

Low-Code

Abstract

Este artigo científico investiga o crescente fenômeno do desenvolvimento de software através da abordagem *Low-Code*. Discutimos a relevância crescente do *Low-Code* no cenário tecnológico atual, enfatizando o seu papel na simplificação do processo de desenvolvimento e na redução da lacuna entre os desenvolvedores e os utilizadores finais. Além disso, exploramos as implicações sociais do *Low-Code*, destacando o seu potencial para promover a inclusão digital e capacitar comunidades menos privilegiadas ao fornecer ferramentas acessíveis para criar aplicações personalizadas. Analisamos as tecnologias comercialmente disponíveis no campo do *Low-Code*, destacando a sua variedade e flexibilidade para atender às diversas necessidades de desenvolvimento. Também examinamos como a inteligência artificial está a ser implementada para com plataformas *Low-Code*, melhorando a automação e a eficiência no processo de desenvolvimento. Por fim, abordamos a interface intuitiva de *Drag&Drop*, uma característica fundamental do *Low-Code* que simplifica a criação de aplicações, permitindo que até mesmo utilizadores sem experiência técnica possam desenvolver soluções funcionais. Este estudo destaca o potencial transformador do *Low-Code* na democratização do desenvolvimento de software, apontando para uma era de inovação mais acessível e inclusiva.

Keywords: Low-Code, No-Code, Drag&Drop, Desenvolvimento Eficiente, Inteligência Artificial, ChatBot, OutSystems, Zoho Creator

1 Introdução

O *Low-Code*, pelo o que o próprio nome indica, é uma ferramenta de desenvolvimento de *software* que permite a criação de aplicações com pouco ou nenhum código, como tal implica que o desenvolvimento das aplicações seja mais rápida, o que pode causar uma redução da equipa de desenvolvedores, que subsequentemente reduz o custo de desenvolvimento. O *Low-code* permite ainda, que não seja necessário um curso superior para trabalhar com esta ferramenta, uma vez que não é preciso saber programar para usar ferramentas relacionadas ao *Low-Code*.

2 O que é Low-Code



O *low-code* é uma abordagem revolucionária no desenvolvimento de software projetada para simplificar e acelerar o processo de criação de aplicações, consistindo na utilização de elementos pré-definidos que podem ser arrastados para a tela (*Drag&Drop*) e interligados entre si, efetivamente permitindo que esses elementos cooperem de forma lógica para que se chegue ao pretendido. Isto permite que desenvolvedores e até mesmo utilizadores sem experiência em programação projetem e construam soluções, neste mesmo contexto também existe o conceito de *nocode*, que é uma versão mais extrema do conceito de *low-code*.^[1]

Para tal há a necessidade de obtenção de formação na própria plataforma, mas tal formação é factualmente muito mais rápida e intuitiva do que a formação necessária para o domínio de qualquer linguagem de programação convencional - Domínio esse visado ao mesmo nível que seria comparativamente necessário para o desenvolvimento da mesma qualidade e efetividade das soluções de software desenvolvidas. ^[1]

Com o *low-code*, as equipas de desenvolvimento podem criar aplicações de forma consideravelmente mais rápida, reduzindo significativamente o tempo necessário para o seu *Deployment*. Esse avanço é possível graças à abstração de camadas complexas de programação em código convencional, o que permite que os desenvolvedores se concentrem na lógica do negócio e menos na implementação técnica. Dessa forma, o Low-Code mostra poder ser uma abordagem benéfica e de grande utilidade, não apenas por reduzir os custos e o tempo associados ao desenvolvimento de software, mas também

por aumentar a produtividade e capacidade de inovação das equipas de desenvolvimento, e também proporciona uma experiência de mercado mais agradável e eficiente ao consumidor final, que irá ver as suas soluções de software mais rapidamente implementadas e com custos diminuídos comparativamente ao mesmo desenvolvimento com tecnologias convencionais. [2]

2.1 História

Nos últimos anos, têm surgido várias tendências na indústria com o objetivo de reduzir a quantidade de código manual necessário para produzir *software*. Desde as linguagens de quarta geração (4GLs) e as ferramentas CASE (Computer-Aided Software Engineering) na década de 1980, até ao Desenvolvimento Rápido de Aplicações (Rapid Application Development) nos anos 90, ao Desenvolvimento pelo Utilizador Final (End-User Development) nos anos 2000 e à Engenharia Dirigida por Modelos (MDE - Model-Driven Engineering) nas últimas duas décadas.[3]

A expressão "*low-code*" teve origem em 2014, com a firma de análise de mercado Forrester, que definiu as plataformas de desenvolvimento *low-code* (LCDPs) como "plataformas que permitem a rápida entrega de aplicações de negócios com um mínimo de codificação manual e investimento inicial mínimo em configuração, formação e implementação". Inicialmente, estas plataformas foram identificadas como específicas para a produção de aplicações empresariais, como *software* para planear recursos empresariais, gestão de relacionamento com o cliente, gestão de processos de negócios e outras aplicações que aumentam a produtividade.[3]

Desde então, a definição evoluiu, e em 2017, a Forrester caracterizou as LCDPs como "produtos e/ou serviços na nuvem para desenvolvimento de aplicações que empregam técnicas visuais e declarativas em vez de programação, estando disponíveis para os clientes a baixo ou nenhum custo em dinheiro e tempo de formação para começar, com os custos a aumentarem proporcionalmente ao valor comercial das plataformas". Esta definição enfatiza as interfaces visuais e as técnicas declarativas, com destaque para o desenvolvimento visual WYSIWYG (What You See Is What You Get) e o desenvolvimento orientado por modelos.[3]

Em 2016, a Gartner identificou um segmento semelhante, denominado de plataforma de aplicação *low-code* (LCAP), introduzindo as LCAPs empresariais, que visam produzir aplicações de classe empresarial com requisitos de desempenho, escalabilidade, disponibilidade, recuperação de desastres, segurança, SLAs (Service Level Agreements), rastreamento do uso de recursos, suporte técnico do fornecedor e acesso API para e de serviços locais e em nuvem.[3]

Em 2021, a maioria dos grandes fornecedores de nuvem oferece LCDPs dentro das suas soluções baseadas na nuvem. A Microsoft foi uma das primeiras a adotar a tendência ao lançar o Power Apps LCPD em novembro de 2016. Em janeiro de 2020, o Google adquiriu o fornecedor de LCPD AppSheet e tornou-o a sua principal solução *low-code*. Em junho de 2020, a Amazon lançou o Honeycode, uma LCPD para desenvolvimento de aplicações web e móveis.[\[3\]](#)

2.2 Drag&Drop

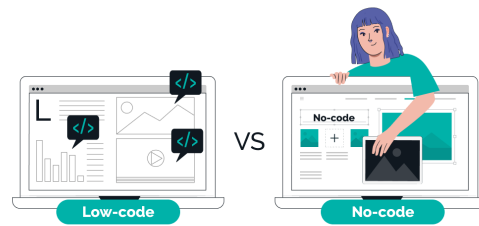
No âmbito do desenvolvimento de software, o *drag&drop* é uma técnica fundamental facilitada pelo paradigma do *low-code*. Esta abordagem permite aos utilizadores arrastar elementos predefinidos e soltá-los numa interface de desenvolvimento, criando assim uma interação visual e intuitiva na construção de aplicações.

A origem remonta aos primórdios da interface gráfica do utilizador (GUI), onde foi introduzido como um método alternativo para manipular objetos digitais. Desde então, tornou-se uma característica padrão em muitas ferramentas de *software*, incluindo aquelas utilizadas para o desenvolvimento de aplicações.

No contexto do *low-code*, o *drag&drop* desempenha um papel crucial na simplificação do processo de criação de *software*. Ao permitir que os utilizadores arrastem componentes como botões, campos de texto, tabelas e lógica de negócios diretamente numa área de trabalho visual, o *drag&drop* elimina a necessidade de escrever manualmente linhas de código complexas.

Além da sua utilidade na construção de interfaces de utilizador, o *drag&drop* também pode ser aplicado na definição de fluxos de trabalho, integração de serviços externos e configuração de regras de negócio. Isso proporciona uma experiência de desenvolvimento mais ágil e eficiente, permitindo que as equipas se concentrem na criação de valor para o negócio em vez de se preocuparem com detalhes técnicos.

3 Low-Code vs No-Code



O desenvolvimento de software passou por uma revolução com a ascensão das tecnologias *Low-Code* e *No-Code*, duas abordagens diferentes que simplificam o processo de criação de aplicações. Enquanto ambos os métodos têm em comum o objetivo de reduzir a dependência de código manual e acelerar o desenvolvimento, suas diferenças fundamentais podem impactar significativamente a escolha entre eles e a forma de aplicá-las em diferentes contextos.[4]

O *Low-Code*, como mencionado anteriormente, permite a criação de aplicações com o uso mínimo de código manual. Ele fornece uma abordagem intermediária, em que os desenvolvedores podem usar interfaces visuais e componentes pré-configurados para agilizar o processo de desenvolvimento. Esta abordagem é ideal para projetos que exigem um alto nível de personalização e flexibilidade, mas onde a velocidade de entrega e a praticidade ainda são uma prioridade. Apesar disso, *Low-Code* pode apresentar algumas limitações em termos de complexidade e escalabilidade, especialmente em projetos altamente customizados ou de grande porte.[4]

Por outro lado, o *No-Code* vai um passo além, eliminando completamente a necessidade de se escrever código. Isso significa que até mesmo utilizadores sem conhecimento técnico sobre programação podem criar aplicações funcionais usando de interfaces visuais intuitivas e técnicas declarativas. O *No-Code* é especialmente adequado para projetos simples ou para utilizadores que procuram criar rapidamente um protótipo para uma ideia, mas pode encontrar muitas limitações em termos de personalização e complexidade.[4]

Uma diferença central entre as duas abordagens está na sua acessibilidade e facilidade de aprendizagem. Enquanto o *Low-Code* requer algum nível de familiaridade com conceitos de programação e desenvolvimento de software, e pode exigir treino adicional na plataforma que está a ser usada, o *No-Code* é projetado para ser tão intuitivo

que até mesmo utilizadores sem experiência prévia em programação podem começar a desenvolver os seus programas imediatamente.[4]

O *Low-Code* oferece mais opções de personalização e controle sobre o processo de desenvolvimento, permitindo que os desenvolvedores adaptem as soluções de acordo com as necessidades específicas de seus projetos. Por outro lado, o *No-Code* pode ser mais limitado em termos de liberdade que o desenvolvedores têm, com menos opções de personalização disponíveis.[4]

4 Implementação de IA com Low-Code

4.1 Chatbot[5]

No desenvolvimento de aplicações/sites em *Low-Code*, para ajudar o utilizador a navegar ou quando este tem alguma dúvida implementa-se um *Chatbot*. Um *Chatbot* não passa de um programa onde está integrado uma IA[6] capaz de simular uma conversa humana, por texto ou por voz. Alguns exemplos que vemos no dia a dia são:

- **Atendimento ao cliente**
Quando o utilizador está com algum problema, em certos sites ou aplicações, existe um assistente virtual para o ajudar, sendo este um *Chatbot*.
- **Bancos**
Alguns bancos disponibilizam de *Chatbots* para algumas tarefas que não requeiram de uma pessoa.
- **Educação**
Em plataformas de *self learning* online existem *Chatbots* para ajudar o utilizador a pôr em prática o que aprendeu.
- **Comércio online**
Os *Chatbots* podem ser usados para localizar um produto ou ajudar o utilizador com algum problema durante as compras.
- **Saúde**
Para se fazer a marcação de uma consulta pode-se usar um *Chatbot*.

5 Soluções do Low-Code - Um pouco do que há no mercado

Nos últimos anos, o mercado de plataformas de desenvolvimento *low-code* tem testemunhado um crescimento significativo em resposta à demanda por soluções de desenvolvimento de software mais rápidas e eficientes. Uma variedade de empresas, desde grandes players do setor de tecnologia até startups inovadoras, têm desenvolvido e lançado suas próprias plataformas de low-code, cada uma com as suas características e funcionalidades únicas.

Exploraremos algumas das principais soluções disponíveis no mercado de *low-code*, oferecendo uma visão geral de suas capacidades e áreas de foco. Cada plataforma abordada aqui representa uma abordagem distinta para o desenvolvimento de software de baixo nível, oferecendo uma gama de ferramentas e recursos para atender às necessidades de diferentes tipos de projetos e organizações.

Iremos analisar com mais pormenor duas das plataformas líderes de *low-code* disponíveis atualmente:

- OutSystems;
- Zoho Creator;

5.1 OutSystems



Em 2001, OutSystems foi fundada com uma visão audaciosa: revolucionar a entrega de *software*. Os seus fundadores, experientes profissionais de IT, perceberam que a maioria dos projetos de *software* enfrentava problemas com prazos e orçamentos.[7]

Naquela época, predominava a abordagem "*waterfall*" [8] no desenvolvimento de *software*, onde mudanças tardias se tornavam cada vez mais custosas. OutSystems questionou esse paradigma, propondo uma nova abordagem: e se as mudanças fossem rápidas e económicas em qualquer estágio do ciclo de vida do *software*?[7]

Essa visão levou à criação da plataforma OutSystems, projetada para tornar as alterações em aplicações rápidas, robustas e acessíveis, independentemente do tamanho

do projeto. Essa abordagem ágil e flexível colocou a OutSystems na vanguarda do desenvolvimento de *software*, oferecendo soluções que se adaptam dinamicamente às necessidades em constante evolução das empresas.[7]

OutSystems é uma plataforma moderna de aplicações projetadas para acelerar significativamente o desenvolvimento das aplicações mais críticas de uma empresa, oferecendo níveis sem precedentes de flexibilidade e eficiência. Desenvolvedores usam um ambiente de desenvolvimento integrado único que abrange todo o ciclo de vida do desenvolvimento: desenvolvimento, garantia de qualidade, implementação, monitorização e gerenciamento.[9]

A plataforma da OutSystems foi criada por uma equipa de profissionais de IT portugueses, com o objetivo de aproveitar todas as vantagens da abordagem *low-code* no desenvolvimento de *software*. Ela serve como um ambiente visual de desenvolvimento, compilador, repositório e centro de testes, proporcionando uma experiência unificada para o desenvolvimento de Web Apps, Mobile Apps e outras aplicações.[9]

A abordagem *low-code* de OutSystems permite que os desenvolvedores criem aplicações de forma mais rápida e eficiente, reduzindo significativamente o tempo e os custos associados ao desenvolvimento de *software* tradicional. A plataforma oferece um ambiente de desenvolvimento visual integrado, no qual os desenvolvedores podem criar a interface do utilizador, o back-end, a base de dados e a integração com sistemas ou serviços existentes de forma intuitiva e eficaz.[9]

Além disso, OutSystems fornece uma série de recursos e funcionalidades para cobrir todo o ciclo de vida do desenvolvimento de aplicações. Isso inclui suporte para desenvolvimento rápido e integração com sistemas e base de dados existentes, implementação fácil e segura de aplicações em produção, monitorização contínua da saúde das aplicações e facilidades de gerir operações diárias.[9]

A plataforma é altamente flexível e extensível, permitindo que os desenvolvedores criem uma ampla variedade de aplicações empresariais para diferentes necessidades e cenários de uso. Além disso, a OutSystems oferece suporte para a construção de aplicações móveis altamente interativos, embora possa não ser otimizada para casos de uso extremamente específicos, como jogos altamente interativos.[10]

5.1.1 Principais Recursos, Use Cases e Benefícios da Plataforma OutSystems

A plataforma OutSystems oferece uma ampla gama de recursos e funcionalidades projetadas para acelerar o desenvolvimento de aplicações empresariais e impulsionar a inovação digital. Alguns dos principais recursos e benefícios que tornam a OutSystems uma solução poderosa:

- **Desenvolvimento Rápido e Eficiente**

Com a abordagem *low-code* de OutSystems, os desenvolvedores podem criar aplicações de forma rápida e eficiente, reduzindo significativamente o tempo necessário para o desenvolvimento. O ambiente de desenvolvimento visual integrado permite a criação de interfaces de utilizador, lógica de negócios e integrações com sistemas externos de maneira intuitiva e eficaz.[9]

- **Flexibilidade e Extensibilidade**

A plataforma OutSystems é altamente flexível e extensível, permitindo que os desenvolvedores criem uma ampla variedade de aplicações empresariais para diferentes necessidades e use cases. Com suporte para uma variedade de tecnologias e integrações, as aplicações desenvolvidas em OutSystems podem ser facilmente adaptadas e expandidas para atender às necessidades em constante evolução das empresas.[9]

- **Integração com Sistemas e Dados Existentes**

OutSystems oferece recursos abrangentes para integração com sistemas e dados existentes, permitindo que as aplicações desenvolvidas na plataforma se conectem facilmente a sistemas, bases de dados corporativas e serviços externos. Isso permite que as empresas aproveitem seus investimentos existentes em tecnologia e maximizem o valor de seus dados.[9]

- **Implementação e Gestão Simplificados**

A plataforma OutSystems facilita a implementação e o gestão de aplicações em produção, com recursos para automação, versionamento de código e monitorização contínuo da saúde das aplicações. Isso ajuda as empresas a manter as suas aplicações funcionarem de forma eficiente e confiável, garantindo uma experiência contínua para os utilizadores finais.[9]

Use Cases:

OutSystems é adequada para uma ampla variedade de use cases, desde aplicações empresariais internas até aplicações de consumo voltados para o cliente. Alguns exemplos incluem aplicações de CRM[11], gestão de inventário, automação de processos, aplicações móveis para funcionários e clientes, entre outros.[10]

5.2 Zoho Creator[12]



O Zoho Creator é uma ferramenta que tal como o OutSystems serve para desenvolver aplicações em *Low-Code*. A Zoho, antigamente chamada de AdventNet, Inc., foi fundada em 1996. O Zoho Creator foi criado em 2006 usado só pela sua empresa.

Uma das vantagens do Zoho Creator é a sua acessibilidade, uma vez que pode ser acedido pela *web* ou por aplicações para Android ou iOS.

Em comparação com OutSystems o Zoho Creator tem uma configuração mais fácil, não precisa de qualquer tipo de treino ou curso para se começar a usar, o construtor é totalmente baseado na *cloud* e *updates* e *backups* automáticos.

O Zoho Creator permite:

- Criação de aplicações consoante a necessidade do utilizador
- Automação de fluxo de trabalho com facilidade
- Conectar ou extender outras aplicações

Use Cases:

O Zoho Creator é ideal para um uso abrangente, desde aplicações empresariais até aplicações de consumo voltados para o cliente. Alguns exemplos incluem aplicações de CRM[11], usar IA, como *Chatbots*, nas suas aplicações, automação de processos, aplicações móveis, obter relatório e *insights* a tempo real, entre outros.

6 Conclusão

O *Low-Code* representa uma mudança significativa na forma como o *software* é desenvolvido e implementado, oferecendo uma abordagem mais rápida, eficiente e acessível para a criação de aplicações. Ao simplificar o processo de desenvolvimento, o *Low-Code* permite que as empresas acelerem a entrega de soluções de *software*, aumentem a produtividade das equipes de desenvolvimento e ofereçam uma experiência de mercado mais ágil e eficiente para os consumidores finais.

No entanto, é importante reconhecer que o *Low-Code* não é uma solução única para todos os problemas de desenvolvimento de *software*. Existem casos em que a programação manual pode ser necessária para lidar com requisitos complexos ou personalizados. Portanto, é essencial avaliar cuidadosamente as necessidades de cada projeto e escolher a abordagem de desenvolvimento mais adequada.

Além disso, o *Low-Code* tem potencial para promover a inclusão digital, permitindo que pessoas com pouca ou nenhuma experiência em programação contribuam para o desenvolvimento de aplicações. Isso pode levar a uma democratização do desenvolvimento de *software*, abrindo portas para uma gama mais ampla de talentos e perspectivas.

Para futuras investigações, áreas interessantes a serem exploradas incluem a integração de *Low-Code* com inteligência artificial para automação de processos, o impacto do *Low-Code* na agilidade e inovação das empresas e o desenvolvimento de aplicações específicas para setores como saúde, educação e finanças.

Em suma, o *Low-Code* está posicionado para continuar a moldar o futuro do desenvolvimento de *software* nos próximos anos, oferecendo oportunidades emocionantes para inovação, crescimento e inclusão.

References

- [1] OutSystems: Low-code basics: The future of app development (2020)
- [2] Bock, A., Frank, U.: Low-code platform (2021)
- [3] Di Ruscio, D., Kolovos, D., Lara, J., Pierantonio, A, W.M. Tisi. M: Low-code development and model-driven engineering: Two sides of the same coin? (2022)
- [4] Kissflow, T.: Difference between low-code and no-code platform (2024)
- [5] Daniel, G., Cabot, J., Deruelle, L., Derras, M.: Xatkit: A multimodal low-code chatbot development framework (2020)
- [6] Oracle: What is ai? learn about artificial intelligence
- [7] OutSystems: It began with a vision
- [8] Lutkevich, B.: What is the waterfall model? (2022)
- [9] OutSystems: Developing with outsystems
- [10] OutSystems: What is not a good project for outsystems?
- [11] Oracle: What is crm? the complete crm guide.
- [12] Zoho: Zoho blog