

# No-code Low-code



# — low-code

Baixo uso de código



## — Low-code

- *Low-code* ou “baixo uso de código” é uma maneira para desenvolvedores de todos os níveis de habilidade projetarem aplicativos rapidamente e com o mínimo de codificação manual, arrastando e soltando blocos visuais de código existente em um fluxo de trabalho para criar aplicativos.
- Construir software com *low-code* é o mesmo que construir software de qualquer outra forma, com a principal diferença sendo os tipos de caminhos oferecidos.
- Em vez de codificar manualmente um sistema de gerenciamento de usuários, aprender a estrutura de programação mais recente ou escrever 10 testes antes de uma única linha do código do aplicativo, você vai direto para a criação de algo novo e valioso.



## — Low-code – Vantagens

- **Velocidade:** com baixo código, você pode criar aplicativos para várias plataformas simultaneamente e mostrar exemplos de trabalho para as partes interessadas em dias ou até horas.
- **Mais recursos:** Se você está trabalhando em um grande projeto, não precisa mais esperar que desenvolvedores com habilidades especializadas concluam outro projeto demorado, o que significa que as coisas são feitas mais rapidamente e a um custo menor.
- **Baixo risco / alto ROI:** com baixo código, processos de segurança robustos, integração de dados e suporte de plataforma cruzada já estão integrados e podem ser facilmente personalizados - o que significa menos risco e mais tempo para se concentrar em seus negócios.
- **Implantação com um clique:** com baixo código, um único clique é tudo o que você precisa para enviar seu aplicativo para produção. O dia do lançamento não é mais uma experiência estressante



**no-code**

Sem código



## — no-code

- As soluções sem código também apresentam desenvolvimento visual de arrastar e soltar.
- Ao contrário do código baixo, eles atendem principalmente a pessoas de negócios ou outros em TI que podem não saber nenhuma linguagem de programação real, mas desejam desenvolver um aplicativo para um caso de uso específico - geralmente para seu departamento.
- Em outras palavras, o no-code permite que as organizações equipem as equipes com as ferramentas de que precisam para criar aplicativos sem treinamento formal de desenvolvimento.



## — no-code - Vantagens

- Sem código é ótimo se você precisa de um aplicativo simples para resolver um único problema comercial ou de departamento e não quer esperar que a TI o crie e entregue dentro de 3 a 6 meses.
- As plataformas sem código exigem muito pouco treinamento, portanto, qualquer pessoa em sua organização pode criar um aplicativo, geralmente no domínio do gerenciamento de processos de negócios, como aprovações de despesas.
- O sem código oferece aos usuários de negócios a liberdade de atender a uma necessidade imediata sem desviar a TI de projetos de desenvolvimento de missão crítica.



# — Futuro

Gartner aponta o futuro do no-code ou low-code





## — Low-code e no-code são o futuro ?

Até 2023, mais de 50% das médias e grandes empresas terão adotado como uma das plataformas de aplicativos estratégicos.

Low-code será responsável por 65% da atividade de desenvolvimento de aplicativos até 2024.



## — Prós das ferramentas NoCode

- Criação de aplicativos móveis com pouca ou nenhuma experiência em codificação.
- Criação de um aplicativo da web com pouca ou nenhuma experiência em codificação.
- Obter uma melhor compreensão dos princípios de codificação.
- Grande diversidade e capacidade de alterar seus aplicativos com um simples arrastar e soltar.
- Economize recursos porque os desenvolvedores podem não ser necessários.

## — **Contras das ferramentas NoCode**

- Por mais úteis e avançadas que sejam as ferramentas NoCode, existe a possibilidade de você não conseguir criar seu aplicativo da maneira exata que deseja.
- Riscos de segurança potenciais ao usar código que outra pessoa escreveu. Suponha que alguém com intenções maliciosas em sua mente injete pedaços de código. As chances são boas de que um problema se desenvolva.

**Web Apps**

**Business Apps**

**Automation and Workflow**

**Mobile Apps**

**LowCode**

**Chatbot**

**Marketing**

**Database**

**Landing Page**

**Payments**

**Design**

**Membership**

## STATE OF NOCODE

## LowCode

## Landing Page



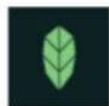
unicornplatform



Hubspot



Launchaco



Leeflets



Unbounce



Instapage



GetResponse



PORY.IO

## Design



Smaal



Pastry

## Payments



Nuclino



Spider



FormBeaver



ActionDesk

waley

Waley



Trolley



Servicebot



PayRequest



payhere



Buy Me A Coffee

CP

Checkout Page

INSTANDA

Indanda

## Membership



MemberSpace



MemberStack

## — TECNOLOGIAS PARA EXPLORAR

- WORDPRESS
- AppGyver
- Bubble
- Cronapp
- FlutterFlow
- Mendix
- Outsystems
- Adianti Builder - PHP

47 ferramentas

<https://userguiding.com/pt-br/blog/ferramentas-no-code/>

Testei +100 ferramentas e essas são

**AS MELHORES FERRAMENTAS NOCODE**

<https://www.youtube.com/watch?v=wbdU5wUc9Ws>

# **PORQUE ESSE MOVIMENTO?**

Quais as Perguntas?



## — Perguntas

Os programadores irão sumir?

Surgiu por conta do apagão?

Todo mundo vai saber fazer seus próprios aplicativos?

Quais ferramentas posso usar ?  
Realmente acelera a entrega ?

O Shadow IT pode vir a ser o problema?


Tudo estará padronizado?








Opinião de outros  
CDF Código Fonte TV  
**CLIQUE PARA VER O VIDEO**



Opinião de outros  
Renato Asse – Sem Codar  
— No Code x Low Code x  
Programação: A verdade  
que ninguém te contou  
ainda

**CLIQUE PARA VER O VIDEO**





Desmistificando o NO-Code  
Jair Jr e Caio Calderari  
**CLIQUE PARA VER O VIDEO**



Plataformas

**Source Forge – Guia**  
**18 plataformas –**

# Bubble.io – Léo Andrade – #1 a #12

#1

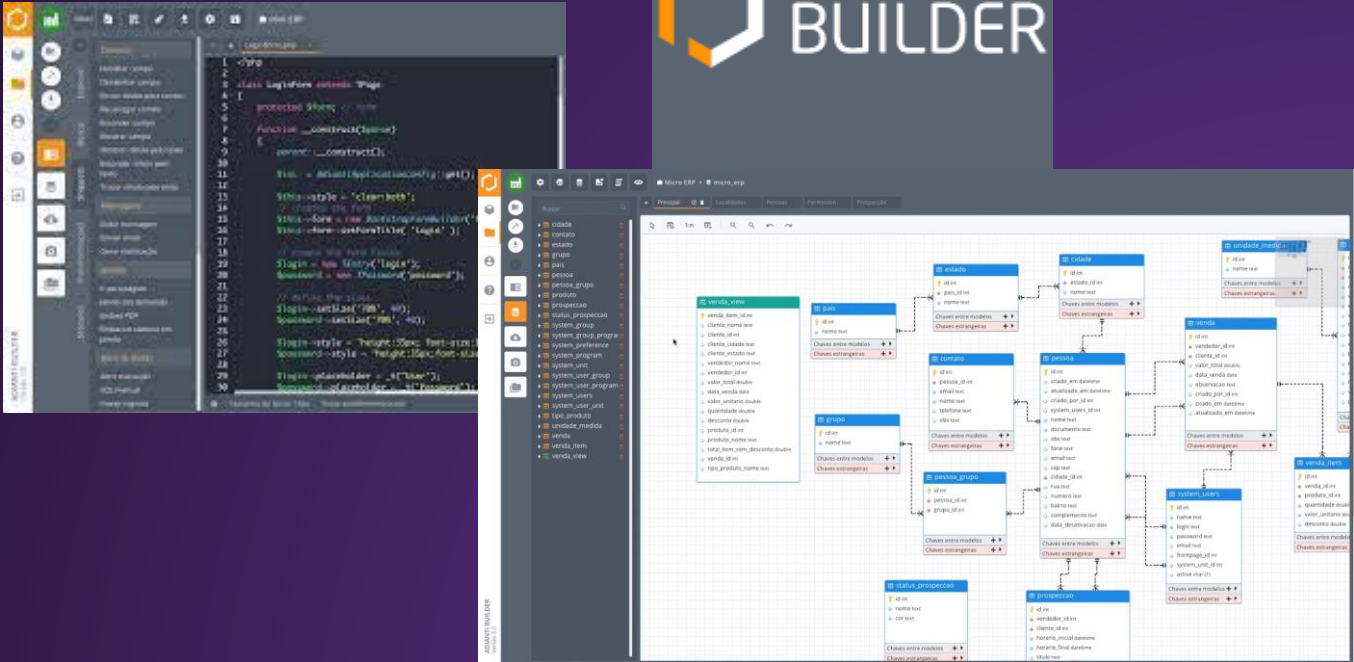
#2

#4

#3

LIVE  
7 erros do Bubble

Downloaded from <http://ajph.org/> at University of California, San Diego on June 11, 2015





```

void CSSdescramble(unsigned char *sec,unsigned char *key) {
    unsigned int t1,t2,t3,t4,t5,t6;
    unsigned char *end=sec+0x800;
    t1=key[0]^sec[0x54]|0x100;
    t2=key[1]^sec[0x55];
    t3=((unsigned int *)(key+2))^((unsigned int *)(sec+0x56));
    t4=t3&7;
    t3=t3*2+8-t4;
    sec+=0x80;
    t5=0;
    while(sec!=end) {
        t4=CSSt2[t2^CSSt2[t1]];
        t2=t1>>1;
        t1=((t1&1)<<8)^t4;
        t4=CSSt5[t4];
        t6((((((t3>>3)^t3)>>1)^t3)>>8)^t3)>>5)&0xff;
        t3=(t3<<8)|t6;
        t6=CSSt4[t6];
        t5+=t6+t4;
        *sec++=CSSt1[*sec]^(t5&0xff);
        t5>>=8;
    }
}

```

# THANKS!

Alguma Questão?

