

Disposition

Table des matières

Introduction No-code/low-code	6
1. 1. No-code/Low-code.....	7
1.1. La croissance de la programmation sans code ou peu de code	7
1.1.1. Les avantages et les inconvénients.....	7
1.1.2. Les usages les plus répandus	Erreur ! Signet non défini.
2. From-cratch	9
2.1. Les besoins	9
2.1. Les usages les plus répandus.....	9
Conclusion.....	10

Table des illustrations

Figure 1 : Illustration d'outils no-code/low-code	6
---	---

Table des légendes

Légende 1 : [Anakeen plateforme]	6
--	---

Introduction

Démocratiser l'accès à la création d'applications, réduire le temps de développement et de mise en ligne des sites web, tout en diminuant drastiquement les coûts. Telles sont les alléchantes promesses des plateformes no-code et low-code.

« De nombreuses start-up font leur premier tour de table avec un site no-code. »

Le no-code est très simple : pas de code, pas de programmation. Une confusion existe entre le low et no code, parfois entretenu par les éditeurs eux-mêmes.

L'objectif est de remplacer le code par une interface graphique de l'application, notamment la partie API, connexion aux données et la prise en charge de la mobilité doit être transparente.

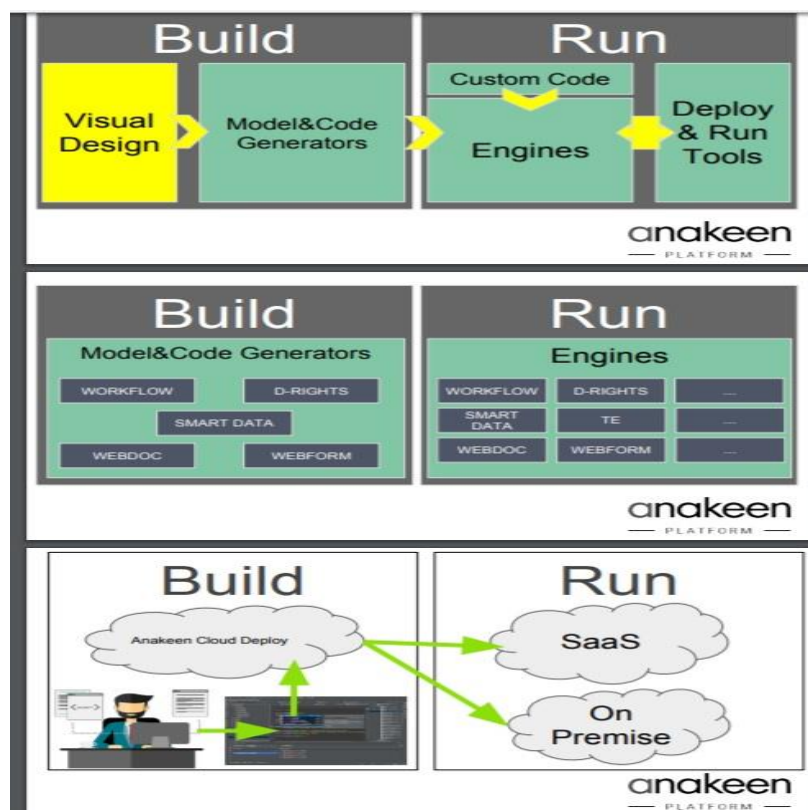


Figure 1 outils no-code

Source : Anakeen platform

1. No-code/Low-code

Le No-code/Low-code permet aux utilisateurs ne sachant pas écrire de ligne de code, de développer les mêmes applications qu'un développeur web. Plusieurs plateformes en ce sens, ont fleuri au cours de cette dernière décennie, pour la création de sites web.

Il est fréquent que ce qui est désigné sous le nom de No Code soit en fait du Low Code. En effet, il s'agit bien souvent d'outils qui nécessitent des notions de code (HTML, CSS, voire JS, Python ou SQL) pour parvenir à des résultats probants. Le Low Code permet aux développeurs d'intégrer des briques plus personnelles de code dans un environnement.

Parmi ses plateformes nous pouvons citer :

ZAPIER, GLIDE, WEBFLOW, BUBBLE.

1.1. La croissance de la programmation sans code ou peu de code

Bien qu'il soit le fruit de l'évolution technologique, le développement sans code n'est pas un concept nouveau.

Les outils de génie logiciel assisté par ordinateur (CASE) des années 90 sont antérieurs aux plates-formes sans code d'aujourd'hui. Mais les outils CASE, comme leur nom l'indique, ne visent qu'à faciliter certaines activités du cycle de vie du développement de logiciels, et non le codage lui-même. Microsoft Visual Basic et Adobe Dreamweaver - qui nécessitent encore une connaissance du code - sont considérés comme des itérations antérieures des outils sans code. Le développement sans code est également une solution à un problème d'offre et de demande : une demande croissante pour générer plus de logiciels, mais un nombre limité de développeurs qui peuvent créer ces logiciels.

1.1.1. Les avantages et les inconvénients

Il permet de créer et d'utiliser des applications. Le no-code/low-code est utile aux développeurs et aux utilisateurs lambda. Les SI possédant une ou plusieurs équipes de développement surchargées, peuvent retirer une valeur ajoutée de ce type d'outils.

Il permet aussi une meilleure agilité car il se fait via une interface glisser-déposer, construite visuellement avec des modules prédéfinis. Ce type de produit permet donc d'accéder à des outils numériques performants pour des coûts et des délais qui restent maîtriser la ou les budgets sont sans cesse le moteur tendant vers son utilisation.

A contrario, la décision d'avoir recours à ce type de plateforme doit être une décision mûrement réfléchie.

En effet l'utilisateur doit posséder une compréhension exacte des besoins en relation avec le projet. Un cahier des charges rigoureux permettra de palier à cet impératif, car cela évitera des problèmes liés à l'aspect fonctionnel et évolutif du projet.

A cela peuvent s'ajouter les problèmes de sécurité et de propriété intrinsèque de votre code.

1.1.2. Les usages les plus répandus

Le développement no-code/ low-code est particulièrement bien adapté aux usages suivants :

Sites Web avec des fonctions simples ;

Prototypes de sites SaaS ;

CRM, gestion de fichiers clients ;

2. From-cratch

Un site from scratch est un site web codé sur-mesure par des développeurs. Il répond à un cahier des charges fonctionnel précis et comporte généralement des fonctionnalités assez avancées. Il est voué à évoluer régulièrement en fonction de l'évolution de la structure. Il proposera certainement des fonctionnalités novatrices qui ne peuvent être prise en charge par des composants existants.

2.1. Les besoins

La décision de développer un site web from-scratch résulte d'une inexistence sur le marché de solution correspondant aux attentes recherchées.

Un site web de ce type nait pour répondre à un besoin, évolue et demande une maintenance quasi quotidienne tout au cours de son exploitation. Tant en termes de fonctionnalité et de sécurité.

Il ne faudra pas non plus négliger son coût de mise en œuvre car cela nécessitera l'intervention d'une ou de plusieurs équipes de développeur ainsi qu'un pôle rattaché au graphisme. Néanmoins la qualité de service et le résultat final correspondra à vos attentes.

2.1. Les usages les plus répandus

Le développement from-scratch est particulièrement bien adapté aux usages suivants :

Sites Web avec des fonctions complexes ;

Sites Web avec intégration dans des processus d'acquisition de leads;

PoC – Proof of Concept – de sites SaaS et/ou d'applications Web et mobiles ;

Logiciels orientés « métier » ;

Conclusion

Les outils low-code et no-code sont devenues un sujet important dans le monde du développement web, et pour cause. Des interfaces simples et conviviales avec des fonctionnalités de glisser-déposer permettent à quiconque de développer des applications professionnelles sans avoir besoin d'être un expert dans le développement.

Tant les développeurs expérimentés, que les développeurs autodidactes, le fait de bénéficier d'un processus de développement rapide et fiable dû à l'outil low-code/no-code n'est pas négligeable. La flexibilité de la technologie low-code et no-code permet une grande maniabilité dans le développement d'applications allant de la création simpliste à des solutions bien plus évoluées.

Compte tenu de tous ces facteurs et des avantages du low-code et du no-code, nous pouvons nous attendre à ce que le développement d'applications low-code/no-code devienne plus répandu.

Leur adaptabilité, leur agilité et leur capacité à automatiser les étapes métier font de la technologie low-code et no-code un allié indispensable pour les entreprises qui espèrent prospérer à l'ère numérique.

Néanmoins la particularité de créer un site web from scratch permet de disposer de beaucoup plus de liberté comparée aux outils no-code/low-code. En effet, la possibilité d'intégrer et de créer des fonctionnalités, confère à votre site une dimension sur mesure permettant une adaptation certaine dans l'élaboration d'un projet sur-mesure. Cela permettra d'avoir un site unique et parfaitement adapté à vos besoins.

En créant un site from scratch, vous êtes en accord parfait avec la vision de base de vos fonctions et leurs utilités.

L'intérêt est d'avoir un résultat final professionnel.

Avec un site internet créé from scratch, vous disposerez d'un site à l'apparence unique et avec des fonctionnalités qui lui seront propres. C'est d'ailleurs pour cela que de nombreuses grandes entreprises ont choisi cette option pour créer leur site web.