outubro 2024	Novembro 2024	Dezembro 2024	Janeiro 2025
Tarefa 1					
Tarefa 2					
Tarefa 3					
Tarefa 4					

Tarefa 1 - Revisão bibliográfica: Uma revisão teórica utilizando a revisão sistemática de artigos utilizando a metodologia *PRISMA*, usando as principais bases de dados científicas de forma a fundamentar o estudo.

Tarefa 2 - Estudo e análise de trabalhos relacionados: Foram também analisados relatórios e casos práticos bem como *software* que utiliza tecnologia Low-Code permitindo identificar as principais tendências e lacunas relacionadas com o tema.

Tarefa 3 - Análise dos dados e interpretações: Os dados foram analisados identificando padrões de utilização, benefícios e desafios enfrentados. Como ferramentas, usaram-se ferramentas estatísticas e análises quantitativas para consolidar os resultados, tendo sido aplicados em contexto empresarial. Os resultados serão apresentados neste relatório com o objetivo de partilhar os *insights* e contribuir para a adoção estratégica da utilização do Low-Code em empresas e a nível académico.

Tarefa 4 - Desenvolvimento da Ferramenta de teste (Prototipagem): Com o desenvolvimento de uma ferramenta de teste pretendemos demonstrar como as plataformas Low-Code podem ser uma solução eficaz como alternativa às soluções tradicionais criando soluções ágeis, acessíveis e consolidando os conhecimentos adquiridos da nossa licenciatura.

2. Estado da Arte e Análise de Trabalhos Relacionados

O Estado da Arte e a Análise de trabalhos relacionados são das parte mais importantes de todo o trabalho de investigação. Nas subsecções seguintes, são descritos detalhadamente os passos seguidos na realização da revisão bem como a descrição do que é o Low-Code e o seu impacto no desenvolvimento de Software.

2.1. Estado da Arte

Irá de seguida ser apresentada as principais informações e tendências relacionadas com o tema, destacando a relevância para as tecnologias Low-Code, incluindo o seu impacto na transformação digital, os principais benefícios, os desafios enfrentados e as oportunidades emergentes.

O Low-Code tem desempenhado um papel cada vez mais central na estratégia tecnológica das empresas, permitindo reduzir significativamente o tempo e os custos associados ao desenvolvimento de aplicações, ao mesmo tempo em que promove a colaboração entre equipas técnicas e de negócios. Com a adoção crescente destas plataformas, empresas de diferentes setores estão a explorar formas de impulsionar a inovação, aumentar a eficiência operacional e responder mais rapidamente às mudanças do mercado.

Além disto, serão abordadas as diferenças entre as tecnologias Low-Code e No-Code, permitindo compreender como cada uma se adequa a diferentes cenários e necessidades empresariais. Também será analisado o impacto destas tecnologias no desenvolvimento de software, tanto em termos de vantagens como de limitações, ressaltando os fatores que devem ser considerados pelas empresas ao adotar estas soluções.

Por fim, será explorado o panorama atual das plataformas Low-Code, incluindo as ferramentas mais populares disponíveis no mercado e as suas principais características de cada uma. Esta análise permitirá evidenciar como estas soluções estão a contribuir para a democratização do desenvolvimento de software e criar novas oportunidades de negócio.

2.1.1. Tecnologias Low-Code

A tecnologia Low-Code é uma tecnologia recente que minimiza a necessidade de programação tradicional permitindo que qualquer pessoa, tenha conhecimentos ou formação em programação, ou não, consiga desenvolver soluções de *Software*, através do desenvolvimento de interfaces visuais, elementos pré-construídos e fluxos de trabalho automatizado que são convertidos em *software* funcional, promovendo ainda colaborações entre as equipas de negócios e equipas de informática.

O Low-Code tem como principais características: uma Interface Visual, utiliza um ambiente intuitivo de “arrastar e soltar”, substituindo a codificação manual por ferramentas gráficas; Automatização de Processos, permite a integração de tecnologias como automatização robótica de processos para simplificar tarefas repetidas; Desenvolvimento Rápido, reduz significativamente o tempo necessário para criar, testar e implementar aplicações; [3] Reutilização de Componentes, os elementos desenvolvidos podem ser reaproveitados em diferentes projetos, otimizando recursos; Inclusão do “cliente” no desenvolvimento, os utilizadores que solicitaram a criação da aplicação podem participar no seu desenvolvimento, contribuindo com o conhecimento do processo e diminuindo a dependência de especialistas. [4]

Os benefícios do Low-Code são: Redução de Custos, diminui a necessidade de contratar consultores externos e reduz custos com sistemas relacionados; [3] Agilidade e Escalabilidade, facilita adaptações rápidas às mudanças no mercado e escalabilidade organizacional; Aumento da Produtividade, a automatização e usabilidade simplificada melhoram a eficiência operacional; Melhoria na Colaboração: Aproxima as equipas de negócios com as de informática, garantindo que as aplicações respondam melhor às necessidades empresariais.

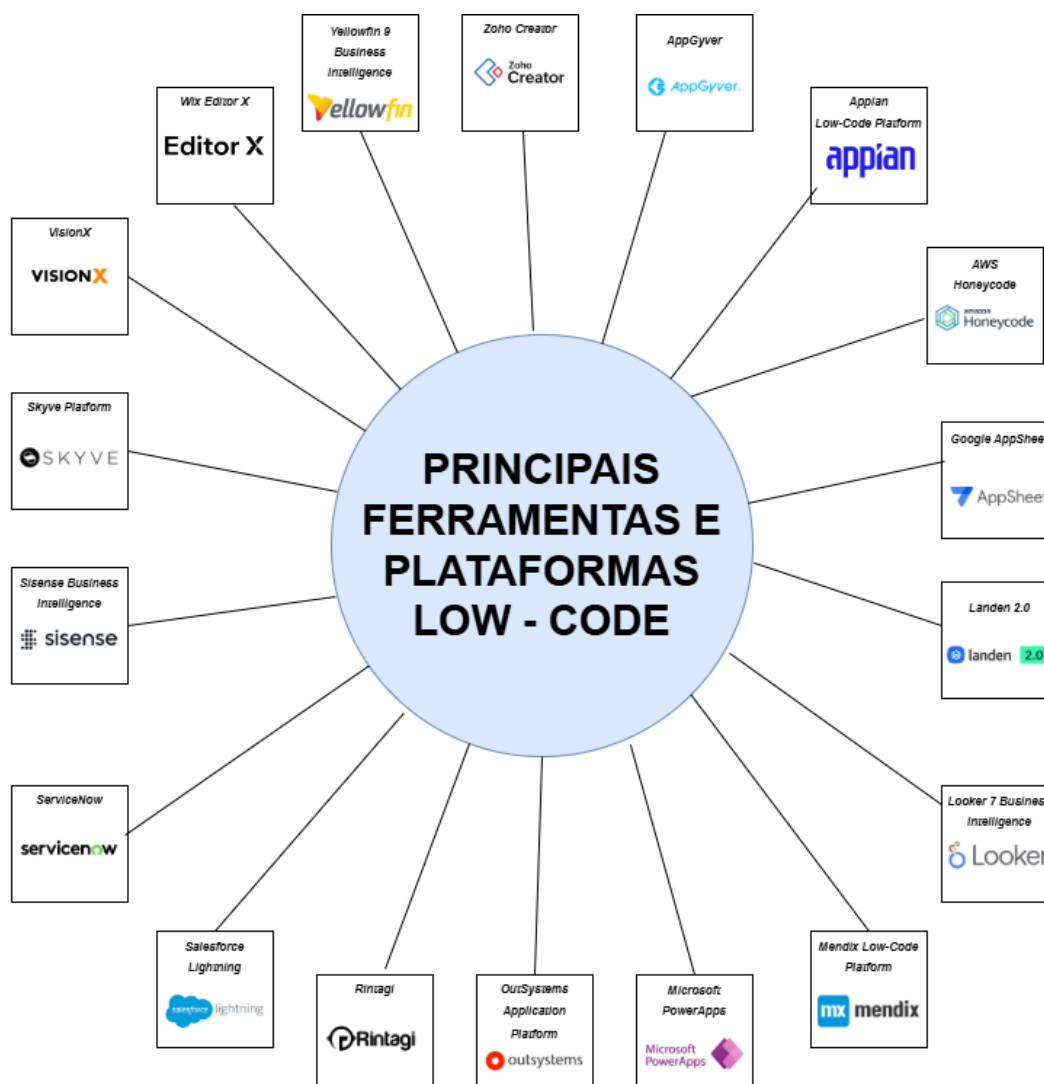


Figura 1 - Esquema com os logótipos das principais plataformas Low-Code

O Low-Code apresenta alguns desafios como a segurança, requerendo uma atenção redobrada para evitar riscos de segurança e manter conformidade regular e a necessidade de ter diretrizes bem definidas de modo a controlar o desenvolvimento e uso das plataformas de desenvolvimento.

Entre as principais ferramentas e plataformas Low-Code, destaca-se o **AppGyver**, uma ferramenta com foco no desenvolvimento de aplicações *web* e móveis, com funcionalidades Low-Code e No-Code. É ideal para protótipos rápidos ou projetos de pequena escala.

A **Appian Low-Code Platform** é uma plataforma empresarial que integra o desenvolvimento de aplicações com automatização de processos de negócio. Destaca-se pelos recursos robustos para automatização, incluindo Automatização Robótica de Processos (RPA) e inteligência artificial, facilitando a transformação digital de grandes empresas. [5]

A **AWS Honeycode** é uma solução da *Amazon* que permite criar aplicações personalizadas sem necessidade de programação avançada, tendo foco na

integração com os serviços *AWS* e sendo ideal para equipas que já utilizam a infraestrutura *cloud* da *Amazon*. [6]

O **Google AppSheet** é uma ferramenta do *Google* que permite criar aplicações simples baseadas em dados de folhas de cálculo ou outras fontes do *Google Workspace*. Destaca-se pela integração nativa com o ecossistema *Google*, permitindo criar rapidamente soluções a partir de dados existentes. [7]

O **Landen 2.0** é uma ferramenta de design para criar *websites* de forma rápida e intuitiva, sendo ideal para *freelancers* ou pequenas empresas que precisam de uma presença *online* profissional sem grande esforço técnico. [8]

O **Looker 7 Business Intelligence** é uma plataforma de inteligência empresarial que permite criar relatórios personalizados e *dashboards* interativos, com foco em análises avançadas e visualização de dados, amplamente utilizada por empresas com grandes volumes de informação. [9]

A **Mendix Low-Code Platform** é uma plataforma de desenvolvimento de aplicações empresariais que combina Low-Code com funcionalidades avançadas. É excelente para projetos complexos, com suporte para Internet das Coisas (*IoT*), Inteligência Artificial (IA) e integração com sistemas ligados. [10]

O **Microsoft PowerApps** é uma solução da *Microsoft* para criar aplicações personalizadas no ecossistema do *Office 365* e *Azure*. Destaca-se pela integração perfeita com ferramentas como *Teams*, *Dynamics* e *Excel*, sendo ideal para empresas que já utilizam o ambiente *Microsoft*. [11]

A **OutSystems Application Platform**, é uma plataforma portuguesa de referência global, é amplamente reconhecida no desenvolvimento de aplicações *web* e móveis. É altamente escalável, com foco na integração de sistemas ligados e com forte suporte a IA. [12]

O **Rintagi** é uma plataforma que permite criar aplicações empresariais personalizáveis, especializada em soluções de longo prazo com atualizações fáceis, sendo ideal para grandes empresas. [13]

O **Salesforce Lightning** é uma plataforma integrada ao *CRM Salesforce* para criar aplicações empresariais, com foco em soluções que integram dados de vendas, *marketing* e atendimento ao cliente. [14]

A **ServiceNow** é muito utilizada na gestão de serviços de Informática e fluxos de trabalho corporativos. [15]

A **Sisense Business Intelligence** é uma plataforma de análise de dados para empresas, focada em inteligência empresarial, com ferramentas avançadas de visualização e análise de dados. [16]

A **Skyve Platform** é uma plataforma *open-source* para desenvolvimento de aplicações empresariais, sendo uma excelente opção para empresas que procuram flexibilidade e personalização sem custos elevados. [17]

A **VisionX** é focada para o desenvolvimento de aplicações empresariais, com funcionalidades para personalização profunda, suportando integração com sistemas existentes, oferecendo escalabilidade para grandes empresas. [18]

O **Wix Editor X** é uma ferramenta de criação de *websites* com funcionalidades avançadas para *design* responsivo, sendo ideal para *designers* profissionais que desejam maior controle criativo sem necessidade de programar. [19]

O **Yellowfin 9 Business Intelligence** é uma plataforma de análise de dados focada em *business intelligence* e relatórios avançados, fornecendo *insights* automatizados e uma experiência visual de alta qualidade. [20]

Por fim, o **Zoho Creator** é ideal para criar aplicações empresariais, especialmente para pequenas e médias empresas, com suporte para automatização de negócios e integração com o ecossistema *Zoho*. [21]

O Low-Code está cada vez mais a ser adotado por empresas para estratégias de transformações digitais, promovendo inovações rápidas e acessíveis, minimizando barreiras técnicas.

Uma tendência recente é o surgimento de grandes fornecedores empresariais e de *cloud computing*, como a *Amazon*, o *Microsoft* e o *Google*, que competem com fornecedores tradicionais de Low-Code, como a *OutSystems*, a *Mendix* e a *Appian*. Estas empresas de grande porte têm investido significativamente para oferecer aos clientes integrações mais profundas com sistemas empresariais existentes e conjuntos de plataformas *cloud*. [22]

2.1.2. Tecnologias No-Code

O No-Code é uma abordagem de desenvolvimento de *software* que permite criar aplicações por meio de plataformas visuais, eliminando completamente a necessidade de escrever código. É uma solução que capacita pessoas sem experiência técnica ou conhecimentos em programação a desenvolver soluções digitais, automatizar tarefas e implementar ideias de forma rápida e acessível.

A proposta do No-Code é clara: facilitar o desenvolvimento de *software*. Dá autonomia a profissionais de áreas como *marketing*, vendas, recursos humanos e outros setores para criarem as suas próprias ferramentas, sem depender de equipas de Informática ou *developers* especializados. Isto é possível graças a interfaces “amigáveis” e intuitivas, que oferecem recursos no estilo “arrastar e soltar” e componentes pré-configurados para construir aplicações.

As plataformas No-Code utilizam ambientes visuais e simplificados para que os utilizadores possam montar aplicações como se estivessem a montar blocos, como por exemplo, o *LEGO*. Em vez de escrever linhas de código, os utilizadores escolhem elementos, definem ações e configuram fluxos de trabalho diretamente na interface.

Os elementos básicos, como botões, tabelas, formulários e gráficos, já estão pré-configurados e podem ser usados arrastando-os para o local desejado na aplicação.

As plataformas permitem que o utilizador crie a lógica da aplicação, como automatizações e fluxos de dados, de maneira intuitiva, configurando condições e ações com menus simples.

Embora algumas plataformas ofereçam integrações limitadas, muitas incluem conexões pré-configuradas com sistemas populares, como *Google Sheets*, *Slack*, *Mailchimp* e *CRMs*. [23]

2.1.3. Diferença entre Low-Code e No-Code

As principais diferenças entre o Low-Code e o No-Code estão relacionadas ao nível de complexidade e ao grau de envolvimento técnico necessário para desenvolver aplicações. Ambos desempenham um papel relevante no contexto empresarial, mas possuem características e aplicações distintas. [24] [25] [26]

- Complexidade e conhecimento técnico necessário

A diferença mais evidente entre Low-Code e No-Code está no nível de conhecimento técnico exigido para o desenvolvimento:

O Low-Code embora reduza significativamente a quantidade de código necessário, exige algum nível de conhecimento técnico. O *developer* precisa compreender lógica de programação, estrutura de dados e, em muitos casos, conceitos de integração com *APIs* e sistemas corporativos. Esta abordagem é voltada para *developers* profissionais ou equipas de Informática que desejam acelerar o processo de criação sem abrir mão de personalização e controlo.

Já o No-Code elimina completamente a necessidade de escrever código, sendo projetado para utilizadores sem qualquer experiência técnica. Utiliza interfaces simples, geralmente no estilo "arrastar e soltar", permitindo que qualquer pessoa crie aplicações rapidamente sem precisar aprender sobre programação ou desenvolvimento de *software*.

- Flexibilidade e personalização

Outro ponto que diferencia as duas tecnologias é o grau de flexibilidade oferecido na criação e personalização das aplicações:

O Low-Code oferece maior flexibilidade, permitindo criar aplicações complexas e personalizadas. O utilizador pode interagir diretamente com o código em partes específicas do projeto, caso necessário, para incluir funcionalidades avançadas ou integrações personalizadas.

Embora seja extremamente acessível e simples, o No-Code é limitado em termos de personalização. A maioria das ferramentas No-Code oferece funcionalidades pré-configuradas e recursos básicos, o que pode ser um entrave

para projetos que necessitem de soluções mais sofisticadas ou integrações robustas com sistemas externos.

- Público-alvo

O público-alvo de cada tecnologia também é diferente:

O Low-Code direcionado principalmente para *developers* ou equipas técnicas que procuram aumentar a sua produtividade. É útil para empresas que desejam criar aplicações complexas de forma mais rápida e eficiente, mas que ainda contam com profissionais qualificados para lidar com a parte técnica.

O No-Code é voltado para profissionais não técnicos, como analistas de negócios, gestores de projetos e equipas de *marketing* ou vendas. A sua proposta é democratizar o desenvolvimento, permitindo que qualquer colaborador tenha autonomia para criar soluções digitais.

- Velocidade de desenvolvimento

Embora ambas as tecnologias acelerem o desenvolvimento em comparação ao método tradicional, existem diferenças na velocidade com que as soluções são criadas:

O Low-Code como exige algum nível de programação ou ajustes técnicos, o tempo de desenvolvimento pode ser um pouco maior do que o No-Code, especialmente em projetos mais complexos.

O No-Code oferece um desenvolvimento extremamente ágil, permitindo que protótipos e soluções simples sejam criados em questão de horas ou dias, o que o torna ideal para tarefas que precisam de uma solução rápida.

- Integração com sistemas externos

As diferenças entre Low-Code e No-Code também se refletem na capacidade de integração com sistemas externos:

O Low-Code é amplamente utilizado para criar aplicações empresariais que precisam de se conectar a múltiplos sistemas, como *ERPs*, *CRMs* ou plataformas de dados. Permite configurar integrações avançadas e até mesmo criar conexões personalizadas.

As ferramentas da tecnologia No-Code geralmente possuem integrações pré-configuradas com sistemas populares, mas não permitem a criação de conexões mais complexas ou específicas. Isto limita a sua aplicação em ambientes empresariais mais sofisticados.

- Aplicações empresariais e casos de uso

Os tipos de projetos que podem ser desenvolvidos usando Low-Code ou No-Code variam de acordo com a complexidade e o objetivo da aplicação:

Quanto ao Low-Code, projetos de médio e grande porte, como sistemas corporativos, ferramentas de automatização empresarial e aplicações com lógica

avançada. Também é utilizado para criar soluções críticas que exigem alta escalabilidade e manutenção a longo prazo.

O No-Code é ideal para protótipos, soluções rápidas e aplicações simples, como formulários de contacto, *landing pages* ou fluxos básicos de automatização de tarefas.

- Controlo e governança

No ambiente corporativo, a governança é um fator importante:

O Low-Code por ser mais técnico permite maior controle sobre o *design*, a segurança e o desempenho das aplicações. Também é mais adequado para lidar com políticas de *compliance* e normas específicas.

No No-Code a sua simplicidade e acessibilidade podem gerar desafios de governança, especialmente em grandes organizações onde o controlo sobre as aplicações desenvolvidas por diferentes departamentos é essencial.

Resumo das diferenças

Tabela 2 - Resumo das diferenças entre Low-Code e No-Code

Aspeto	Low-Code	No-Code
Conhecimento técnico	Requer algum conhecimento técnico	Nenhum conhecimento necessário
Personalização	Alta	Limitada
Público-alvo	Developers e equipas de Informática	Utilizadores não técnicos
Velocidade	Rápida, mas com ajustes técnicos	Muito rápida
Integração	Alta capacidade de integração	Integrações pré-configuradas
Complexidade de projetos	Alta	Baixa

Para concluir, enquanto o Low-Code é uma tecnologia poderosa para *developers* acelerarem projetos complexos, o No-Code destaca-se pela simplicidade e acessibilidade, permitindo que qualquer pessoa contribua para a criação de soluções digitais. A escolha entre os dois depende das necessidades específicas do projeto e da experiência dos utilizadores envolvidos. [27] [28]

2.1.4. Impacto do Low-Code no desenvolvimento de Software

Embora os benefícios e a relevância das tecnologias Low-Code sejam amplamente reconhecidos, os estudos científicos que investigam aspectos como a produtividade proporcionada por estas ferramentas ainda são limitados. Isto destaca a importância de não apenas caracterizar as pesquisas realizadas até ao momento sobre o Low-Code, mas também de explorar detalhadamente os componentes que constituem um projeto desenvolvido com esta tecnologia, incluindo as suas particularidades, restrições, vantagens e demais características.

A limitação de estudos científicos sobre tecnologias Low-Code não é, por si só, um impacto negativo da tecnologia, mas sim um desafio relacionado ao seu estágio de adoção e maturidade no campo da pesquisa académica. Isto significa que, embora o Low-Code já demonstre benefícios práticos claros, como aumento de produtividade e inclusão de desenvolvedores cidadãos, ainda há uma lacuna na análise académica aprofundada para validar e quantificar dos seus efeitos em diferentes contextos, como produtividade e qualidade de *software*.

Portanto, a falta de estudos não reflete diretamente uma desvantagem intrínseca do Low-Code, mas sim uma oportunidade para investigadores contribuírem com análises mais detalhadas sobre os impactos desta tecnologia.

As tecnologias de desenvolvimento Low-Code apresentam uma série de impactos positivos e negativos que devem ser considerados pelas empresas.

2.1.4.1 Impactos Positivos

Quanto a impactos positivos, o Low-Code aumenta a velocidade de desenvolvimento, [3] a redução de custos, [3] acessibilidade para não Programadores [26] e Facilidade de Atualizações e Manutenção. Neste último impacto é importante destacar que a natureza modular das plataformas Low-Code facilita a implementação de atualizações e modificações, permitindo que as aplicações evoluam conforme as necessidades da empresa. [29]

2.1.4.2 Impactos Negativos

Quanto a impactos negativos, o Low-Code enfrenta desafios e limitações, limitações de personalização, embora eficazes para aplicações padrão as plataformas Low-Code podem não oferecer a flexibilidade necessária para soluções altamente personalizadas ou complexas; [4] os riscos de segurança e governança, a facilidade de desenvolvimento por utilizadores não técnicos pode resultar em aplicações que não correspondem aos padrões de segurança ou conformidade da empresa aumentando os riscos de vulnerabilidades; a dependência de plataformas, as empresas podem ficar dependentes das funcionalidades e atualizações fornecidas pelo fornecedor da plataforma Low-Code, limitando a capacidade de inovação e a adaptação às necessidades específicas; problemas de escalabilidade, aplicações desenvolvidas em plataformas Low-Code podem enfrentar desafios de desempenho e escalabilidade,

especialmente quando projetadas sem uma arquitetura adequada; [30] a complexidade na integração, ao integrar aplicações Low-Code com sistemas existentes pode ser um desafio exigindo especialização técnica. Implementar estas integrações pode limitar a atuação da solução criada e requerer esforços adicionais.

Ao considerar a adoção de tecnologias Low-Code, é fundamental que as empresas avaliem cuidadosamente esses impactos, alinhando as capacidades das plataformas às suas necessidades específicas e implementando práticas de governança eficazes para mitigar potenciais riscos.

3. Descrição detalhada da metodologia adotada

Realizou-se uma revisão sistemática de literatura sobre a tecnologia Low-Code, complementada por um estudo de caso prático de um projeto desenvolvido utilizando esta tecnologia. A revisão sistemática visou identificar, avaliar e sintetizar pesquisas e publicações relevantes que abordam o uso de Low-Code em diversos contextos, permitindo uma compreensão mais profunda das suas aplicações, benefícios e desafios. Paralelamente, o estudo de caso envolveu a análise detalhada de um projeto real, desde a concepção até a implementação, destacando as metodologias adotadas, as ferramentas Low-Code utilizadas e os resultados obtidos. Este duplo foco, combinando revisão bibliográfica e aplicação prática, permite uma avaliação mais completa e detalhada do impacto e potencial da tecnologia Low-Code em diferentes cenários.

3.1. Análise de Trabalhos Relacionados

O objetivo deste capítulo é pesquisar e analisar em detalhe estudos científicos que contenham informações sobre o uso do Low-Code, as qualificações que um profissional na área informática deve ter para exercer neste ramo bem como um protótipo de uma aplicação móvel e website de um jornal de notícias digital.

3.1.1. Estudo e análise de trabalhos relacionados

Esta revisão foi realizada de acordo com a metodologia de revisão sistemática *PRISMA* [1], uma diretriz desenvolvida para auxiliar autores na elaboração de relatórios claros e completos de revisões sistemáticas e meta-análises. A versão mais recente, *PRISMA 2020*, apresenta uma lista de verificação com 27 itens que abrangem as seções de introdução, métodos, resultados e discussão de um relatório de revisão sistemática. O seu objetivo principal é melhorar a transparência e a qualidade dos relatos de revisões sistemáticas, garantindo que os leitores possam compreender claramente o processo metodológico. [31]

Incluem-se os seguintes tópicos:

- A. Questões de pesquisa (2.1)
- B. Critérios de inclusão (2.2)
- C. Estratégia de pesquisa (2.3)
- D. Resultados (2.4)
- E. Extração e análise dos dados (2.5)
- F. Discussão (2.6).

3.1.2. Questões de pesquisa

As questões de pesquisa utilizadas foram as seguintes:

Questão 1: - Qual é o efeito do Low-Code no desempenho dos *developers*?

Questão 2: - Como o Low-Code afeta a colaboração entre os departamentos de negócio e tecnologia?

Questão 3: - O Low-Code consegue corresponder aos requisitos de segurança e escalabilidade em projetos de alta complexidade?

Questão 4: - As plataformas Low-Code facilitam a aprendizagem e aceleram o processo de desenvolvimento de software?

3.1.3. Critérios de inclusão

Neste tópico são definidos os critérios para a seleção dos estudos.

Critério 1: Estudos compreendidos entre 2015 e 2024.

Critério 2: Estudos escritos em inglês.

Critério 3: Estudos em que o texto integral encontra-se disponível.

Critério 4: Estudos que contenham informação sobre o uso de tecnologia Low-Code.

3.2. Estratégia de pesquisa

Na estratégia de pesquisa, recorreremos às bases de dados de artigos científicos: *IEEE Xplore*, *Scopus* e *ACM Digital Library*. Os termos de pesquisa usados para atender aos 3 requisitos do projeto foram: "*low code*" OR "*impact of low code*" OR "*low code vs no code*" OR "*low code platforms*".

Abrangendo tanto o Low-Code como No-Code pela fácil adaptação da informação entre os dois, garantindo mais resultados na pesquisa.

A pesquisa e a análise destes artigos aconteceram entre setembro, outubro de 2024.

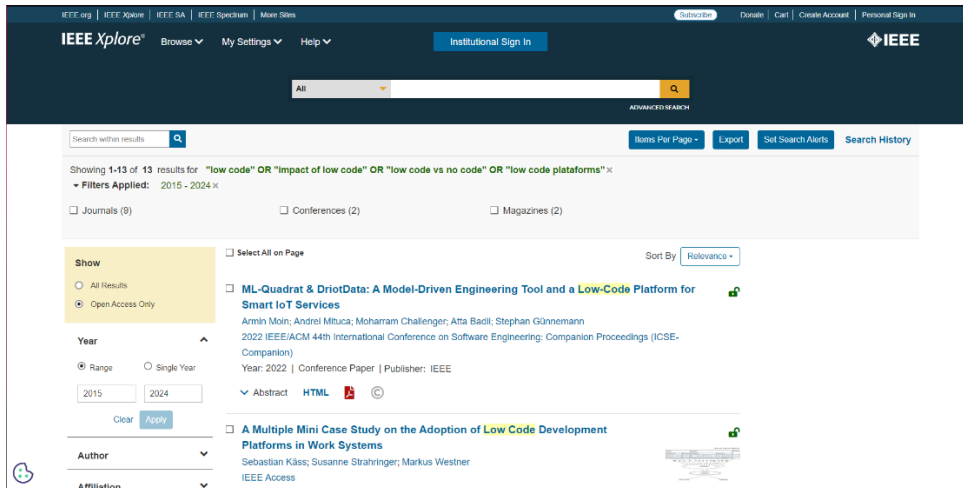


Figura 2 - Pesquisa na base de dados IEEE Xplore

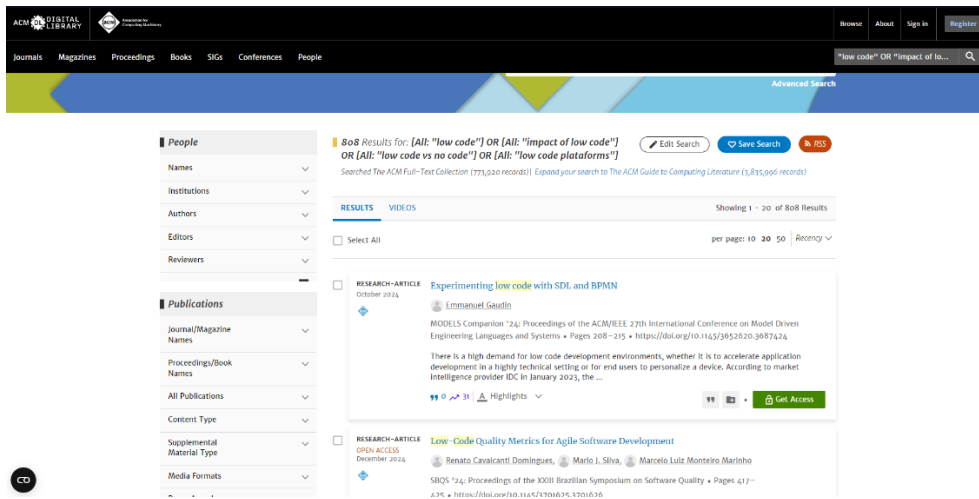


Figura 3 - Pesquisa na base de dados ACM Digital

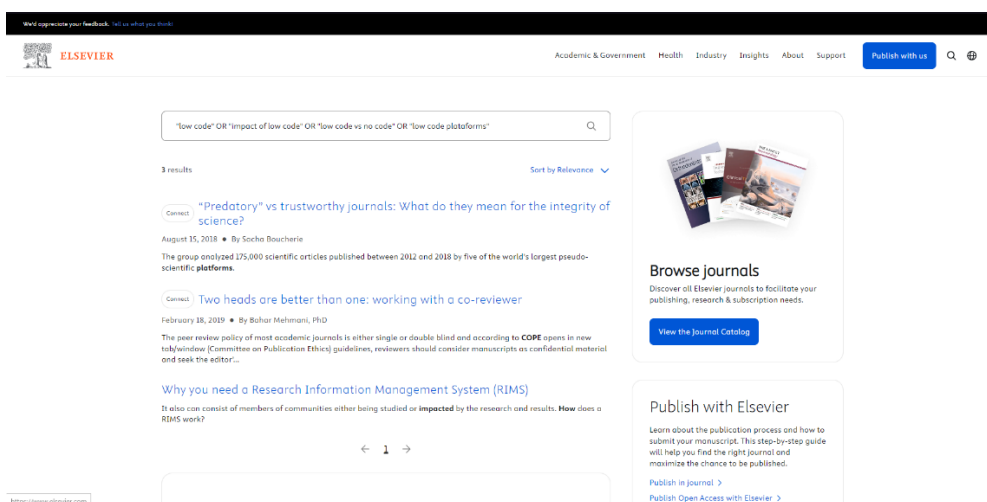


Figura 4 - Pesquisa na base de dados Scopus

3.3. Resultados

Tal como consta na Figura 5 foram encontrados 380 artigos científicos, 369 da base de dados *IEEE Xplore*, 8 da base de dados *ACM Digital* e 3 da base de dados *Scopus*. Com a aplicação do Critério 1 houve a exclusão de 60 estudos. Foram analisados 320 artigos com base no título e no resumo, sendo excluídos 296 com base no Critério 2 e 3. A análise do texto integral dos estudos científicos resultantes foi realizada e aplicando o Critério 4, foram eliminados 17 estudos. Os restantes 7 foram incluídos para revisão.

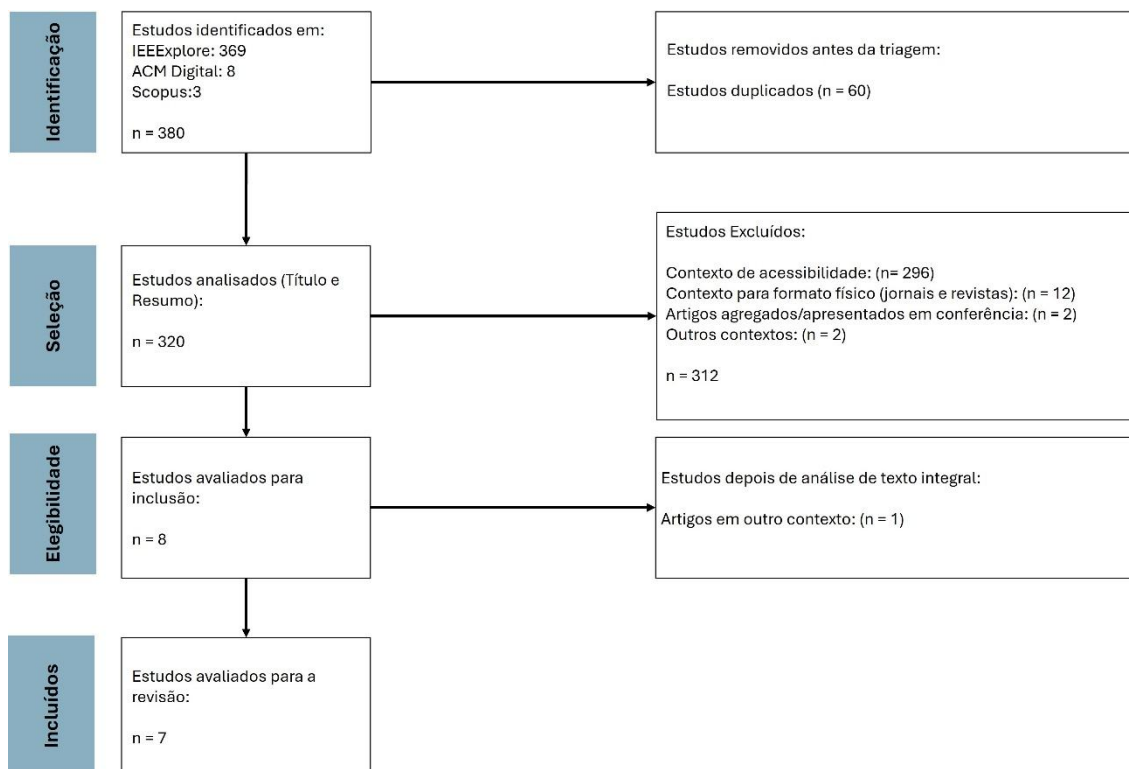


Figura 5 - Fluxograma

3.4. Extração e análise dos dados

Tabela 3 - Estudos científicos analisados

Nome	Ano de lançamento	Low Code	No Code	Programação Tradicional
The Total Economic Impact™ Of Pega Platform For Low Code [32]	2020	X	X	X
Low-Code adoption as a driver of digital transformation [33]	2024	X	X	X
Modelling in Low-Code development: a multi-vocal systematic review [34]	2022	X		
Low Code Approach for Business Analytics [35]	2023	X		
Characteristics and Challenges of Low-Code Development: The Practitioners' Perspective [36]	2021	X		
Low-Code Programming Models [37]	2023	X		
Technology Trends for 2024 [38]	2024	X	X	

O estudo “*The Total Economic Impact™ Of Pega Platform For Low Code*” [32] aborda o impacto económico da plataforma de desenvolvimento Low-Code da plataforma *Pega*, destacando como a adoção dessa solução permite acelerar a criação de aplicações críticas para o negócio ao reduzir a dependência de consultores externos e capacitar tanto *developers* quanto colaboradores não técnicos. Entre os benefícios principais estão a redução de custos de desenvolvimento em 60%, melhorias significativas na produtividade dos utilizadores finais e economias oriundas da desativação de sistemas legados. Além disso, a flexibilidade do sistema favorece a colaboração entre equipas de TI e negócio, melhorando a qualidade e a eficiência das soluções entregues, enquanto reduz os custos e complexidades de manutenção técnica. O estudo conclui que a plataforma não apenas gera um retorno sobre o investimento de 598% em três anos, mas também potencializa a transformação digital empresarial com escalabilidade contínua e maior envolvimento dos *stakeholders*.

Em “*Low-Code adoption as a driver of digital transformation*” [33] destaca-se o aumento da adoção de plataformas Low-Code como impulsionadores da transformação digital, permitindo às empresas desenvolver soluções de *software* rapidamente e com menor dependência de codificação manual. Com base numa pesquisa abrangente com 2.000 tomadores de decisão, o estudo aborda os desafios de segurança e governança, o papel estratégico destas plataformas e a sua integração com tecnologias emergentes, como inteligência artificial. Além disso, identifica tendências regionais e a importância de funcionalidades específicas para maximizar a eficiência organizacional, revelando que 81% das empresas consideram o Low-Code uma prioridade estratégica para operações mais ágeis e flexíveis. O estudo conclui que o Low-Code é essencial para enfrentar os desafios de mercado e promover a inovação.

Em relação ao terceiro estudo, “*Modelling in Low-Code development: a multi-vocal systematic review*” [34], é apresentada uma revisão sistemática multivocal sobre o desenvolvimento Low-Code, com foco especial na sua relação com a engenharia orientada por modelos (*MDE*). A análise abrangeu 58 estudos primários, incluindo literatura académica, para identificar tendências de publicação, tecnologias centrais, ferramentas de suporte e benefícios relatados. O Low-Code é descrito como um método visual e semi-automatizado para o ciclo de vida do desenvolvimento de *software*, que compartilha princípios fundamentais com o *MDE*, como abstração e automação. Entre os benefícios destacados estão o retorno melhorado sobre o investimento, o desenvolvimento baseado em interfaces gráficas e a flexibilidade, enquanto as ferramentas mais mencionadas incluem *Mendix*, *OutSystems* e *Microsoft Power Apps*. O estudo conclui que o Low-Code, apesar de estar numa fase inicial de pesquisa, é uma abordagem promissora para modernizar o desenvolvimento de *software* em diversos domínios empresariais.

O estudo “*Low Code Approach for Business Analytics*” [35] aborda o desenvolvimento de uma plataforma Low-Code projetada para facilitar a criação de *dashboards* personalizados compostos por gráficos e *widgets*. Utilizando tecnologias como *Node.js*, *Vue.js* e *MongoDB*, a plataforma permite que utilizadores, mesmo sem conhecimentos avançados de programação, possam adicionar, editar e compartilhar painéis de controlo de forma intuitiva. Os resultados destacam a simplicidade no uso e a potencial aplicação em diversos domínios, enquanto o futuro do projeto inclui a incorporação de algoritmos de Inteligência Artificial para análise de dados mais avançada e maior personalização.

Em “*Characteristics and Challenges of Low-Code Development: The Practitioners’ Perspective*” [36] são explorados as características e os desafios do desenvolvimento Low-Code a partir da perspetiva de profissionais, utilizando dados de comunidades online como *Stack Overflow* e *Reddit*. Os resultados destacam que o Low-Code oferece interfaces gráficas intuitivas com elementos como “*drag and drop*” e unidades pré-desenhadas, o que facilita o uso e acelera o desenvolvimento. Além disso, identifica os principais benefícios, como rapidez, custo reduzido e facilidade de uso, assim como limitações, incluindo falta de personalização,

dependência de fornecedores e necessidade ocasional de programação. O estudo conclui que o Low-Code é adequado para projetos com processos automatizados, mas requer uma análise cuidadosa das suas características para evitar desafios relacionados com as suas limitações.

O estudo “*Low-Code Programming Models*” [37] aborda os modelos de programação Low-Code, explorando técnicas como linguagens de programação visual (VPLs), programação por demonstração (PBD) e programação por linguagem natural (PBNL). Ele destaca o potencial do Low-Code para democratizar a criação de software, permitindo que desenvolvedores cidadãos (sem formação técnica) e profissionais colaborem. O estudo também analisa os desafios, como a ambiguidade em técnicas mais abstratas, e o impacto da inteligência artificial no avanço dessas soluções. A pesquisa propõe o padrão de *Model-View-Controller* (MVC) e linguagens específicas de domínio (DSLs) como fundamentos unificadores para ferramentas Low-Code, incentivando o uso de blocos reutilizáveis e o desenvolvimento de ambientes mais acessíveis.

O estudo “*Technology Trends for 2024*” [38] destaca mudanças significativas e continuidades em várias áreas tecnológicas. A inteligência artificial, particularmente os modelos generativos como a família GPT, emergiu como o grande disruptor, com crescimento exponencial no interesse por conteúdos relacionados. Em contraste, áreas tradicionais como desenvolvimento de *software* enfrentaram declínios, refletindo o impacto de ferramentas como o *GitHub Copilot*, que automatizam partes do processo de codificação. O estudo também ressalta o crescimento de novos paradigmas como plataformas de engenharia, o papel crescente da segurança na nuvem e a consolidação de arquiteturas distribuídas no desenvolvimento de sistemas. Apesar de 2023 ser um ano de rápida evolução, especialmente devido à IA, o estudo sublinha que a adaptação contínua e a formação em habilidades complementares, como *design* e gestão, são essenciais para profissionais manterem a sua relevância no setor.

3.5. Discussão

Como foi demonstrado até agora, o desenvolvimento Low-Code tem ganhado cada vez mais destaque, permitindo que as empresas acelerem a criação de aplicações com menos complexidade e maior eficiência. Para entender melhor o impacto desta abordagem no mercado, realizou-se um inquérito direcionado a empresas e profissionais que trabalham com Low-Code.

Neste relatório, apresentamos os principais *insights* obtidos, destacando tendências, desafios e benefícios percebidos pelos participantes. Os resultados oferecem uma visão clara sobre como o Low-Code está a ser utilizado, quais plataformas se destacam e como esta tecnologia está a moldar o futuro do desenvolvimento de software.

Nome da empresa:

31 respostas

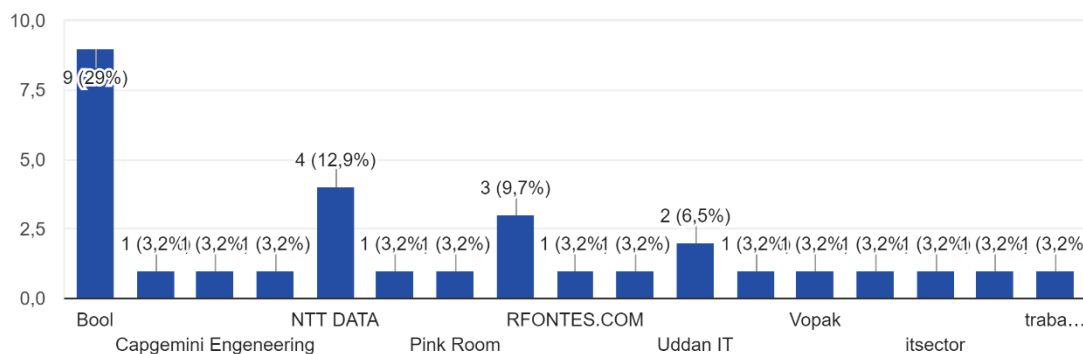


Figura 6 - Nome da empresa

Há quanto tempo a sua empresa utiliza tecnologias Low-Code?

31 respostas

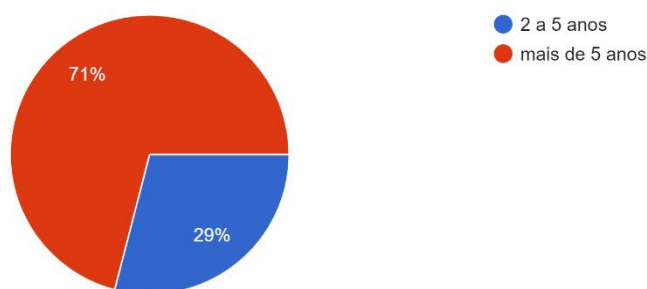


Figura 7 - Há quanto tempo a sua empresa utiliza tecnologias Low-Code?

Que plataforma(as) Low-Code são utilizadas na sua empresa? (Selecione todas as opções aplicáveis)

31 respostas

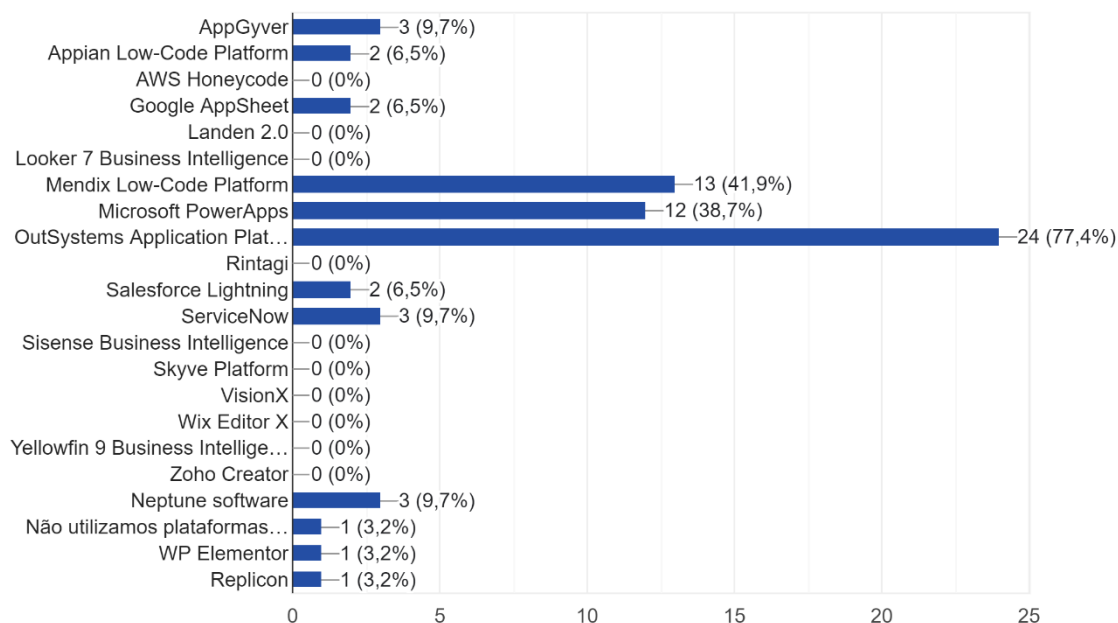


Figura 8 - Que plataforma(as) Low-Code são utilizadas na sua empresa?

Qual(ais) as principais finalidades na utilização? (Selecionar todas as opções aplicáveis)

31 respostas

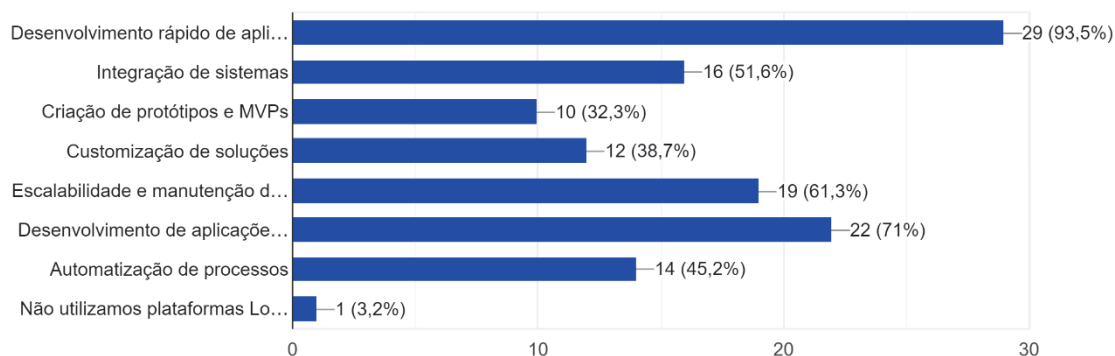


Figura 9 - Qual(ais) as principais finalidades na utilização?

A sua empresa oferece formação ou workshops para os colaboradores sobre tecnologias Low-Code?

31 respostas

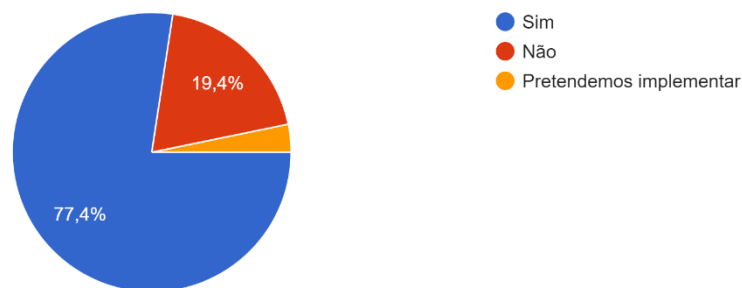


Figura 10 - A sua empresa oferece formação ou workshops para os colaboradores sobre tecnologias Low-Code?

Em que medida a plataforma Low-Code responde às necessidades específicas da empresa?

31 respostas

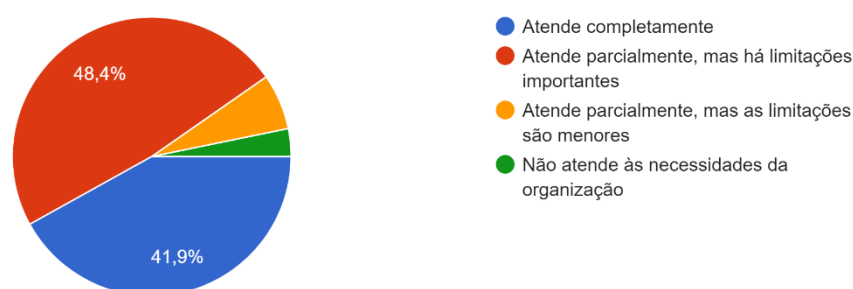


Figura 11 - Em que medida a plataforma Low-Code responde às necessidades específicas da empresa?

Quais funcionalidades de personalização sente falta? (Selecione todas que se aplicam)

31 respostas

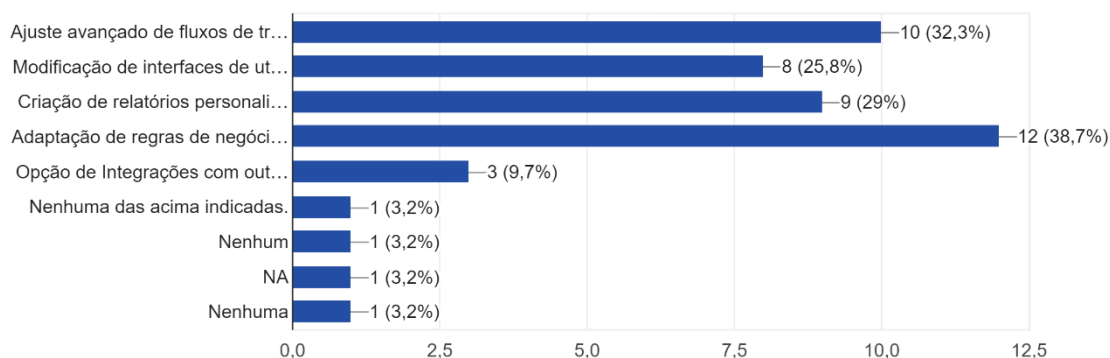


Figura 12 - Quais funcionalidades de personalização sente falta?

Como avalia a facilidade de integração da plataforma com outros sistemas?

31 respostas

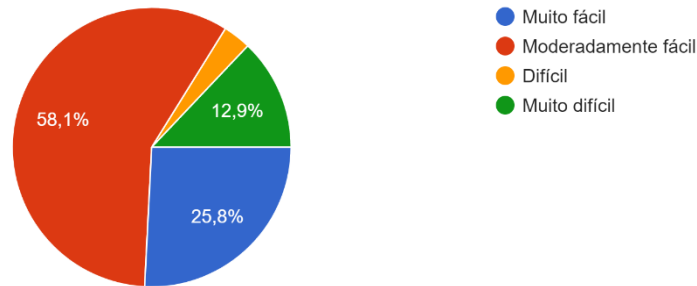


Figura 13 - Como avalia a facilidade de integração da plataforma com outros sistemas?

Enfrentou limitações técnicas ao integrar com outros sistemas?

31 respostas

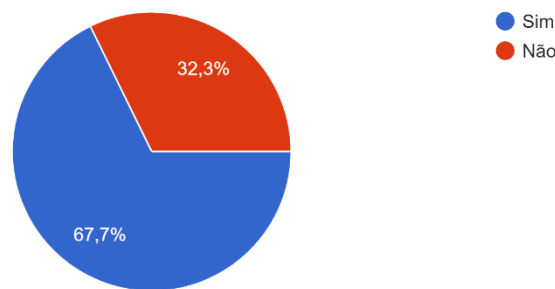


Figura 14 - Enfrentou limitações técnicas ao integrar com outros sistemas?

Quais foram as ferramentas ou sistemas que conseguiu integrar com a plataforma? (Selecione todas que se aplicam)

31 respostas

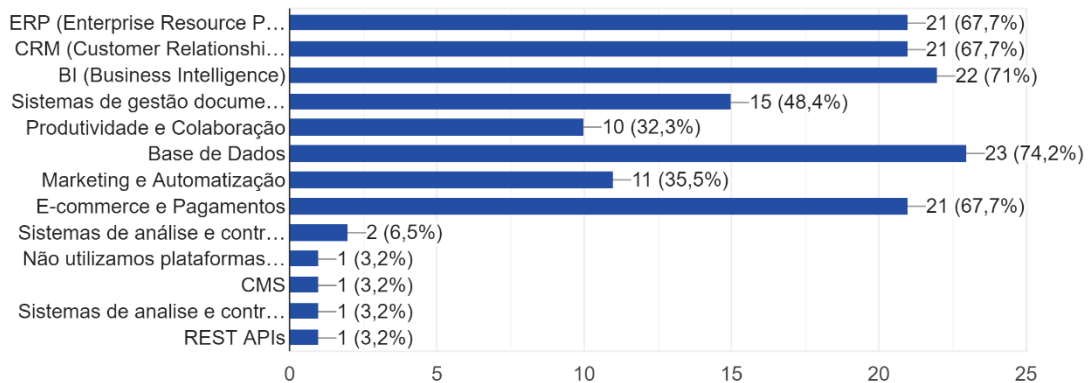


Figura 15 - Quais foram as ferramentas ou sistemas que conseguiu integrar com a plataforma?

Qual o nível de dependência do suporte do fornecedor?

31 respostas

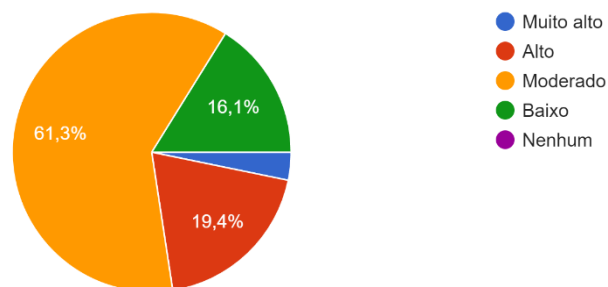


Figura 16 - Qual o nível de dependência do suporte do fornecedor?

A documentação fornecida pelo fornecedor é suficiente para uso independente?

31 respostas

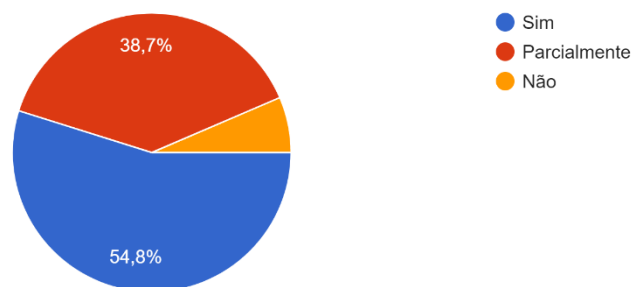


Figura 17 - A documentação fornecida pelo fornecedor é suficiente para uso independente?

A plataforma atende aos requisitos de conformidade da sua organização?

31 respostas

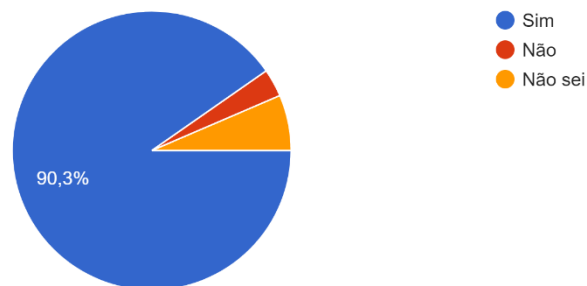


Figura 18 - A plataforma atende aos requisitos de conformidade da sua organização?

Como avalia as funcionalidades de segurança da plataforma? (Escala de 1 a 5)

31 respostas

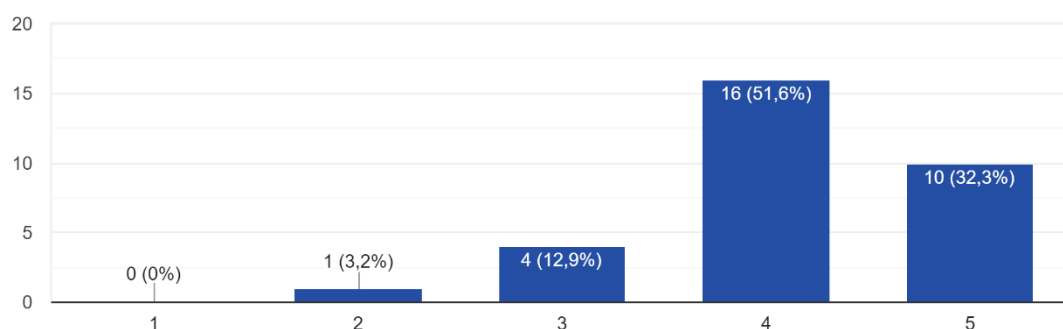


Figura 19 - Como avalia as funcionalidades de segurança da plataforma?

Já enfrentou preocupações relacionadas à privacidade de dados ao usar a plataforma?

31 respostas

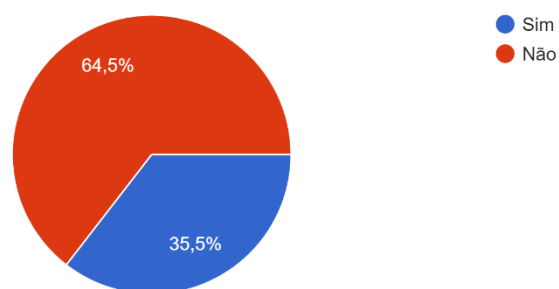


Figura 20 - Já enfrentou preocupações relacionadas à privacidade de dados ao usar a plataforma?

A plataforma suporta o crescimento da sua organização?

31 respostas

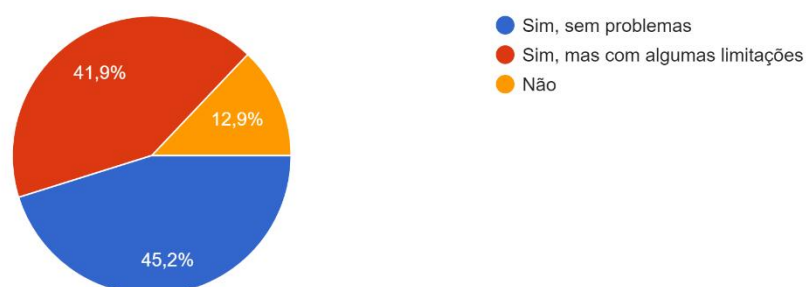


Figura 21 - A plataforma suporta o crescimento da sua organização?

Já enfrentou problemas de desempenho com a plataforma em cenários de alto uso?

31 respostas

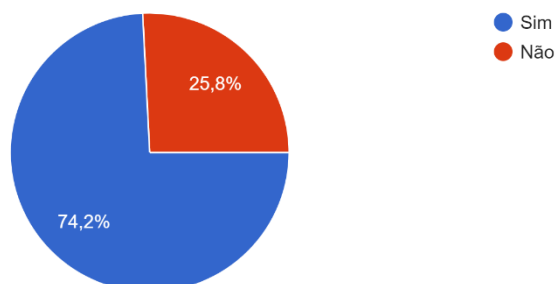


Figura 22 - Já enfrentou problemas de desempenho com a plataforma em cenários de alto uso?

Quais habilidades são mais necessárias para trabalhar com a plataforma? (Selecione todas que se aplicam)

31 respostas

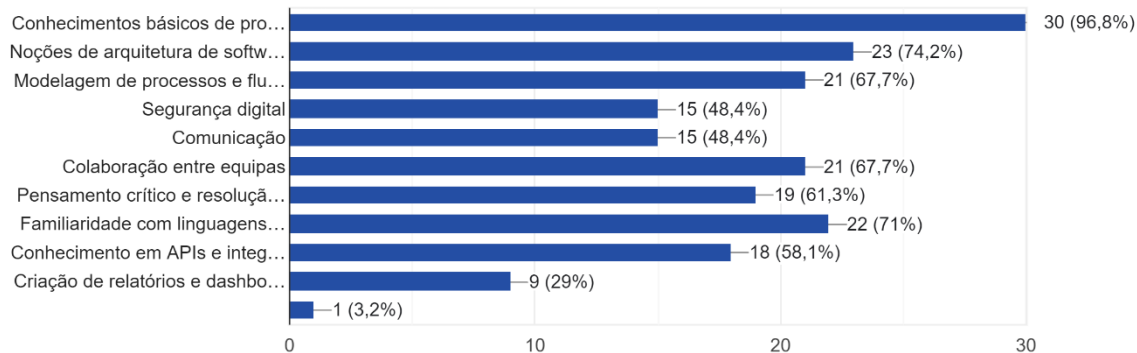


Figura 23 - Quais habilidades são mais necessárias para trabalhar com a plataforma?

Das vantagens descritas em baixo relativamente ao uso de uma plataforma Low-Code escolha 3 que considere que se adaptem mais à sua organização:

31 respostas

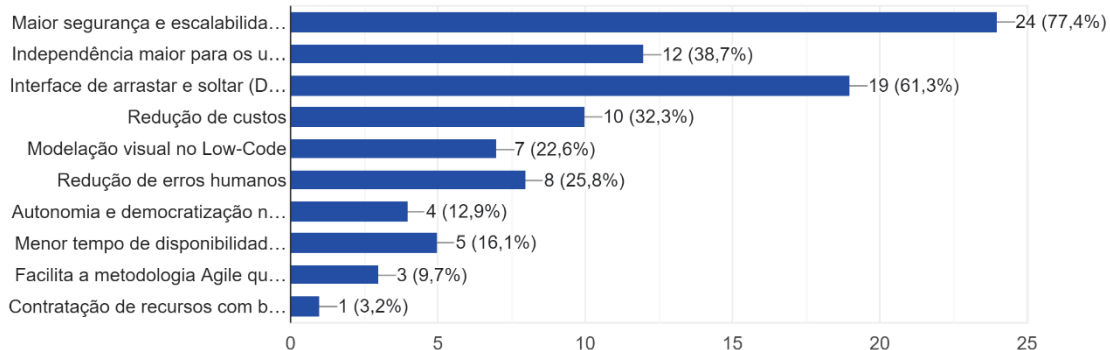


Figura 24 - Das vantagens descritas em baixo relativamente ao uso de uma plataforma Low-Code escolha 3 que considere que se adaptem mais à sua organização

A sua equipa precisou de treino específico para usar a plataforma?

31 respostas

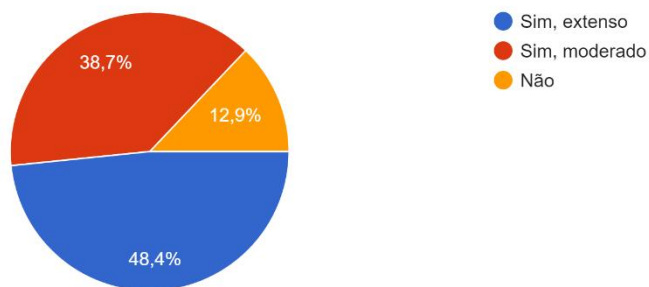


Figura 25 - A sua equipa precisou de treino específico para usar a plataforma?

Como avalia a colaboração entre equipas usando a plataforma? (Escala de 1 a 5)

31 respostas

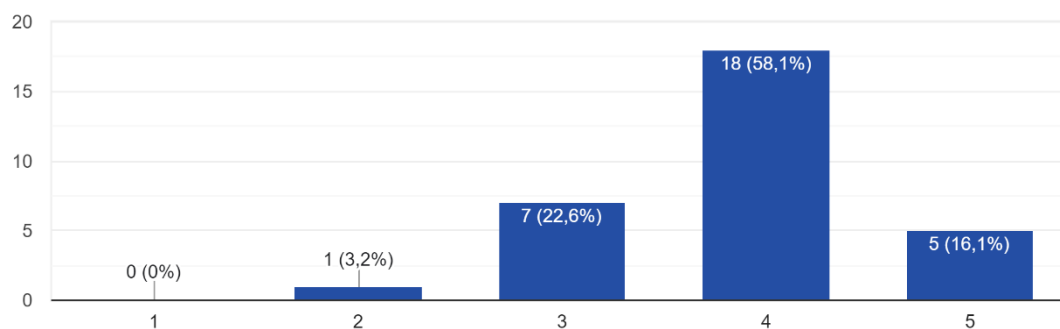


Figura 26 - Como avalia a colaboração entre equipas usando a plataforma?

Considera importante a preparação específica para a utilização de plataformas Low-Code?

31 respostas

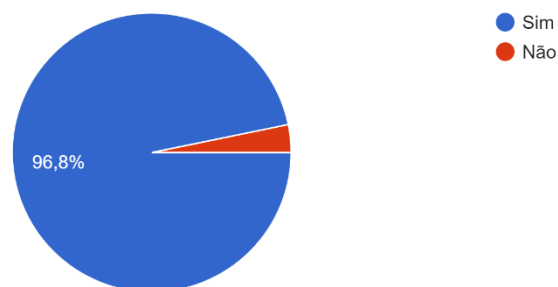


Figura 27 - Considera importante a preparação específica para a utilização de plataformas Low-Code?

Se respondeu sim escolha a opção que considera mais relevante:

30 respostas

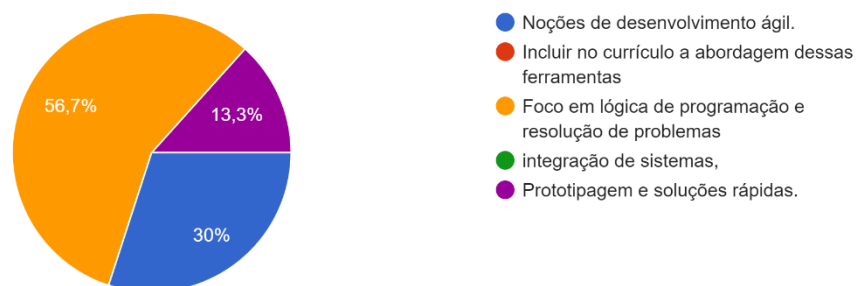


Figura 28 - Se respondeu sim escolha a opção que considera mais relevante

Embora a formação tradicional dos currículos das licenciaturas na área de informática ofereça bases para aprender Low-Code rapidamente, consid...olha as 3 opções que considere mais relevantes)

31 respostas

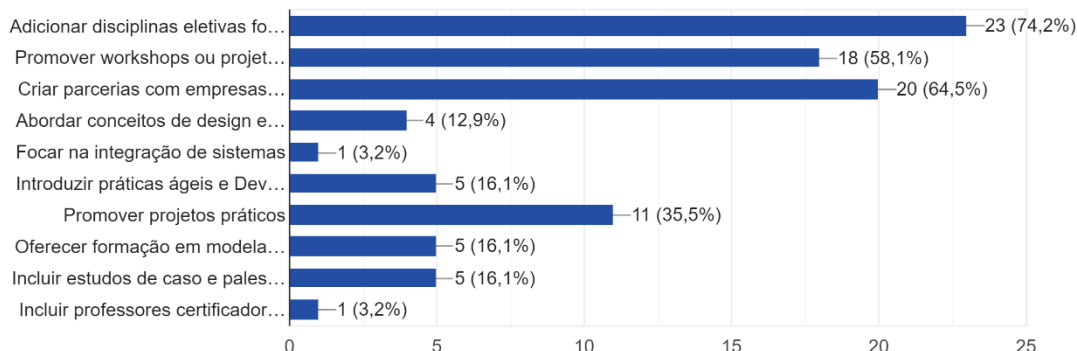


Figura 29 - Embora a formação tradicional dos currículos das licenciaturas na área de informática ofereça bases para aprender Low-Code rapidamente, considera que seria útil

Deseja partilhar mais algum comentário ou sugestão sobre o uso de tecnologias Low-Code na empresa? 5 respostas

Continuação de um bom trabalho

Não utilizamos plataformas Low-Code, algumas das questões não estavam adequadas a responder que não era utilizado plataforma de low-code.

Penso que a programação tradicional e o low code tem as duas mercado, no entanto a low code tem uma vantagem na questão do tempo de desenvolvimento que é menor, entendo que as licenciaturas de IT tem de obrigatoriamente hoje tem de dar ferramentas aos alunos para o que o mercado pede

Bom trabalho 🤖

Bom trabalho 🙌

Figura 30 - Deseja partilhar mais algum comentário ou sugestão sobre o uso de tecnologias Low-Code na empresa?

Os resultados recolhidos sobre o uso de tecnologias Low-Code nas empresas revelam diferentes perspetivas sobre a sua adoção e impacto. No geral, as respostas demonstram uma visão equilibrada entre os benefícios e os desafios desta abordagem no contexto corporativo.

Um dos pontos levantados é que algumas empresas ainda não utilizam plataformas Low-Code, o que pode indicar uma falta de necessidade, conhecimento

ou uma infraestrutura já estabelecida que não exige mudanças tecnológicas nesse sentido. Essa resistência pode estar relacionada a questões culturais, custos de implementação ou mesmo à percepção de que a programação tradicional oferece maior controle e flexibilidade. Isso reforça a importância de capacitação e informação para ampliar a adoção destas ferramentas, especialmente para empresas que procuram maior agilidade sem comprometer a segurança e a qualidade do desenvolvimento de *software*.

Outro ponto relevante é a percepção de que tanto a programação tradicional quanto o Low-Code possuem espaço no mercado. Enquanto a programação tradicional continua a ser essencial para soluções mais complexas e personalizadas, o Low-Code destaca-se pela agilidade no desenvolvimento de aplicações, reduzindo significativamente o tempo entre a concepção da ideia e da sua implementação prática. Esta característica pode ser particularmente vantajosa para empresas que procuram inovação rápida e eficiência operacional, como *startups*, setores dinâmicos e organizações que precisam de soluções tecnológicas adaptáveis às mudanças do mercado. Além disso, o Low-Code pode ser um facilitador para profissionais de outras áreas, permitindo que desenvolvam soluções tecnológicas sem a necessidade de um conhecimento aprofundado em linguagens de programação, promovendo uma maior inclusão digital dentro das empresas.

Além disso, um dos respondentes destacou a necessidade de que as licenciaturas em Tecnologia da Informação incluam ferramentas Low-Code nos seus currículos. Esse apontamento reflete a exigência crescente por profissionais que saibam utilizar essas plataformas, alinhando a formação acadêmica às exigências do mercado. A formação tradicional em Informática historicamente prioriza linguagens de programação convencionais, mas com a crescente adoção de soluções Low-Code e No-Code, torna-se essencial que os currículos acadêmicos sejam atualizados para contemplar esta realidade, preparando os futuros profissionais para um mercado de trabalho cada vez mais dinâmico e automatizado.

Por fim, observa-se que a percepção geral é positiva, com diversos elogios ao trabalho desenvolvido. Isso sugere que, mesmo diante das diferenças de adoção e conhecimento sobre o Low-Code, há reconhecimento do seu valor e potencial para otimização de processos empresariais. A evolução constante destas tecnologias indica que a sua integração no mercado tende a crescer, conforme mais empresas percebam os seus benefícios e desafios.

3.6. Protótipo

Como referimos anteriormente um dos objetivos deste projeto é o desenvolvimento de um protótipo usando uma plataforma Low-Code a fim de avaliar a sua eficácia, facilidade de utilização e viabilidade para o desenvolvimento de aplicações.

Nesta fase vamos apresentar o protótipo desenvolvido sendo depois a sua implementação realizada em Projeto II (implementação e teste).

Com este protótipo funcional pretende-se criar uma aplicação de notícias que permita aos jornalistas publicar e gerir conteúdo de maneira simples e eficiente. O portal será acessível ao público, que poderá visualizar, filtrar e guardar notícias, além de receber notificações personalizadas sobre novas publicações de acordo com suas preferências, entre outras funcionalidades.

O desenvolvimento passa por utilizar uma plataforma de desenvolvimento Low-Code, investigando-a e comparando-a com a abordagem No-Code e com a programação tradicional.

Optou-se por criar uma aplicação para o jornal recente da cidade de Castelo Branco, o *Canal Albicastrense*. Este publica as suas notícias nas redes sociais, *Facebook* e *Instagram*. Achou-se por bem desenvolver uma aplicação para este, visto poder-se ajudar a impulsioná-lo e um dos elementos do trabalho pertencer à equipa do meio de comunicação.

Nesta secção são apresentados os requisitos que definem a proposta de solução para o desenvolvimento da aplicação do *Canal Albicastrense*. Através da criação do Diagrama de Contexto, identificam-se os diferentes utilizadores do sistema (Visitante, Jornalista e Administrador), bem como os *stakeholders* envolvidos, que contribuem para o seu funcionamento. Além disto, são descritas as *personas* para evidenciar a abordagem centrada no utilizador do *design* de interação, o que facilita a criação de cenários de uso.

A Arquitetura da Informação organiza as atividades do sistema, permitindo definir os requisitos com clareza. Por meio desta estruturação, é possível criar o Diagrama de Serviços, que mapeia os serviços disponíveis e identifica os objetos informacionais (*Data Objects*). A integração do conhecimento obtido no Diagrama de Contexto e no Diagrama de Serviços serve de base para a criação de cenários detalhados, que descrevem os fluxos de trabalho e definem os requisitos funcionais do sistema de forma precisa.

Principais Funcionalidades/Requisitos do sistema

Para perceber os requisitos, as funcionalidades e necessidades da plataforma foram numa primeira fase descritos e analisado o público-alvo e os intervenientes da aplicação.

O público-alvo da aplicação serão o Visitante, o Jornalista e o Administrador.

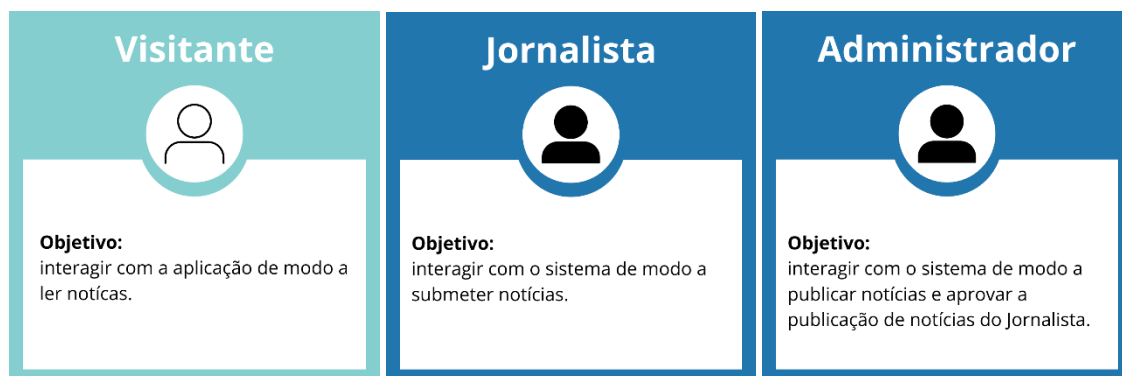


Figura 31 – Caracterização dos atores

Identificámos três diferentes tipos de atores representados na figura 31, que definem o *User Model*: Visitante (o único que interage diretamente com o sistema como utilizador final), Jornalista e Administrador (atores que interagem indiretamente com o sistema, mas desempenham papéis essenciais no seu funcionamento). A tabela 4 apresenta os papéis atribuídos a estes atores durante a sua interação com o sistema.

Tabela 4 - Descrição dos atores

Atores	Descrição
Visitante	Interage diretamente com a aplicação para consultar notícias publicadas, filtrar conteúdos de interesse, guardar notícias favoritas, criar uma conta de utilizador, configurar preferências para receber notificações personalizadas e aceder a informações gerais sobre a aplicação.
Jornalista	Interage diretamente com a aplicação para redigir, editar e submeter notícias para revisão. Pode gerir os conteúdos associados à sua autoria, mas a publicação efetiva das notícias depende da aprovação do Administrador.
Administrador	Interage diretamente com a aplicação para publicar notícias, aprovar ou editar publicações submetidas pelos Jornalistas e gerir a equipa (adicionando ou removendo utilizadores, passando a ser Jornalista).

Após reuniões com a equipa do projeto definiu-se os seguintes requisitos funcionais, não funcionais e técnicos:

Requisitos Funcionais

RF.01 - Ver notícias

O sistema deve permitir que todos os atores (Visitante, Jornalista e Administrador) possam ver as notícias publicadas.

O sistema deve permitir que todos os atores (Visitante, Jornalista e Administrador) possam guardar a notícia.

O sistema deve permitir que todos os atores (Visitante, Jornalista e Administrador) possam partilhar notícias via redes sociais.

RF.02 - Fazer registo

O sistema deve permitir o registo do Utilizador.

RF.03 - Ver notícias guardadas

O sistema deve permitir que todos os atores (Visitante, Jornalista e Administrador) possam ver as notícias guardadas.

RF.04 - Pesquisar notícias

O sistema permite que todos os atores (Visitante, Jornalista e Administrador) possam pesquisar notícias pelos títulos ou pelas categorias.

RF.05 - Escrever notícias

O sistema deve permitir que o Jornalista, após ter feito o registo e ser adicionado pelo Administrador à equipa, possa escrever notícias e submeterem para o Administrador possa aprovar a publicação das mesmas.

O sistema deve permitir que o Administrador possa escrever notícias e publicá-las, ficando disponíveis para todos os atores.

RF.06 - Aprovar publicação de notícias

O sistema deve permitir que o Administrador após receber as notícias escritas pelos Jornalistas, possa ler, editar e publicar as mesmas, ficando assim disponíveis para todos os atores.

RF.07 - Ver notícias publicadas por mim

O sistema deve permitir que tanto o Jornalista como o Administrador possam ver as notícias que foram publicadas por eles.

RF.08 - Editar notícias

O sistema deve permitir que o Jornalista possa editar as suas notícias, submetendo para aprovação do Administrador.

O sistema deve permitir que o Administrador pode editar as suas notícias, publicando logo de modo que todos os atores tenham acesso às alterações.

RF.09 - Aprovar alterações a notícias

O Administrador pode aprovar as alterações realizadas pelo Jornalista, ou editá-las, ficando disponíveis para todos os atores.

RF.10 - Ver equipa

Tanto o Jornalista como o Administrador podem ver a equipa.

RF.11 - Gerir equipa

O Sistema deve permitir a gestão da equipa pelo Administrador, incluindo as ações:

- Adicionar elementos à equipa.
- Remover elementos à equipa.
- Alterar o nome do jornal.

Requisitos não funcionais:

RNF.1 - A aplicação deve ser responsiva e funcionar em diferentes dispositivos.

RNF.2 - O sistema deve suportar um número elevado de utilizadores e simultâneo.

RNF.3 - A aplicação deve garantir seguranças e privacidade dos dados aos utilizadores.

RNF.4 - O sistema deve estar disponível em pelo menos dois idiomas.

Requisitos técnicos:

RT.1 - A aplicação deve ser desenvolvida usando Low-Code.

RT.2 - O *Back-end* deve ser implementado usando tecnologia Low-Code.

RT.3 - A base de dados deve ser compatível com a tecnologia Low-Code escolhida.

RT.4 - A utilização de *APIs* externas deve ser aplicado como necessário.

3.6.1. Diagrama de Contexto

O seguinte Diagrama de Contexto representa as interações entre os três atores identificados (**Visitante**, **Jornalista** e **Administrador**), o Sistema de Informação e a aplicação. Durante o funcionamento, observa-se o comportamento de cada elemento crítico, representado na figura 32:

- O **Visitante** interage diretamente com a aplicação, acedendo à mesma para consultar notícias, guardar conteúdos do seu interesse, criar uma conta, personalizar notificações e filtrar informações de acordo com suas preferências.

- A **aplicação** responde às atividades realizadas pelo Visitante (exemplo, guardar uma notícia ou configurar preferências de notificação) e transmite as ações relevantes para os componentes internos do sistema, que as processam de acordo com os requisitos estabelecidos.

- O **Administrador** atua no sistema para publicar notícias, editar notícias e aprovar ou rejeitar conteúdos submetidos pelos Jornalistas.

- O **Jornalista** utiliza a aplicação para redigir, editar e submeter notícias para aprovação. Após a validação pelo Administrador, os conteúdos tornam-se disponíveis para os todos os atores.

Este diagrama reflete a interação entre os diferentes atores e o fluxo de informações, evidenciando como a aplicação suporta as atividades e garante o funcionamento eficiente do sistema de notícias.

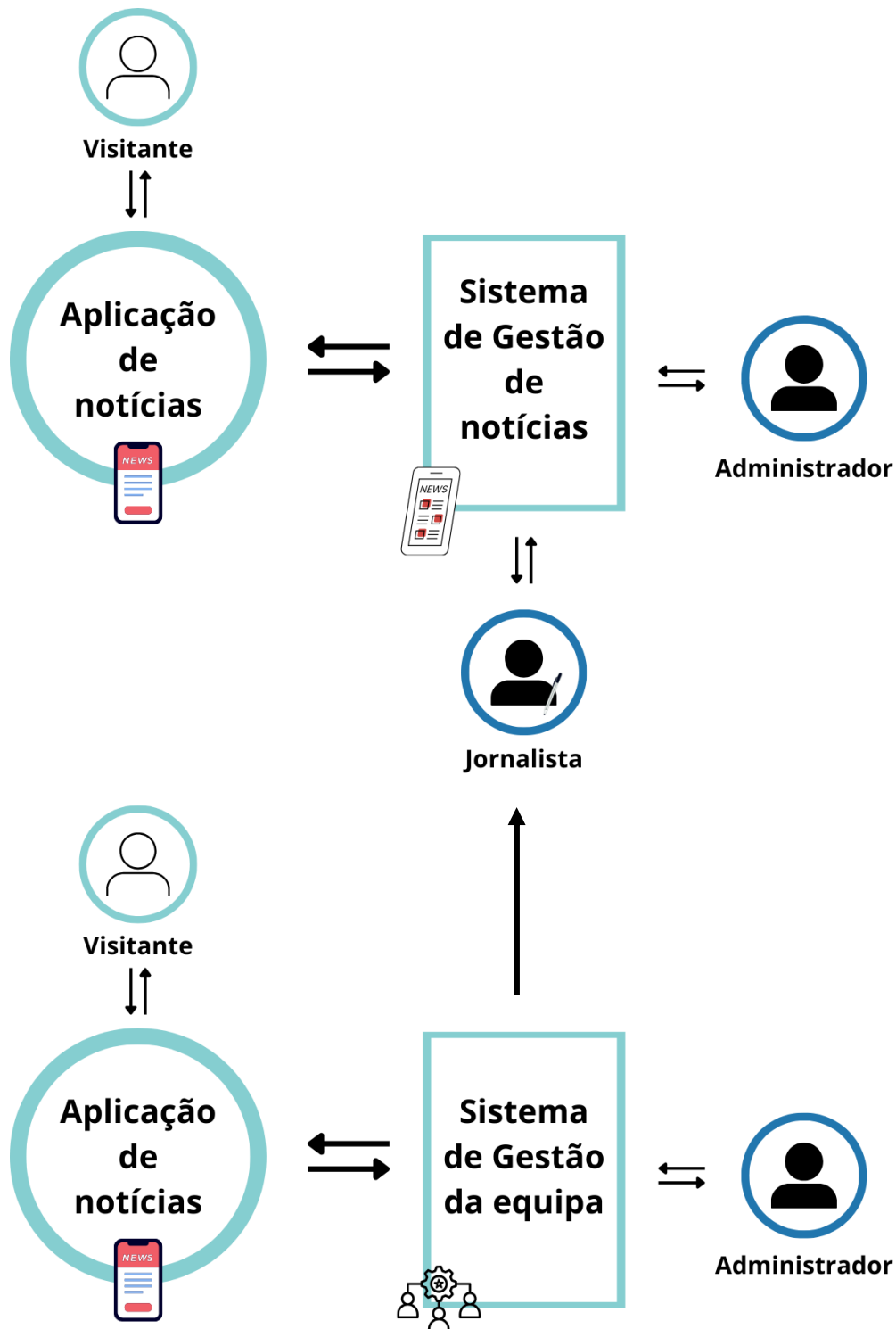


Figura 32 - Diagrama de Contexto

O presente projeto baseia-se nos princípios do *Design de Interação (IxD)* e, por esse motivo, durante a atividade de estabelecimento de requisitos, especialmente os do utilizador final (Visitante), adotámos a abordagem *User-Centered Design*

(*UCD*), que envolve a participação dos utilizadores no processo de conceção do design (fase de ideação) e de interação com a interface – *UI* [39]. Seguindo a abordagem *UCD*, introduzimos um modelo que representa os potenciais utilizadores do sistema, designado por *personas*. Estas são baseadas nos comportamentos, necessidades e motivações de pessoas reais. As *personas* permitem formalizar os padrões de interação, combinando-os com os padrões de comportamento e os objetivos dos utilizadores [40].

No contexto do sistema de notícias do *Canal Albicastrense*, os utilizadores que interagem com a aplicação incluem **Visitantes**, **Jornalistas** e **Administradores**. O público **Visitante** deve ter, preferencialmente, idade igual ou superior a 18 anos, familiaridade com tecnologias digitais e acesso a *smartphones* ou dispositivos semelhantes, visto o *Canal Albicastrense* ter como público atual pessoas a partir dos 18 anos, como podemos observar na figura 33. A criação de *personas* no processo de *IxD* orienta as metas de design e garante que o sistema responda às necessidades e objetivos de cada grupo de *stakeholders*.

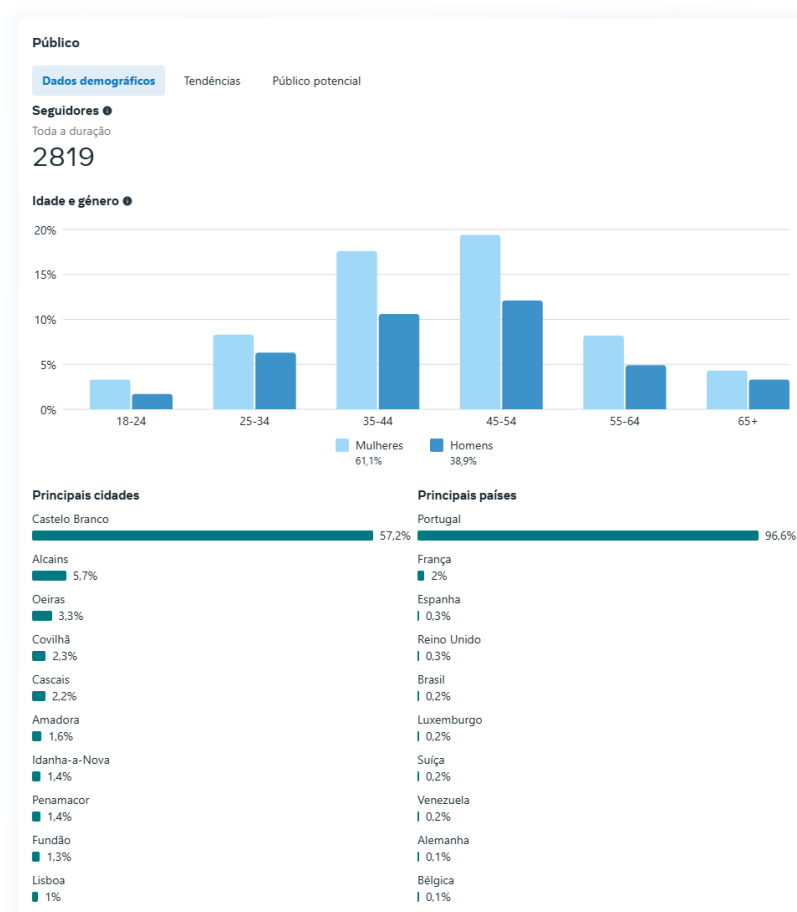


Figura 33 - Estatísticas Meta do Público do Canal Albicastrense (29 de janeiro de 2025 às 19:12)

Na secção 3.6.3., definimos três cenários de contexto atribuídos a cada *persona*, evidenciando o comportamento e as interações específicas de cada utilizador com o sistema.

Eduardo Alves


- ⇒ Nativo Digital
- ⇒ 23 anos
- ⇒ Masculino
- ⇒ Nasceu e vive em Castelo Branco
- ⇒ Solteiro
- ⇒ **Estudante Universitário de Engenharia Informática**

Utiliza o seu *smartphone* principalmente para jogar e fazer pesquisas sobre Tecnologia, visto gostar muito deste tema.

Como é estudante de Engenharia Informática é também muito importante estar a par das mudanças que a Tecnologia tem ao longo do tempo.

Objetivo:

Existir um sistema em que permita consultar todas as notícias sobre Tecnologia, de preferência uma aplicação nativa, que disponibilize a funcionalidade de envio de notificações para alertar, por exemplo sobre uma nova notícia sobre Inteligência Artificial.



"As notificações sobre as notícias de Tecnologia ajudam-me a estar mais informado sobre esta área."

Figura 34 - Caracterização da *persona* Eduardo


Nuno Baptista

- ⇒ Nativo Digital
- ⇒ 22 anos
- ⇒ Masculino
- ⇒ Nasceu e vive em Castelo Branco
- ⇒ Solteiro
- ⇒ **Diretor-Geral e Diretor de Informação de um meio de comunicação**

Utiliza o seu *smartphone* e *tablet* principalmente para gerir o seu meio de comunicação e escrever notícias.

Objetivo:

Existir um sistema em que permita aprovar a publicação de notícias e gerir a equipa em qualquer lugar.



"A gestão da equipa do meu meio de comunicação facilita-me a conseguir estar em qualquer lugar e poder aprovar a publicação de uma notícia ou adicionar um elemento à equipa."

Figura 35 - Caracterização da *persona* Nuno

Inês Azinheiro

- ⇒ Nativo Digital
- ⇒ 22 anos
- ⇒ Feminino
- ⇒ Nasceu e vive em Castelo Branco
- ⇒ Solteiro
- ⇒ **Jornalista de um meio de comunicação**

Utiliza o seu *smartphone* e *tablet* principalmente para escrever notícias quando está no local de uma notícia.

Objetivo:

Existir um sistema em que permita escrever e submeter para aprovação notícias em qualquer lugar.

“A escrita e envio para aprovação de notícias em qualquer lugar facilita-me, de modo a que eu esteja, por exemplo, a fazer uma cobertura da chegada de um Ministro a um local e poder escrever a notícia no momento.”



Figura 36 - Caracterização da *persona* Inês

3.6.2. Diagrama de Serviços

Para a arquitetura da informação (representação visual onde descrevemos a estrutura e a organização do sistema de forma a mostrara a comunicação entre as diferentes partes do sistema e a forma como se relacionam e interagem entre si) usamos o Diagrama de Serviços, que mapeia os serviços disponíveis e identifica os objetos informacionais (*Data Objects*).

De forma a estruturar o diagrama de serviços, utilizámos técnicas da Arquitetura da Informação para definir a estrutura e o conteúdo com uma abordagem centrada no utilizador.

Apresentamos de seguida os serviços que o sistema oferece que posteriormente resultam na ideação da sua interface, foram introduzidos os objetos informacionais (*Data Objects*), que correspondem a uma determinada interface que suporta um serviço de sistema, de forma a identificar quais os dados necessários que uma atividade necessita para executar corretamente a sua função. Com a introdução dos data objects conseguimos identificar qual o comportamento e estrutura de cada atividade ou função, contribuindo para a definição do design da interface. [41]

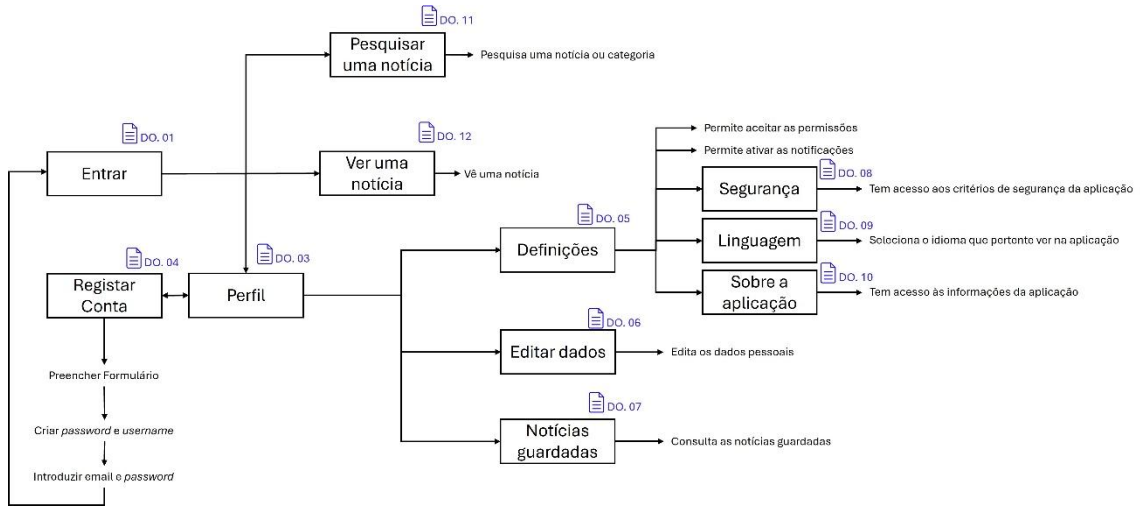


Figura 37 - Diagrama de Serviços do Visitante

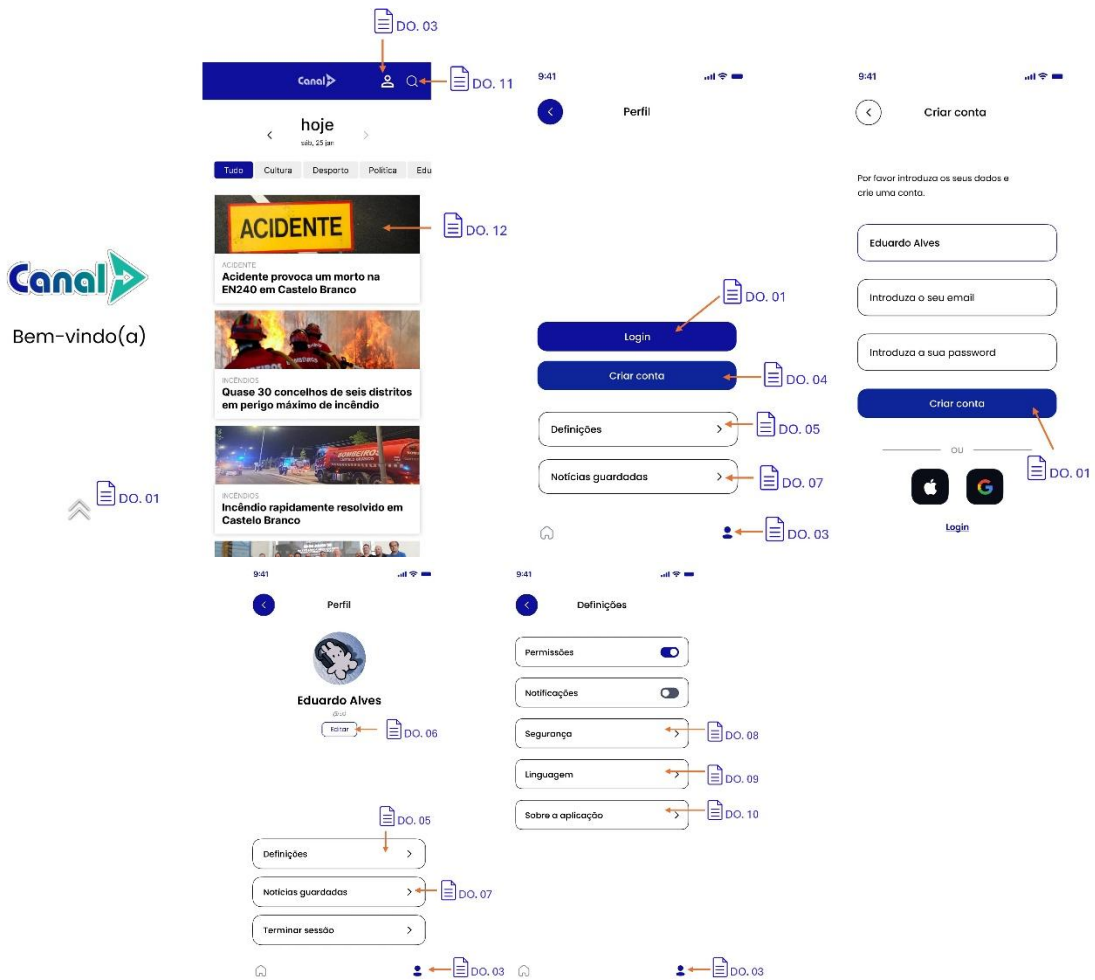


Figura 38 - Mapeamento dos Data Objects nos wireframes do Visitante

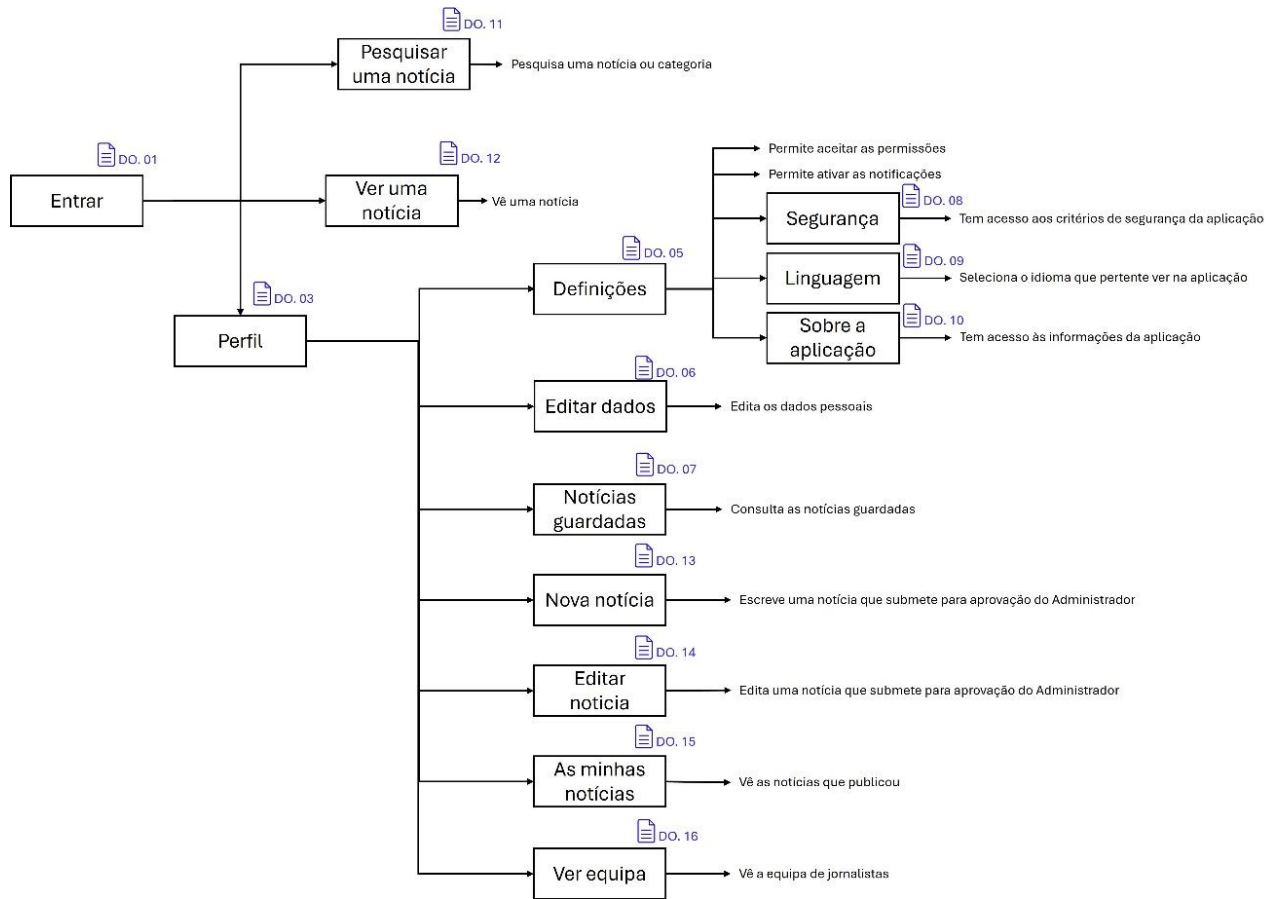


Figura 39 - Diagrama de Serviços do Jornalista



Figura 40 - Mapeamento dos Data Objects nos wireframes do Jornalista

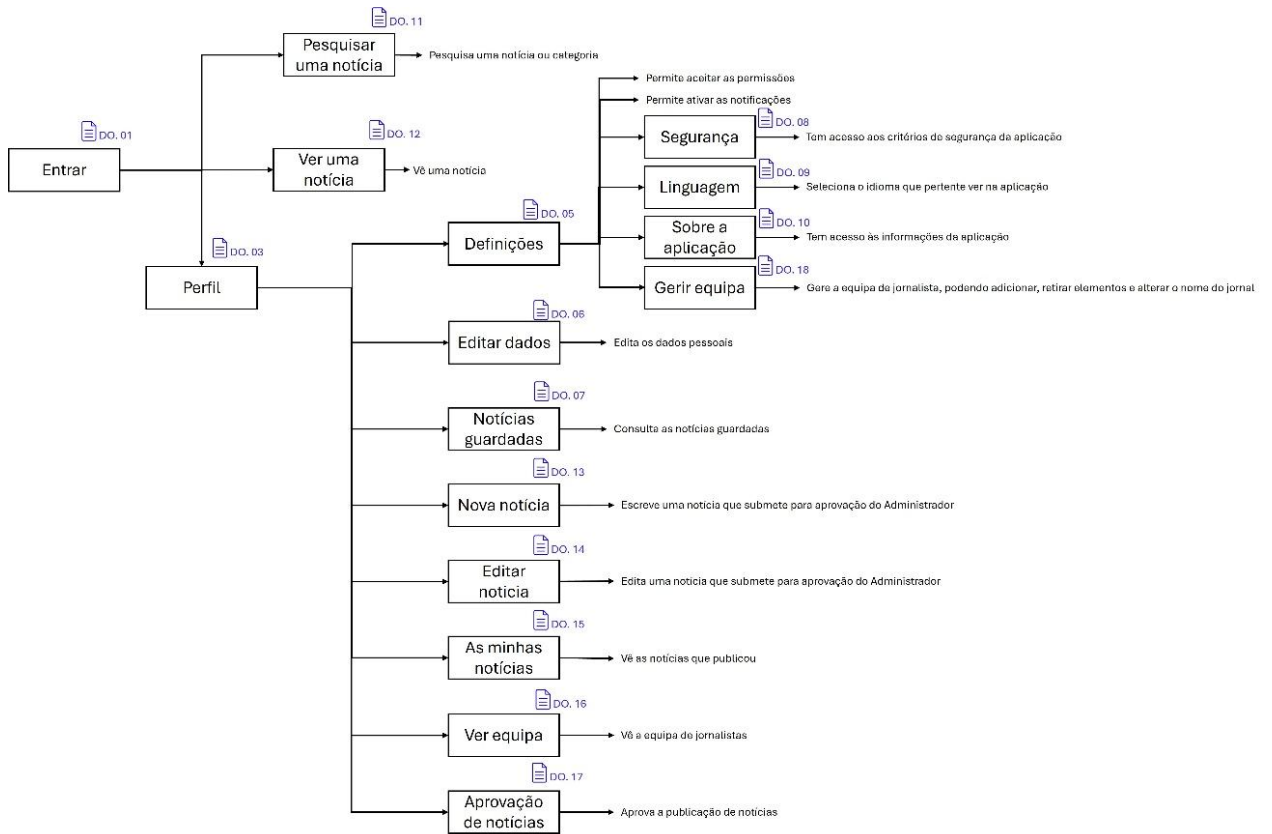


Figura 41 - Diagrama de Serviços do Administrador

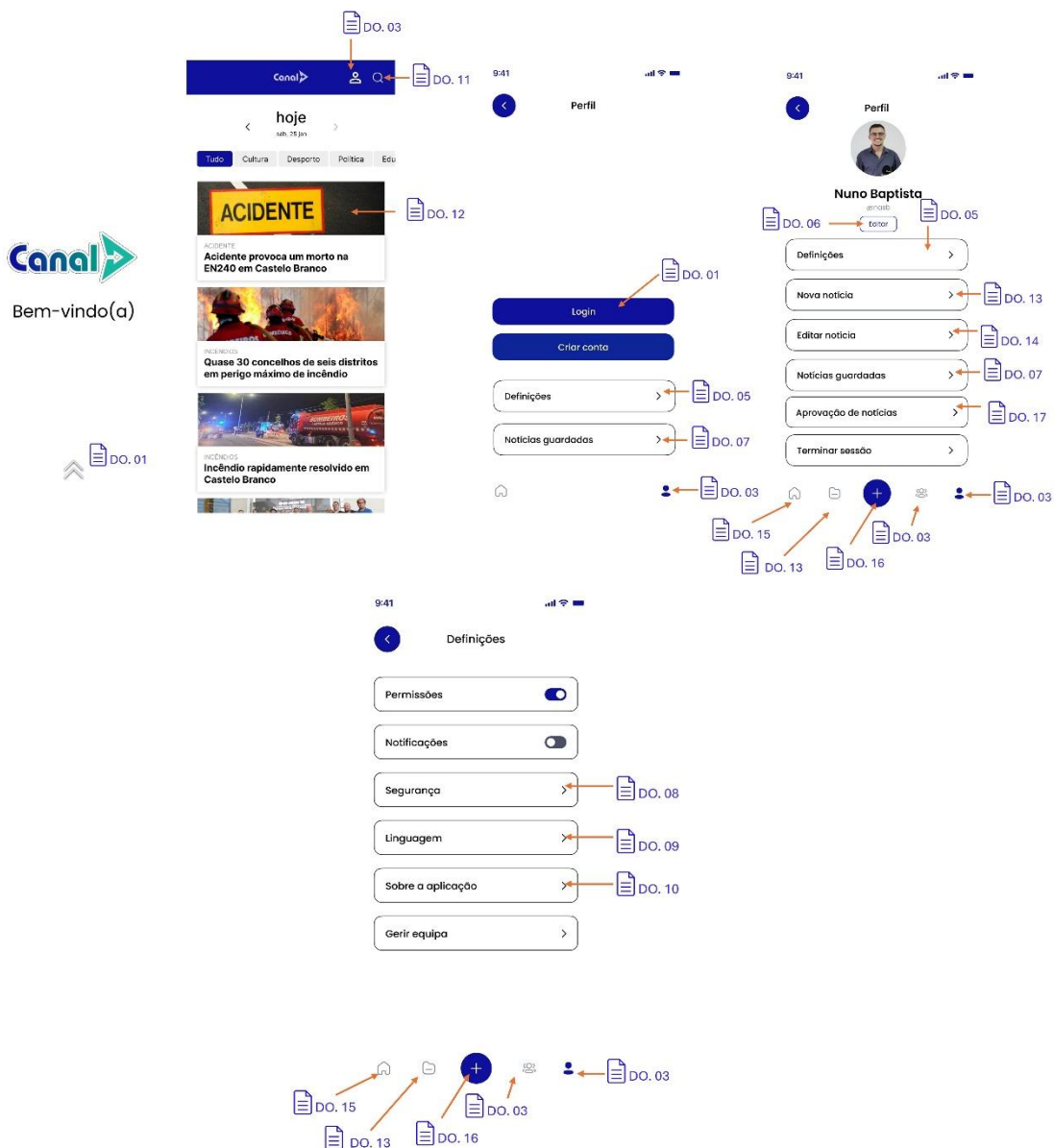


Figura 42 - Mapeamento dos Data Objects nos wireframes do Administrador

3.6.3. Cenários de Contexto

A criação de cenários permite visualizar as experiências dos utilizadores através de descrições detalhadas. Estes cenários representam experiências específicas de um utilizador ideal, que é definido utilizando técnicas de criação de personas. Isto possibilita priorizar ideias aplicadas a contextos de uso particulares [42]. Os cenários são estruturados com base em modelos narrativos ou de storytelling, que facilitam a comunicação de ideias e proporcionam experiências mais claras e envolventes para os utilizadores [40].

Tabela 5 - Cenários

ID	Nome
C1	O Visitante abre a aplicação e lê uma notícia
C2	O Visitante pesquisa uma notícia
C3	O Visitante tenta visualizar a sua conta, se não tiver conta, cria uma e faz login
C4	O Visitante edita os seus dados pessoais
C5	O Visitante consulta as notícias que guardou
C6	O Visitante consulta os termos de segurança, a linguagem e as informações da aplicação
C7	O Administrador inicia sessão e cria uma notícia
C8	O Administrador lê e edita uma notícia sua
C9	O Administrador aprova uma notícia escrita por um Jornalista
C10	O Administrador visualiza a equipa
C11	O Administrador gere a equipa
C12	O Jornalista inicia sessão e edita os seus dados pessoais
C13	O Jornalista escreve uma notícia
C14	O Jornalista lê e edita uma notícia sua
C15	O Jornalista consulta a equipa

C1 - O Visitante abre a aplicação e lê uma notícia:

O Visitante ao abrir a aplicação é lhe aparecido uma página de boas-vidas com o logotipo do jornal, como se pode ver na figura 43.



Figura 43 - Wireframe de Boas-Vindas à aplicação

Ao deslizar a página para cima é aparecido a página principal da aplicação, como se pode ver na figura 44.



Figura 44 - Wireframe da Página Principal

Nesta página principal pode selecionar uma das categorias sugeridas por baixo da data, como se pode ver na figura 44, aparecendo as notícias referidas dessa categoria (figura 45), também, tanto na página principal, como na página da categoria, pode selecionar uma notícia para ler, aparecendo uma pequena visualização da mesma (figura 46), podendo clicar em “Ver mais”, aparecendo a notícia por completo (figura 47).



Figura 45 - Wireframe do destaque Cultura



Figura 46- Wireframes da pequena visualização de uma Notícia



Figura 47 - Wireframes de visualização da Notícia completa

Nestas notícias podem, para além de imagens, ter também áudios e citações, como pode ser visto na figura 48.



Figura 48 - Wireframes de Notícias com áudio e citações

C2 - O Visitante pesquisa uma notícia

Em todas as páginas mostradas até agora é possível pesquisar uma notícia, clicando no ícone da lupa no canto superior direito, aparecendo uma página com todas as categorias e permitindo pesquisar por uma notícia (figura 49). Ao clicar numa das categorias o utilizador é levado para a sessão dessa categoria, por exemplo, ao clicar na categoria Cultura é levado para a sessão da Cultura (figura 45).

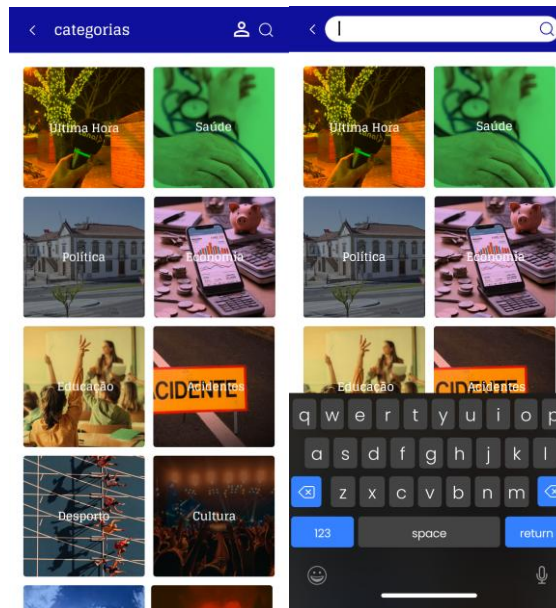


Figura 49 - Wireframes de pesquisa de Notícias

C3 - O Visitante tenta visualizar a sua conta, se não tiver conta, cria uma e faz login

Em todas as páginas apresentadas até agora aparece do lado esquerdo do ícone da lupa um ícone de uma pessoa, o Visitante ao clicar nesse ícone é levado para a sua conta (figura 50), caso tenha sessão iniciada, caso contrário aparece uma página semelhante à da conta, mas com dois botões que permitem criar uma conta e iniciar sessão (figura 51).

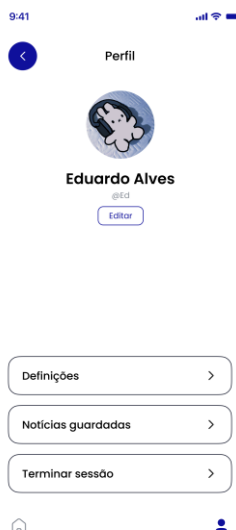


Figura 50- Wireframe de perfil com sessão iniciada



Figura 51- Wireframe de perfil sem sessão iniciada

Na figura 51 o Visitante ao clicar no botão “Login” é levado para a página Login (figura 52), ao clicar no botão “Criar conta” é levado para a página Criar Conta (figura 53).



Figura 52 - Wireframe da Página Login

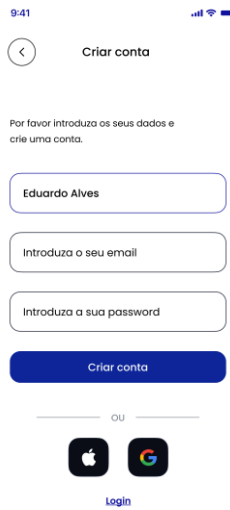


Figura 53 - Wireframe da Página Criar Conta

C4 - O Visitante edita os seus dados pessoais

Na figura 50 é permitindo o Visitante editar os seus dados pessoais (figura 54).



9:41

Editar Perfil [Salvar](#)

Name

Eduardo Alves

Email

eduardojonasalves@gmail.com

Username

@Ed

Number

+351 969593045

Figura 54 - Wireframe de Editar Dados Pessoais

C5 - O Visitante consulta as notícias que guardou

Ainda nas páginas do Perfil (figuras 50) é permitido ter acesso às notícias guardadas (figura 55), clicando no botão “Notícias guardadas”.



Figura 55 - Wireframe de Notícias Guardadas

C6 - O Visitante consulta os termos de segurança, a linguagem e as informações da aplicação

Nas páginas do Perfil (figuras 50) o Visitante tem acesso sempre às definições da aplicação (figura 56), onde pode permitir Permissões e Notificações.



Figura 56- Wireframe da Página Definições

Na figura 56 o utilizador ao clicar no botão “Segurança” é levado para uma página que contém todas as informações referentes à segurança (figura 57), ao

clique no botão “Linguagem” é levado para uma página onde pode selecionar a língua em que pretende que a aplicação lhe seja apresentada (figura 58) e ao clicar no botão “Sobre a aplicação” é levado para uma página onde lhe aparece as informações da aplicação (figura 59).

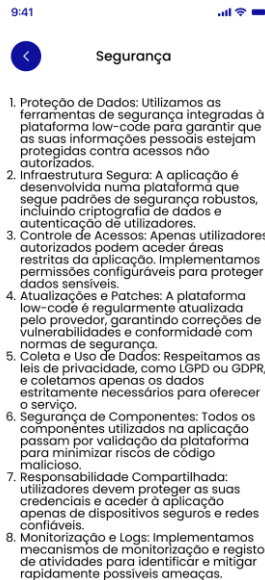


Figura 57 - Wireframe da Página Segurança



Figura 58 - Wireframe da Página de Seleção de Linguagem

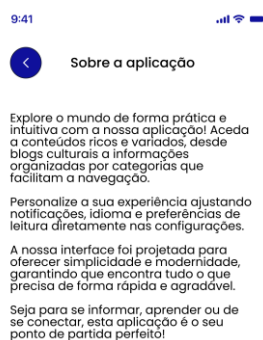


Figura 59 - Wireframe sobre a aplicação

C7- O Administrador inicia sessão e cria uma notícia



Figura 60- Início de sessão administrador

O Administrador inicia sessão colocando o email que utilizou ao criar a conta e a sua password (figura 60).



Figura 61 - Wireframe de perfil administrador



Figura 62 - Wireframe da Página de Definições de Administrador

Para criar uma notícia no Perfil (figura 61) o Administrador clica no botão “Nova notícia” ou no botão circular azul com um símbolo mais (+) no menu inferior, presente também na página das Definições (figura 62), que levam para uma página para criação de uma notícia (figura 63) onde pode acrescentar um título, texto, citações, bem como imagem e áudios e selecionar um destaque. Para adicionar o destaque o Administrador clica no botão “Selecione a Categoria” e leva-o para uma

página de procura onde aparecem todos os destaques (figura 64), selecionado o que pretende.

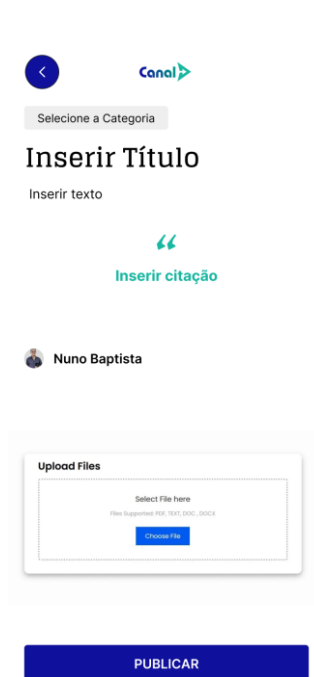


Figura 63 - Wireframe de Criar Notícia

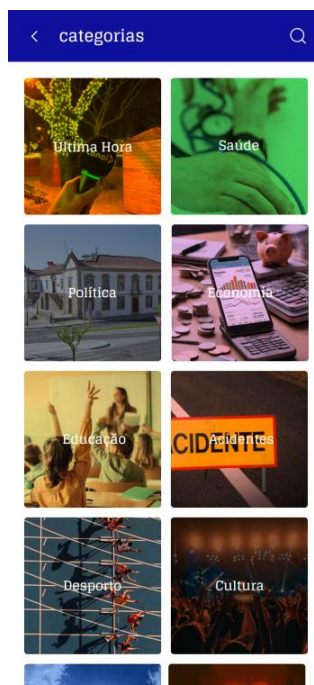


Figura 64 - Wireframe de Selecionar categoria da Notícia

Depois de clicar em “Publicar” a notícia é publicada, sem ter de ser verificada por ninguém visto ser o Administrador a publicar a notícia. É mostrado uma página de sucesso (figura 65).



Figura 65 - Wireframe de Publicação de Notícia

C8 - O Administrador lê e edita uma notícia sua

O Administrador para editar ou ler uma notícia sua, no Perfil (figura 61) clica no botão “Editar Notícia” ou no ícone da pasta do lado esquerdo no menu inserido na parte inferior da página que levam para uma página onde aparecem todas as notícias que este publicou (figura 66).

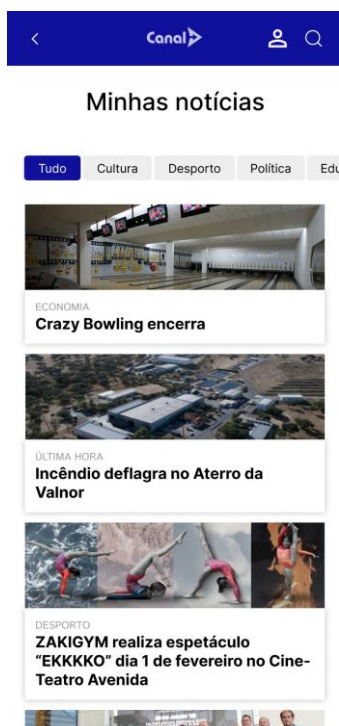


Figura 66- Wireframe da Página de Minhas Notícias

Nesta página o Administrador clica na notícia que pretende editar ou ler e aparece-lhe uma página com uma visualização da notícia e com dois botões: “Editar” e “Ler” (figura 67).

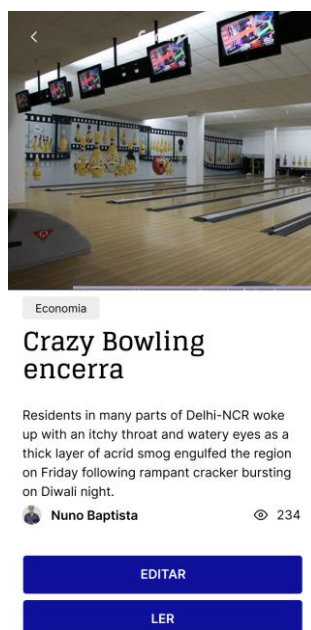


Figura 67 - Wireframe da Visualização da Notícia selecionada para leitura ou edição

O Administrador ao clicar no botão “Ler” lê a notícia como leitor normal (figuras 46). Ao clicar no botão “Editar” é levado para uma página que permite editar o título, o destaque, os conteúdos multimédia e acrescentar mais conteúdos clicando na figura da notícia, caso o pretenda fazer (figuras 68, 69 e 70). Após a edição o Administrador publica a notícia, ficando logo disponível para todos, visto não ter de aguardar por uma aprovação. Aparecendo novamente a página de sucesso (figura 79).

Para editar o destaque o Administrador clica no mesmo e leva-o para uma página de procura onde aparecem todos os destaques, selecionando o que pretende (figura 71).

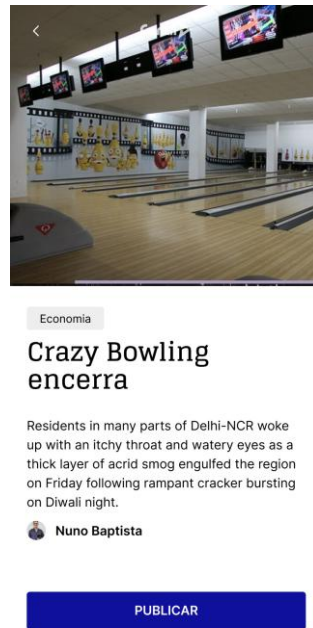


Figura 68- Wireframe de Edição de Notícia Administrador

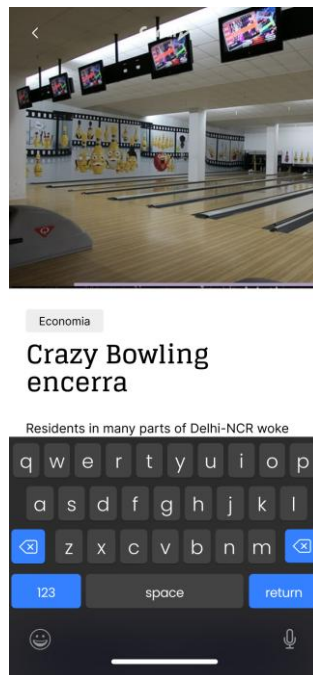


Figura 69- Wireframe de Edição do título e texto de uma Notícia Administrador

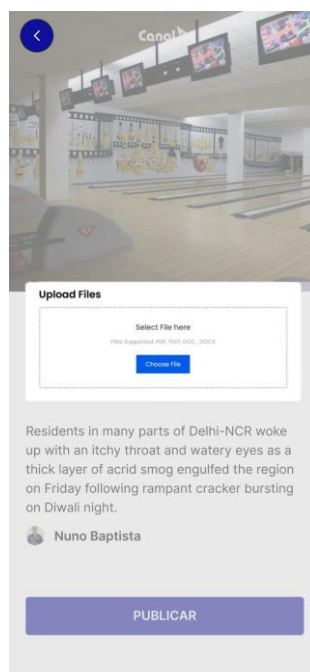


Figura 70- Wireframe de Edição e Adição de Conteúdos Multimédia Administrador



Figura 71- Wireframe da Edição da Categoria

C9 - O Administrador aprova uma notícia escrita por um Jornalista

O Administrador para aprovar notícias, na página do Perfil (figura 61) clica no botão “Aprovação de notícias” que o leva para uma página onde aparecem todas as notícias que se encontram por aprovar (figura 72).



Figura 72- Wireframe da Página Aprovar Notícias

Nesta página o Administrador clica na notícia que pretende aprovar, de seguida é-lhe aparecido uma visualização curta da notícia (figura 73), ao clicar em “Ver mais” é-lhe aparecido a notícia completa com dois botões: “Editar notícia” e “Aprovar publicação” (figura 74).



Figura 73- Wireframe de Pré-Visualização da Notícia para aprovação

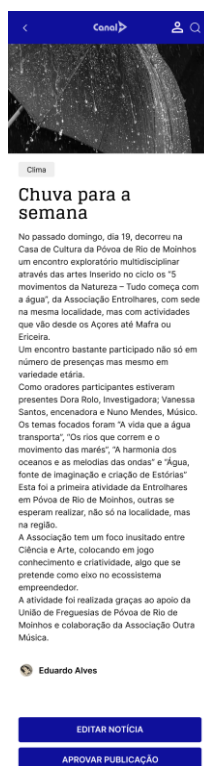


Figura 74- Wireframe da Notícia completa para aprovação

O Administrador ao clicar no botão “Editar notícia” vai para uma página que permite editar o título, o texto, o destaque, a imagem e acrescentar conteúdos multimédia da notícia, clicando na imagem da notícia, podendo depois destas alterações publicar a mesma para todos clicando no botão “Publicar” (figuras 75, 76 e 77). Aparecendo a página de sucesso (figura 79).

Para editar o destaque o Administrador clica no mesmo e leva-o para uma página de procura onde aparecem todos os destaques, selecionando o que pretende (figura 78).



Figura 75 - Wireframe Página de Edição da Notícia por aprovar

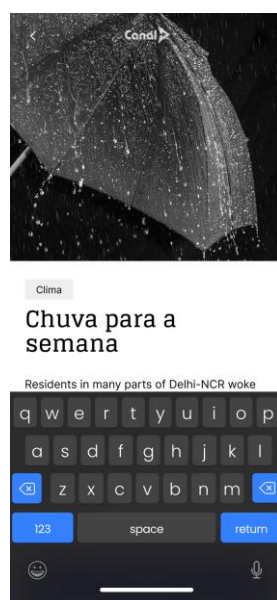


Figura 76- Wireframe da Edição do título e texto da Notícia por aprovar

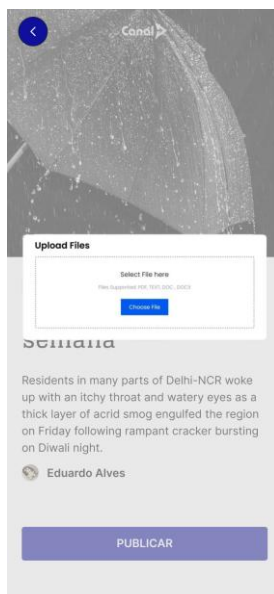


Figura 77- Wireframe de Edição e Adição Edição de conteúdos multimédia da Notícia por aprovar



Figura 78 - Wireframe da Seleção de Categoria



Figura 79 - Wireframe da Página de Sucesso

O Administrador na figura 74 ao optar por clicar no botão “Aprovar publicação” a notícia é publicada para todos e aparece novamente a página de sucesso (figura 79).

C10 - O Administrador visualiza a equipa

Ainda no menu do canto inferior da página do Perfil (figura 61) e da página das Definições (figura 62) o ícone com várias pessoas do lado direito leva para uma página onde o Administrador pode visualizar a equipa (figura 80).

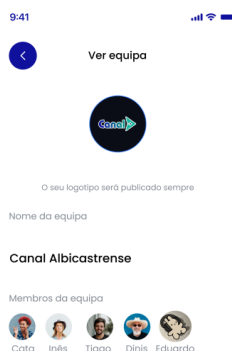


Figura 80 - Wireframe da Página de Visualizar Equipa

C11 - O Administrador gere a equipa

Na página de Definições (figura 62) o Administrador ao clicar no botão “Gerir equipa” pode editar o nome do jornal, remover elementos e adicionar elementos à equipa (figuras 81 e 82).

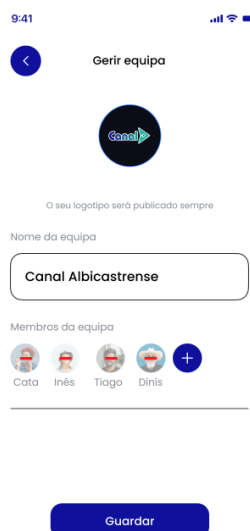


Figura 81 - Wireframe da Página Gerir Equipa

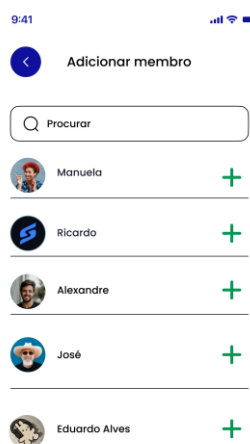


Figura 82 - Wireframe de Adicionar Membro

Para remover um elemento o Administrador basta clicar no menos colocado por cima da fotografia do elemento da equipa na página Gerir equipa (figura 81).

Para adicionar um elemento à equipa o Administrador basta clicar no ícone circular azul com um símbolo mais (+) que está ao lado das fotografias dos elementos da equipa na página Gerir equipa (figura 81). Depois de clicar no + é levado para a página Adicionar membro (figura 82).

Nesta aparecem todos os utilizadores registados na aplicação que não estão adicionados à equipa. O Administrador ao adicionar um utilizador este passa a pertencer à equipa podendo escrever notícias, designando-se como Jornalista. Todas as notícias que o Jornalista escreve têm de ser aprovadas pelo Administrador como podemos observar no cenário já explicado antes (C9).

C12 - O Jornalista edita os seus dados pessoais



Figura 83 - Wireframe do Perfil Jornalista

Para editar os seus dados pessoais, o Jornalista clica no botão “Editar” a azul por baixo do seu nome e *username*, sendo levado para a página Editar Perfil (figura 84).



Figura 84 - Wireframe de Editar Dados Pessoais Jornalista

Para o Jornalista escrever uma notícia, clica no botão “Nova notícia” exibido na página do Perfil (figura 83) ou no ícone circular azul com um mais (+) no menu inserido na parte inferior da página do Perfil (figura 83) e da página das Definições (figura 85). Após esta ação será levado para uma página de escrita de uma notícia (figura 86).



Figura 85- Wireframe da Página Definições

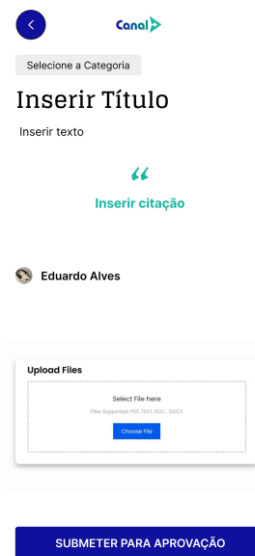


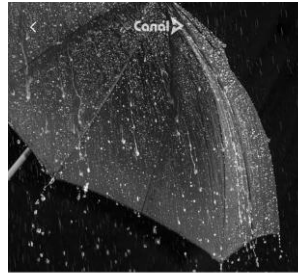
Figura 86 - Wireframe de Nova Notícia

Nesta página o Jornalista Seleciona a Categoria, clicando no botão “Selecione a Categoria” que o levará para uma página com todas as categorias (figura 87), onde o Jornalista seleciona a categoria que considera melhor.



Figura 87 - Wireframe de Selecionar Categoria

O Jornalista na página Criar Notícia (figura 86) pode ainda inserir um título, um texto, citações, imagens e áudios. Depois seleciona o botão “Submeter para aprovação”, sendo esta notícia enviada para o Administrador. Ao Jornalista aparece uma página com uma mensagem de sucesso (figura 88).



NOTÍCIA SUBMETIDA PARA APROVAÇÃO!

Clima

Chuva para a semana

Residents in many parts of Delhi-NCR woke up with an itchy throat and watery eyes as a thick layer of acrid smog engulfed the region on Friday following rampant cracker bursting on Diwali night. (Photo: PTI)

 Eduardo Alves

Figura 88 - Wireframe da página de Sucesso

C14 - O Jornalista lê e edita uma notícia sua

O Jornalista para editar e/ou ler as notícias publicadas por ele, pode clicar no botão Editar notícias exibido na página do Perfil (figura 83) ou no ícone da pasta no lado esquerdo do menu inferior exibido nas páginas do Perfil (figura 83) e Definições (figura 85).

O Jornalista ao clicar nesses botões será levado para uma página que contém todas as notícias que o mesmo publicou e estão disponíveis para todos (figura 89). Nesta página o Jornalista seleciona a notícia que pretender ler e/ou editar. Levando-o para uma página de visualização da notícia com dois botões: “Editar” e “Ver” (figura 90).



Figura 89 - Wireframe da Página Minhas Notícias



Figura 90 - Wireframe de Pré-Visualizar Notícia para edição ou leitura

Na figura 90 o Jornalista ao selecionar o botão “Ler” é levado para a página de visualização do Leitor normal, tendo o botão “Ler mais” (figura 46) que ao clicar no mesmo leva para uma visualização total da notícia (figura 47).

Se o Jornalista optar por selecionar o botão “Editar” será levado para uma página que permite editar o título, o texto, o destaque, a imagem e acrescentar conteúdos multimédia da notícia, clicando na imagem da notícia, podendo depois

destas alterações submeter para aprovação (figuras 91, 92 e 93). Aparecendo a página de sucesso (figura 95).

Para editar o destaque o Administrador clica no mesmo e leva-o para uma página de procura onde aparecem todos os destaques, selecionando o que pretende (figura 94).



Figura 91 - Wireframe de Edição da Notícia Jornalista



Figura 92 - Wireframe de Edição de título e texto de uma Notícia Jornalista

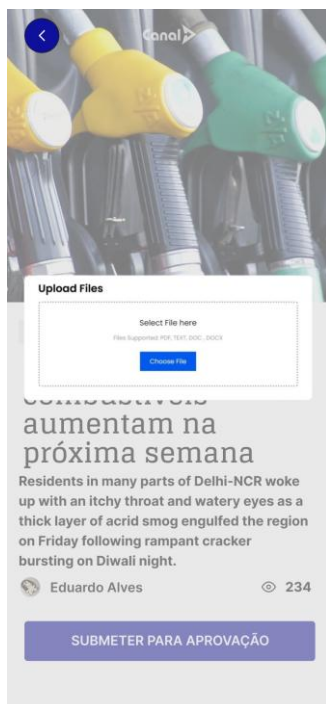


Figura 93 - Wireframe de Edição ou Adição de Conteúdos Multimédia Jornalista



Figura 94 - Wireframe de Seleção de Categoria



Figura 95 – Wireframe da Página de Sucesso

C15 - O Jornalista consulta a equipa

No menu inferior exibido nas páginas do Perfil (figura 83) e Definições (figura 85) no canto inferior o ícone com várias pessoas do lado direito leva para uma página onde o Jornalista pode visualizar a equipa (figura 96).

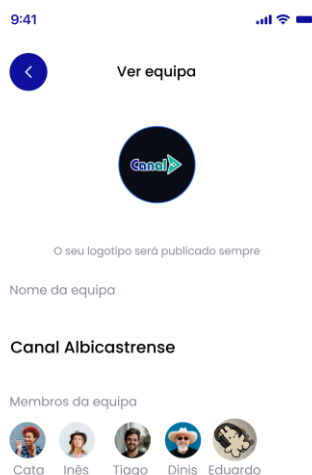


Figura 96 - Wireframe da Página de Visualização da Equipa

3.6.4. Biblioteca de Estilos *UI*

A biblioteca de estilos *UI* aplicada à proposta de solução consiste na definição de estilos que correspondem aos objetivos do projeto. Considerando que a proposta de solução é utilizada em *smartphones* de nova geração, a aplicação nativa deve ser desenvolvida de acordo com o sistema operacional do dispositivo no qual foi testado o protótipo de alta-fidelidade. O *iPhone 13* foi o dispositivo escolhido para testar a interface da aplicação com os utilizadores. Este modelo apresenta especificações avançadas, incluindo um ecrã de 6,1 polegadas, que oferece uma experiência de visualização ideal para o desenvolvimento e avaliação da interface. Além disso, o *iPhone 13*, sendo um *smartphone* de nova geração, garante compatibilidade com as mais recentes tecnologias e padrões de *design*.

Layout

Foram definidas três diferentes áreas da *interface*, com diferentes cores (figura 97): verde, amarelo e vermelho.



Figura 97 - As diferentes áreas da *interface*

A área a verde corresponde à área de possível alcance, onde colocámos os componentes clicáveis. A área a amarelo corresponde à área que é vista imediatamente, o centro, colocando as notícias e os destaques do dia nesta área, por exemplo, de modo a chamar logo à atenção. Todos os componentes que não são introduzidos nestas duas áreas (verde e amarelo), consequentemente pertencentes à área a vermelho, devem ser caracterizados por um tipo de interação específica, como por exemplo a interação slide pode ser aplicada tendo em conta que pode ser realizada em qualquer parte da *interface*. Seguidamente apresentamos na figura 98 a sobreposição entre quatro diferentes *interfaces* do protótipo final e o *layout* base que delimita as diferentes áreas.

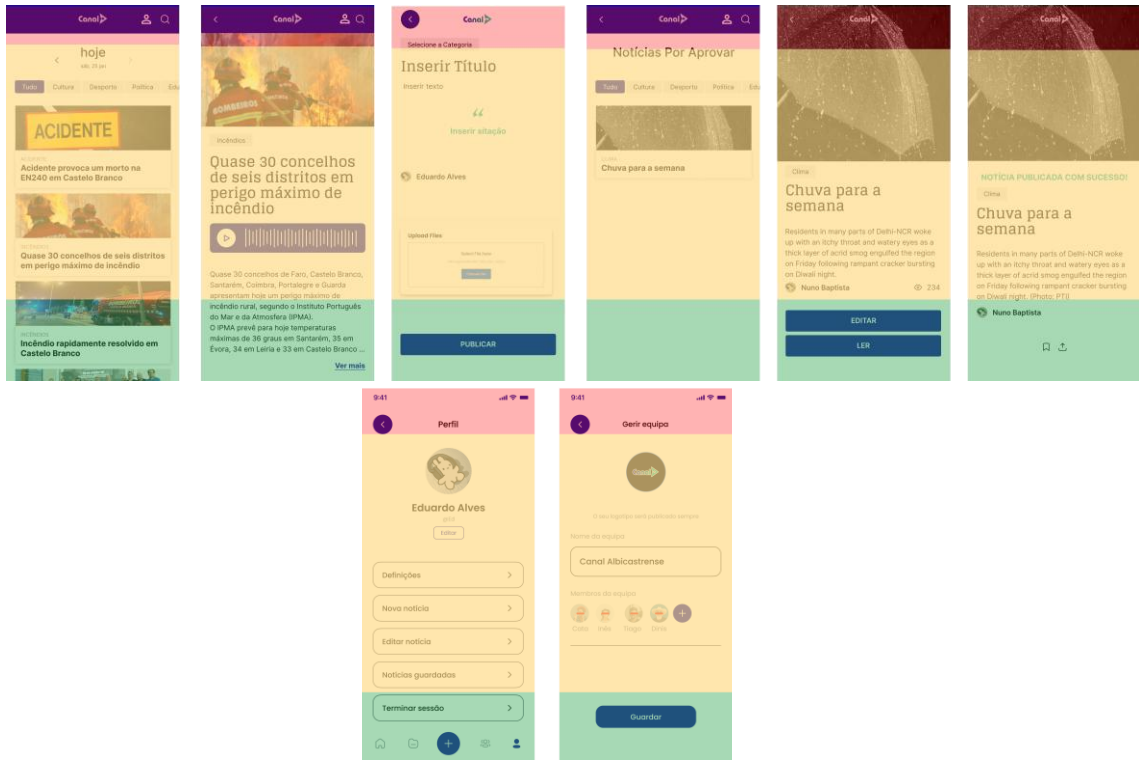


Figura 98 - Comparação entre as áreas de possível alcance com 8 interfaces

Estilo Tipográfico

Para as sessões referentes às notícias escolhemos as seguintes tipografias:

- **Inter**

(fonte: <https://fonts.google.com/specimen/Inter?query=Inter>);



Figura 99 - Tipografia Inter



Figura 100 - Exemplo de wireframes com o tipo de letra *Inter*

- *Glegoo*

(fonte: <https://fonts.google.com/specimen/Glegoo?query=Glegoo>);

Penultimate
The spirit is willing but the flesh is weak
SCHADENFREUDE
3964 Elm Street and 1370 Rt. 21
The left hand does not know what the right hand is doing.

Figura 101 - Tipografia *Glegoo*



Figura 102 - Exemplo de wireframes com o tipo de letra Glegoo

-Istok Web

(fonte: <https://fonts.google.com/specimen/Istok+Web?query=Istok+Web+>)

Penultimate
 The spirit is willing but the flesh is weak
SCHADENFREUDE
 3964 Elm Street and 1370 Rt. 21
 The left hand does not know what the right hand is doing.

Figura 103 - Tipografia Istok Web



Figura 104 - Exemplo de *wireframes* com o tipo de letra *Istok Web*

Optou-se por estas tipografias visto serem utilizadas para escrita de notícias.

Para a sessão dos menus do Perfil, Definições e Gestão da Equipa optou-se pela topografia *Poppins* (fonte: <https://fonts.google.com/specimen/Poppins?query=Poppins>), sendo aplicada através de vários estilos.

Optou-se por este tipo de topografia devido ser uma fonte *sans-serif* geométrica moderna e versátil, muito popular em *designs* digitais e impressos. A *Poppins* tem linhas simples e formas geométricas equilibradas, o que dá uma estética limpa e contemporânea, ideal para projetos modernos. A fonte adapta-se bem tanto a títulos grandes quanto a pequenos textos, mantendo legibilidade em qualquer escala. As suas letras são baseadas em formas geométricas perfeitas, proporcionando uniformidade e harmonia visual no *design*, contribui para um ritmo de leitura mais eficaz.

ABCDEFGHIJKLM
 NOPQRSTUVWXYZ
 abcdefghijklm
 nopqrstuvwxyz
 1234567890
 \$?&%@!#*()=

Figura 105 - Tipografia *Poppins*

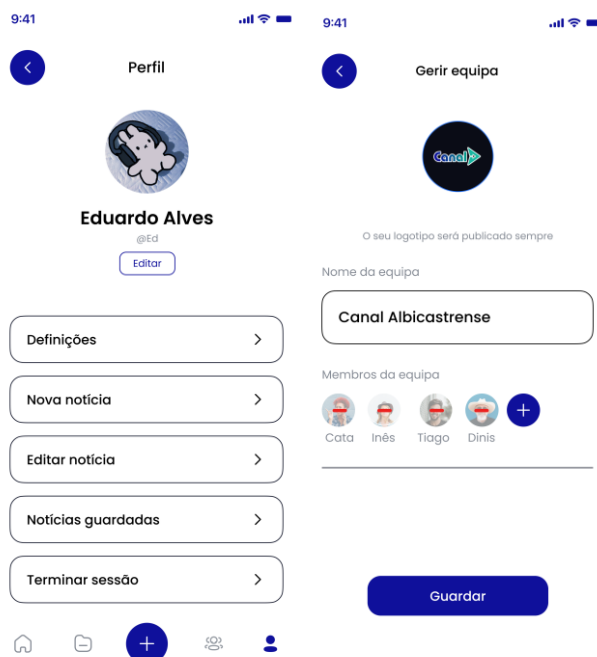


Figura 106 - Exemplo de *wireframes* com o tipo de letra *Poppins*

Cores

As cores introduzidas na interface são divididas em dois grupos: o grupo de cores que define a consistência da interface e o grupo de cores que definem as cores dos destaques das notícias. As cores que definem a imagem visual da interface e a sua consistência são as cores que definem a base de todas as interfaces e imagem de marca da aplicação, seguidamente apresentadas na figura 107.

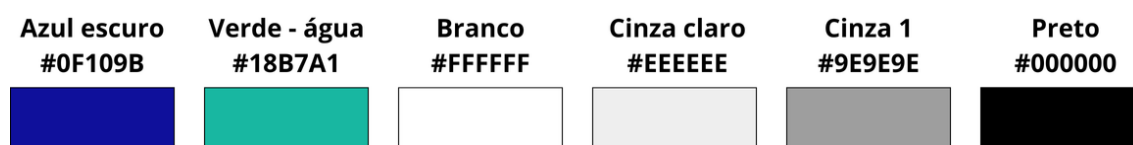


Figura 107 - Cores (1)

Optou-se por escolher o Azul escuro (0F109B) e o Verde-água (18B7A1) devido às cores do logotipo do jornal *Canal Albicastrense* (figura 108).



Figura 108 - Logotipo do *Canal Albicastrense*

As cores que definem os destaques (figura 109) na página da Pesquisa de notícias (figura 110) foram baseadas nas cores que o jornal utiliza atualmente para

o respetivo destaque, por exemplo em notícias de educação a utilização de um amarelo claro (figura 111).

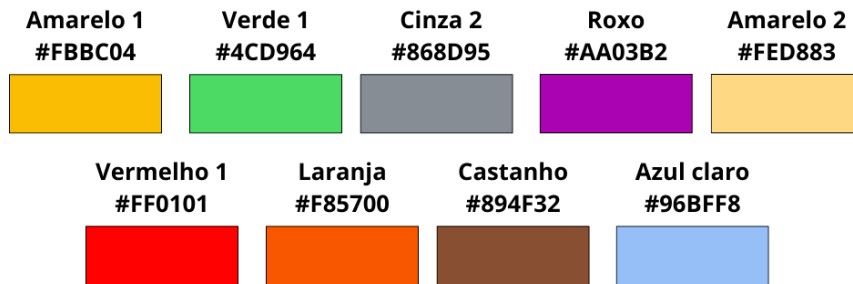


Figura 109 - Cores (2)

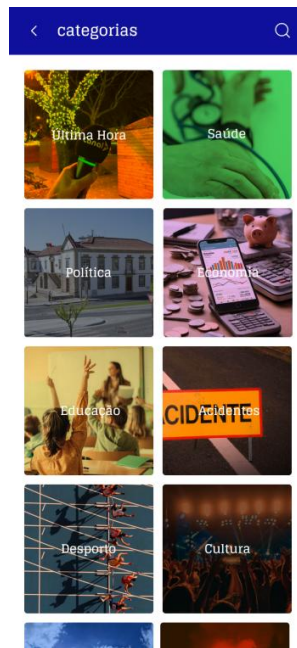


Figura 110 - Wireframe da Pesquisa de Notícias



Figura 111 - Imagem utilizada numa notícia real do Canal Albicastrense referente a educação

Para além destes dois grupos de cores ainda se recorreu a duas cores: Verde (009951) e Vermelho (EC221F) (figura 112). Estas cores foram utilizadas nos ícones de adicionar e remover elemento da equipa (figura 113).

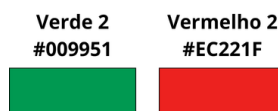


Figura 112 - Cores (3)

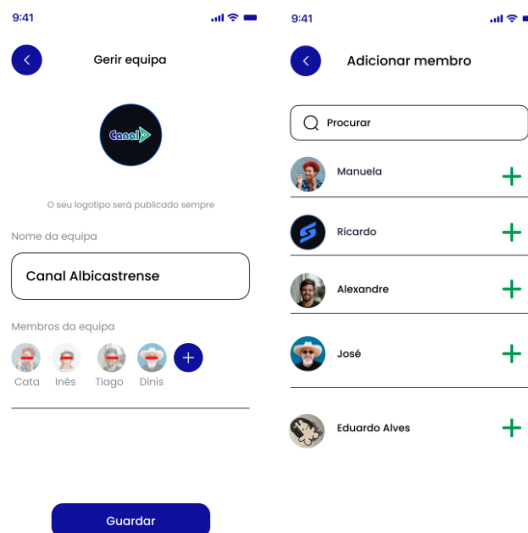


Figura 113 - Wireframes de gestão da equipa

Iconografia

Os ícones são, maioritariamente, familiares a todos de forma a não causar dúvidas no utilizador. A figura 114 representa a biblioteca de ícones aplicada à interface da proposta de solução.

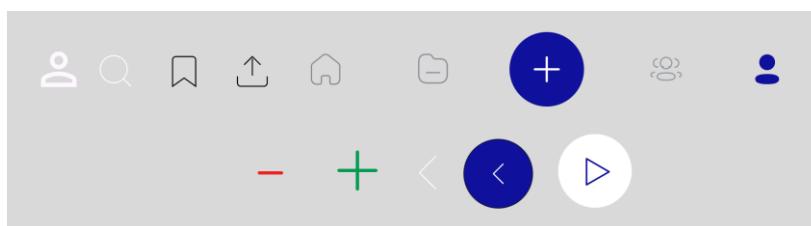


Figura 114 - Iconografia utilizada

Tab Bars

Aplicámos a componente *tab bar* de acesso às *interfaces* notícias (*home*), notícias publicas pelo utilizador (no caso do Jornalista e do Administrador), criação de notícias (no caso do Jornalista e do Administrador), equipa (no caso do Jornalista e do Administrador) e área pessoal. Destacámos o botão de criação de notícias (no caso do Jornalista e do Administrador), pois a partir deste o utilizador consegue aceder mais rápido à criação de uma notícia. Na figura 115 é possível visualizar os dois tipos de *tab bars* criados. Diferindo apenas para o Visitante.

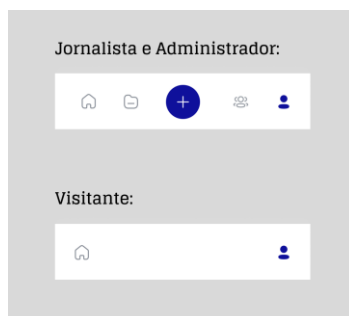


Figura 115 - Tipos de *Tab Bars* utilizados

Botões

Os botões são um elemento essencial na construção de uma *interface* de interação. A figura 116 apresenta os botões de ações principais (exemplo: Publicar uma notícia) são representados com a cor Azul escuro, os botões de ações secundárias são representados com a cor Branco.

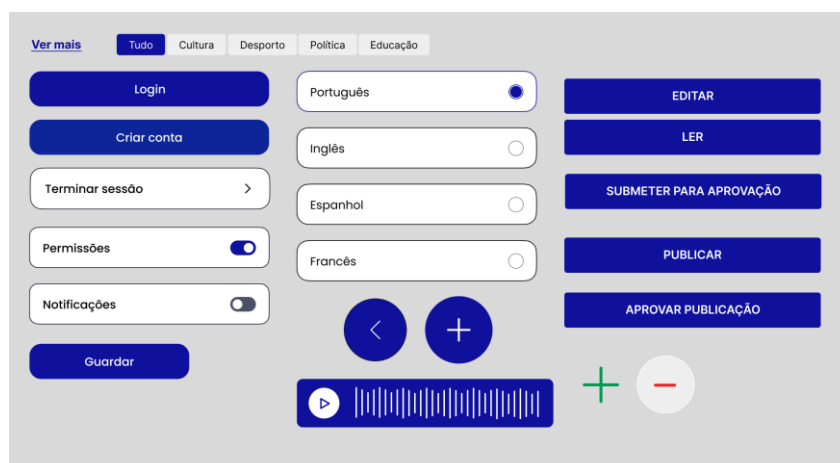


Figura 116 - Botões

Tabs

A componente *tabs* é aplicada nas *interfaces* que apresentam vários conteúdos, por exemplo, na interface das notícias, o utilizador pode o seu perfil e pesquisar uma notícia. A figura 117 apresenta os tipos de *tabs* aplicados à interface da proposta de solução.

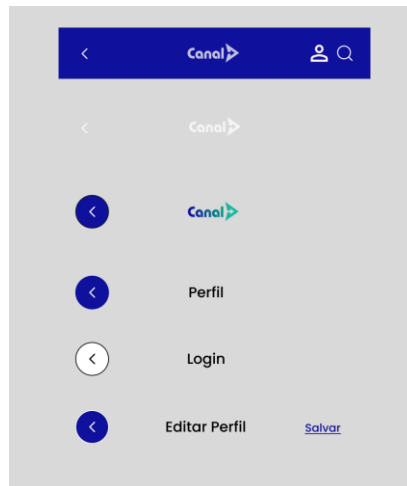


Figura 117 - Tabs utilizadas

Protótipo Clicável

Nesta etapa, foi elaborado um protótipo no *Figma*, permitindo-nos visualizar e testar a interface e funcionalidades planeadas.

As *interfaces*, assim como o protótipo clicável, foram desenvolvidas utilizando o *Figma*. O protótipo clicável teve como objetivo visualizar e testar a *interface* e funcionalidades planeadas e pode ser consultado através dos seguintes *links*:

Visitante:

<https://www.figma.com/proto/ME5OM6rvESuc14XavIRpik/Untitled?node-id=63-1676&p=f&t=tHzyNk946Pt8QjnJ-0&scaling=min-zoom&content-scaling=fixed&page-id=60%3A1457&starting-point-node-id=63%3A1676&show-proto-sidebar=1>

Administrador:

<https://www.figma.com/proto/ME5OM6rvESuc14XavIRpik/Untitled?node-id=63-4072&p=f&t=tHzyNk946Pt8QjnJ-0&scaling=min-zoom&content-scaling=fixed&page-id=38%3A719&starting-point-node-id=63%3A5608&show-proto-sidebar=1>

Jornalista:

<https://www.figma.com/proto/ME5OM6rvESuc14XavIRpik/Untitled?node-id=75-2416&p=f&t=tHzyNk946Pt8QjnJ-0&scaling=min-zoom&content-scaling=fixed&page-id=0%3A1&starting-point-node-id=75%3A4202>

Este protótipo clicável representa todos os cenários da Tabela 5.

4. Conclusões e Trabalho Futuro

Este projeto permitiu uma exploração detalhada das plataformas Low-Code, evidenciando as suas capacidades técnicas e o impacto desta abordagem no desenvolvimento de software. A análise dos artigos e referências selecionadas proporcionou uma compreensão aprofundada das principais tendências, desafios e oportunidades associadas a esta tecnologia, o questionário feito às empresas demonstrou-se eficiente nas conclusões relacionadas com a utilização de plataformas Low-Code.

As realizações de diagramas da área de conhecimento de Arquitetura da Informação contribuíram para a estruturação dos conteúdos, requisitos do Projeto, definindo um diagrama de serviços que representa a estrutura e comportamento do sistema.

No que diz respeito ao design visual, na qual foi definida a biblioteca de estilos, escolha de letra e cores, esta partiu da cor do Logotipo já existente, o que tal como iremos referir no trabalho futuro será alterada de forma a melhorar a experiência de utilização da aplicação.

Ao longo do estudo, constatou-se que o Low-Code tem um papel fundamental na inovação colaborativa e acessibilidade, permitindo que profissionais com diferentes níveis de experiência possam contribuir para o desenvolvimento de aplicações. Esta realidade altera o perfil dos developers, exigindo novas competências, destacando-se (i) a capacidade de integrar ferramentas, (ii) compreender modelos de negócio e atuar de forma mais estratégica na construção de soluções.

Além disso, foi criado um cenário através do desenvolvimento de um protótipo funcional usando o Figma.

Do ponto de vista empresarial, os resultados mostram que os benefícios do Low-Code são evidentes, incluindo a redução do tempo de desenvolvimento e a otimização de recursos. Ainda assim, a sua adoção exige um planeamento adequado para garantir que a escolha da plataforma corresponda às necessidades específicas do projeto e da organização.

Trabalho Futuro

Com base nos resultados alcançados, algumas direções para trabalhos futuros incluem:

Escolher uma metodologia Low-Code - Como próximo passo, será realizada a seleção de uma plataforma Low-Code específica para conduzir os experimentos e validar a aplicabilidade desta abordagem no contexto estudado.

Desenvolvimento e Teste do Protótipo – Criar um protótipo utilizando a metodologia Low-Code escolhida no ponto anterior e explorar cenários mais complexos, avaliando o potencial do Low-Code em aplicações.

Análise da Integração com Outras Tecnologias – Investigar como o Low-Code pode ser combinado com Inteligência Artificial e Automatização, ampliando as capacidades e tornando-o ainda mais eficiente.

5. Referências

[1] Instituto de Gestão de Projetos (2017) A Guide to the Project Management Body of Knowledge (Guia PMBOK), 6ª ed.; Project Management Institute: Newtown Square, PA, EUA.

[2] Moran, A. (2015) Gestão Ágil: Estratégia, Implementação, Organização e Pessoas. Springer International Publishing: Zurique, Suíça.

[3] TrustRadius (n.d.) Low-Code: Pros & Cons. Disponível em: <https://solutions.trustradius.com/buyer-blog/low-code-pros-cons/> [Acedido a 27 de setembro de 2024].

[4] Quixy (n.d.) Pros and Cons of Low-Code Development. Disponível em: <https://quixy.com/blog/pros-and-cons-of-low-code-development/> [Acedido a 27 de setembro de 2024].

[5] AppGyver (n.d.) Disponível em: <https://www.appgyver.com/> [Acedido a 11 de outubro de 2024].

[6] Amazon Web Services (n.d.) Introducing Amazon Honeycode: Build Web & Mobile Apps Without Writing Code. Disponível em: <https://aws.amazon.com/pt/blogs/aws/introducing-amazon-honeycode-build-web-mobile-apps-without-writing-code/> [Acedido a 11 de outubro de 2024].

[7] AppSheet (n.d.) Disponível em: <https://about.appsheet.com/home/> [Acedido a 11 de outubro de 2024].

[8] Umso (n.d.) Disponível em: <https://www.umso.com/> [Acedido a 11 de outubro de 2024].

[9] Google Cloud (n.d.) Looker. Disponível em: <https://cloud.google.com/looker> [Acedido a 11 de outubro de 2024].

[10] Mendix (n.d.) Disponível em: <https://www.mendix.com/> [Acedido a 11 de outubro de 2024].

[11] Microsoft (n.d.) Power Apps. Disponível em: <https://www.microsoft.com/pt-pt/power-platform/products/power-apps/?market=pt> [Acedido a 11 de outubro de 2024].

[12] OutSystems (n.d.) Disponível em: <https://www.outsystems.com/> [Acedido a 11 de outubro de 2024].

- [13] Rintagi (n.d.) Disponível em: <https://www.rintagi.com/> [Acedido a 11 de outubro de 2024].
- [14] Salesforce (n.d.) Disponível em: <https://www.salesforce.com/eu/?ir=1> [Acedido a 11 de outubro de 2024].
- [15] ServiceNow (n.d.) Disponível em: <https://www.servicenow.com/> [Acedido a 11 de outubro de 2024].
- [16] Sisense (n.d.) Disponível em: <https://www.sisense.com/> [Acedido a 11 de outubro de 2024].
- [17] Skyve (n.d.) Disponível em: <https://skyve.org/> [Acedido a 11 de outubro de 2024].
- [18] VisionX (n.d.) Disponível em: <https://visionx.sibvisions.com/> [Acedido a 11 de outubro de 2024].
- [19] Editor X (n.d.) Disponível em: <https://pt.editorx.com/> [Acedido a 11 de outubro de 2024].
- [20] Yellowfin BI (n.d.) Disponível em: <https://www.yellowfinbi.com/> [Acedido a 11 de outubro de 2024].
- [21] Zoho (n.d.) Creator. Disponível em: <https://www.zoho.com/creator/> [Acedido a 11 de outubro de 2024].
- [22] TechTarget (n.d.) What is Low-Code? A Guide to Enterprise Low-Code App Development. Disponível em: <https://www.techtarget.com/searchsoftwarequality/What-is-low-code-A-guide-to-enterprise-low-code-app-development> [Acedido a 26 de outubro de 2024].
- [23] NoCode.Tech (n.d.) What is No-Code? Disponível em: <https://www.nocode.tech/article/what-is-no-code>. [Acedido a 26 de outubro de 2024].
- [24] Seidor (n.d.) Low-Code e No-Code. Disponível em: <https://www.seidor.com/es-es/customer-experience/desarrollo-web-app/low-code>. [Acedido a 26 de outubro de 2024].
- [25] SAP (n.d.) What is Low-Code/No-Code? Disponível em: <https://www.sap.com/products/technology-platform/build/what-is-low-code-no-code.html>. [Acedido a 27 de outubro de 2024].
- [26] Pipefy (n.d.) Low-Code vs. No-Code. Disponível em: <https://www.pipefy.com/blog/low-code-vs-no-code/>. [Acedido a 28 de outubro de 2024].
- [27] Zeev (n.d.) Low-Code e No-Code: Quais São as Principais Diferenças e Vantagens? Disponível em: <https://zeev.it/blog/low-code-e-no-code-quais-sao-as-principais-diferencas-e-vantagens/>. [Acedido a 15 de novembro de 2024].

[28] Cronapp (n.d.) Low-Code vs. No-Code. Disponível em: <https://blog.cronapp.io/low-code-vs-no-code/> [Acedido a 15 de novembro de 2024].

[29] TechTarget (n.d.) Assess Low-Code Platform Pros and Cons. Disponível em: <https://www.techtarget.com/searchsoftwarequality/tip/Assess-low-code-platform-pros-and-cons> [Acedido a 23 de novembro de 2024].

[30] Forbes Tech Council (2023) Low-Code and No-Code: 19 Potential Downsides to Be Aware Of. Disponível em: <https://www.forbes.com/councils/forbestechcouncil/2023/07/28/low-code-and-no-code-19-potential-downsides-to-be-aware-of/> [Acedido a 23 de novembro de 2024].

[31] PRISMA (n.d.) Preferred Reporting Items for Systematic Reviews and Meta-Analyses. Disponível em: <https://www.prisma-statement.org/> [Acedido a 23 de novembro de 2024].

[32] Forrester Consulting (n.d.) The Total Economic Impact™ Of Pega Platform For Low Code.

[33] (n.d.) Low-Code Adoption as a Driver of Digital Transformation.

[34] (n.d.) Modelling in Low-Code Development: A Multi-Vocal Systematic Review.

[35] (n.d.) Low-Code Approach for Business Analytics.

[36] (n.d.) Characteristics and Challenges of Low-Code Development: The Practitioners' Perspective.

[37] (n.d.) Low-Code Programming Models.

[38] (n.d.) Technology Trends for 2024.

[39] Frank E. Ritter, Gordon D. Boxter, Elizabeth F. Churchill (2014) User-Centered Systems Design: A Brief History. (n.d.), citado por MÁRTIRES, I. 2018.

[40] Cooper, A., Reimann, R., & Cronin, D. (2007). About Face 3: The essentials of interaction design. Information Visualization (Vol. 3), citado por MÁRTIRES, I. 2018.

[41] Hartson, R., & Pyla, P. S. (2012). The UX Book: Process and Guidelines for Ensuring a Quality User Experience, citado por MÁRTIRES, I. 2018.

[42] Kalbach, J. (2016). Mapping Experiences. O'Reilly, citado por MÁRTIRES, I. 2018.