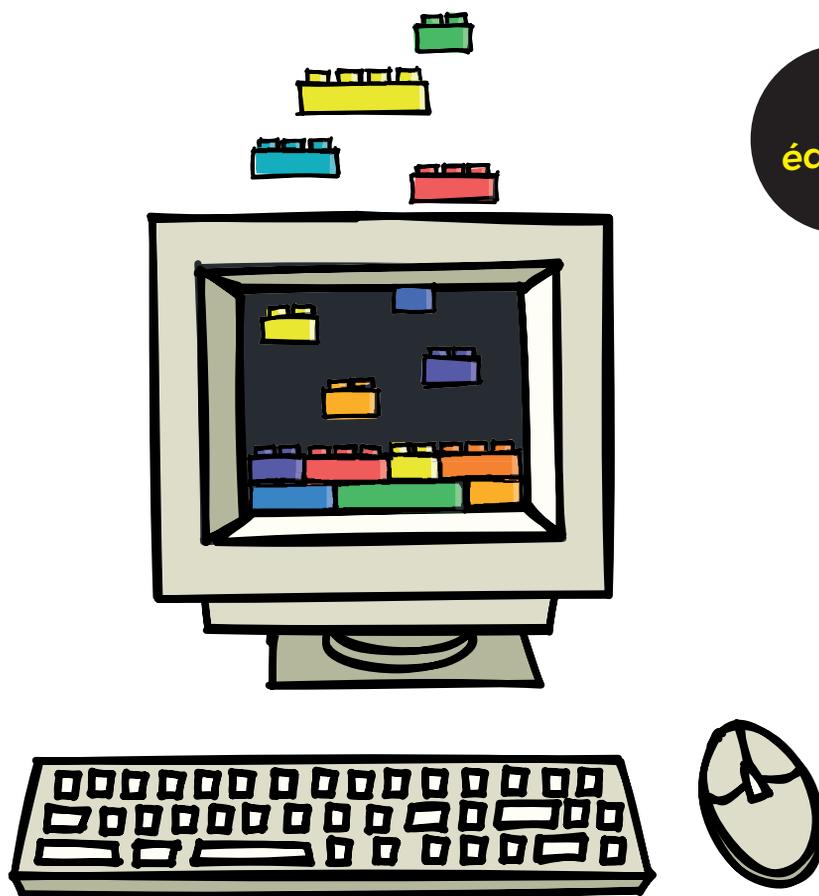


ALEXIS KOVALENKO - ERWAN KEZZAR - FLORIAN REINS

Préface d'Emmanuel Straschnov, fondateur de Bubble

NO-CODE

Une nouvelle génération d'outils numériques



2^e
édition

● Éditions
EYROLLES

Éditions Eyrolles
61, bd Saint-Germain
75240 Paris Cedex 05
www.editions-eyrolles.com

Illustration de couverture : © Victor Grandchamp, ainsi que les illustrations
des pages 9, 28, 31, 63, 112, 130, 161, 231, 237, 244, 257, 298

© Alan Kay : p. 315

Mise en pages : Sandrine Escobar

Depuis 1925, les éditions Eyrolles s'engagent en proposant des livres pour comprendre le monde, transmettre les savoirs et cultiver ses passions ! Pour continuer à accompagner toutes les générations à venir, nous travaillons de manière responsable, dans le respect de l'environnement. Nos imprimeurs sont ainsi choisis avec la plus grande attention, afin que nos ouvrages soient imprimés sur du papier issu de forêts gérées durablement. Nous veillons également à limiter le transport en privilégiant des imprimeurs locaux. Ainsi, 89 % de nos impressions se font en Europe, dont plus de la moitié en France.

En application de la loi du 11 mars 1957, il est interdit de reproduire intégralement ou partiellement le présent ouvrage, sur quelque support que ce soit, sans l'autorisation de l'Éditeur ou du Centre Français d'exploitation du droit de copie, 20 rue des Grands Augustins, 75006 Paris.

© Éditions Eyrolles, 2022, 2024
ISBN : 978-2-416-01521-2

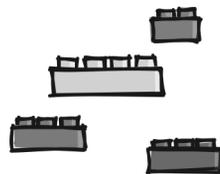
ALEXIS KOVALENKO - ERWAN KEZZAR - FLORIAN REINS

Préface d'Emmanuel Straschnov, fondateur de Bubble

NO-CODE

Une nouvelle génération d'outils numériques

2^e édition



● Éditions
EYROLLES

Préface

En 1984, Apple dévoile le Macintosh et son système d'exploitation Mac OS System 1. Pour la première fois, l'utilisation d'un ordinateur se fait par une interface graphique, et non par la ligne de commande. Un an plus tard, Microsoft lance Windows 1.0 et proclame sa vision « *A microcomputer on every desk and in every home* ». Désormais, il devient clair que l'utilisation des outils informatiques et bureautiques sera destinée au plus grand nombre. L'impact sur la vie professionnelle et personnelle des individus sera profond.

La programmation, en revanche, a longtemps été un domaine réservé aux experts. De nombreuses tentatives ont été faites depuis les années 1990, mais ce n'est qu'à partir des années 2010 que des nouvelles technologies ont réellement commencé à ouvrir la programmation au plus grand nombre. En 2012, avec Josh Haas, nous lançons Bubble, dans le but de permettre à tous de développer des applications web complexes sans compétences techniques. Dix ans plus tard, d'autres outils nous ont rejoints, et cette tendance, désormais appelée le « no-code », est en passe de devenir un standard à part entière, lequel va changer profondément la façon dont les gens créent et innovent. Les outils no-code permettent au plus grand nombre de créer eux-mêmes des outils digitaux – site, application, automatisation – dont ils ont besoin dans leur vie professionnelle et personnelle. L'avantage est immense : les outils créés répondent par conséquent bien plus souvent aux besoins initiaux, et peuvent évoluer au fur et à mesure de l'évolution de la situation, sans dépenses d'ingénierie.

On voit déjà l'impact du no-code sur la vie quotidienne : en novembre 2020, un utilisateur de Bubble, originaire de Rennes, a souhaité me raconter son histoire. Avec deux médecins, il gère une association pour venir en aide à des femmes enceintes et les aider à se préparer à l'accouchement. Du fait des différents confinements, l'association n'est plus en mesure de continuer à opérer en personne, et doit migrer sur une solution en ligne. Des vidéos sont proposées,

mais il leur manque une application pour digitaliser l'expérience qu'ils offraient jusqu'alors en personne. Sans moyens financiers importants, Bubble devient leur seule chance de recréer l'expérience en ligne et de continuer à aider ces futures mamans. Après quelques semaines de développement, ils lancent leur solution en ligne et ont commencé à explorer une expansion dans d'autres régions.

Cette démocratisation est d'autant plus importante que l'exposition au code dépend souvent de l'origine sociale, en particulier chez les jeunes. Et dans la mesure où la création d'entreprise, et pas seulement des start-up, passent de plus en plus par le digital, il est important de casser ce déterminisme. C'est ce qui nous a amené à créer le programme Immerse, aux États-Unis, afin d'offrir un accès encore plus simple à notre technologie par des classes gratuites à des entrepreneurs au sein de la minorité afro-américaine. Après deux ans, près de 50 *fellows* ont lancé leur start-up aux États-Unis et en Afrique. Ce n'est que le début de la démocratisation de la technologie et de l'entrepreneuriat par le no-code.

Les premières années du mouvement du no-code étaient consacrées au développement des outils ; la barre était haute en termes de fonctionnalité et de puissance pour convaincre que ces outils graphiques pouvaient égaler la programmation traditionnelle. Depuis 2020, ces technologies ont atteint cette maturité, et l'effort doit désormais être porté sur l'éducation, afin d'accélérer la transition vers cette nouvelle étape de la révolution technologique. La France est en avance, car la communauté s'est lancée plus tôt qu'ailleurs, il s'agit maintenant d'accélérer. Erwan Kezzar me contactait dès 2014 pour discuter de la meilleure façon d'enseigner Bubble, et je suis ravi de pouvoir écrire cette préface. Ce livre est à la fois une bonne introduction pour comprendre ce que sont les outils no-code, mais aussi pour dépasser le simple exercice de définition, et ainsi s'imprégner de tous les phénomènes qui les entourent – qu'il s'agisse de la façon dont la programmation visuelle s'inscrit dans l'histoire des nouvelles technologies, ou dont les communautés s'emparent de tout ce qui se passe sur ce terrain depuis quelques années.

Emmanuel Straschnov,
fondateur et CEO de Bubble

Table des matières

Avant-propos	1	Un état d'esprit libérant la créativité • 24
PARTIE 1		Qu'est-ce que le vendor lock-in ? • 24
LE NO-CODE,		En quoi le no-code atténue-t-il le vendor lock-in ? • 26
QU'EST-CE QUE C'EST ?	7	Un argument marketing propageant quelques malentendus • 28
1 Émergence du no-code	9	Code vs no-code : un affrontement vain et stérile • 28
Une productivité accrue pour toutes les entreprises • 11		Le no-code ne dispense pas de connaissances en code • 29
Un sujet qui intéresse aussi les géants du Web • 13		Le no-code dépasse le domaine de la programmation • 31
Le no-code annonce-t-il une nouvelle ère dans la fabrication d'outils numériques ? • 15		Des développements qui s'accélèrent • 32
Un vaste choix d'outils • 16		Des projets no-codés en quelques jours • 32
Des outils en grand nombre • 16		Des outils no-code évoluant à vive allure • 34
Des outils qui invitent à être testés • 17		2 Panorama de projets réalisés sans coder
Les grandes catégories d'outils no-code • 18		Des apps no-codées en un week-end aux retombées impressionnantes • 39
Des fonctionnalités à assembler soi-même • 21		
Des fonctionnalités aussi faciles à assembler que des briques de Lego • 21		
Inventaire des principales briques fonctionnelles • 22		

No-code

- Glide et la vague des emergency apps • 40
- D'autres applications d'urgence • 50
- Des transformations internes • 60**
 - « Loom » • 61
 - « L'Équipe Explore » • 64
 - « TED Fellows Program » • 66
- Des lancements d'activités • 67**
 - « Dwellito » • 67
 - « New Story Charity » • 69
 - « Prelo » • 70
 - « L'Intendance » • 72
 - « Comet » • 74
- Des TPE-PME • 76**
 - Top'Isol • 76
 - Vivr'AG • 77
- Des grandes entreprises • 78**
 - Havas Market • 78
 - Enedis • 79
 - Engie • 79
- Des institutions et collectivités publiques • 80**
 - La mairie de Pau • 80
 - L'Observatoire des démarches numériques • 81
- 3 Qu'est-ce qu'un outil no-code ? 83**
 - Le no-code selon Adalo et Formstack • 83
 - Caractérisation des outils no-code • 87
 - La programmation visuelle • 91
 - Mettre en forme des textes • 93
 - Programmer spatialement des écrans • 97
 - Programmer temporellement des scénarios • 111
 - L'interconnexion de services • 115**
 - La « stack » ou les différentes couches d'un projet • 115
 - La gestion des données • 117
 - Typologie des briques interconnectées • 122
 - Différents types d'interconnexion • 123
 - L'accessibilité des outils no-code • 125**
 - Des outils nombreux et variés • 126
 - Des outils aux tarifs abordables • 127
- 4 Histoire et origines techniques du no-code 131**
 - La notion d'abstraction en informatique • 134**
 - Les premiers pas de l'informatique • 134
 - Un besoin d'abstraction • 137
 - La révolution de la programmation orientée objet • 145
 - L'ère des réseaux et des services • 148**
 - L'abstraction de l'hébergement et l'avènement du cloud • 148
 - Le développement d'Internet et des réseaux physiques • 150
 - L'essor des navigateurs web et de JavaScript • 151
 - L'essor de l'UX • 155
 - Un exemple : de Word à Notion • 157

Complexité technique + simplicité ergonomique = « Ça marche » • 159

PARTIE 2

EXISTE-T-IL UN MOUVEMENT

NO-CODE ? 165

5 **Mouvement et communautés** .. 167

Contexte • 167

Les constituants
d'une communauté • 169

Le cas des grandes entreprises • 171

Un exemple iconique de
communauté : Indie Hackers • 173

Qu'est-ce que Indie Hackers ? • 175

Comment Indie Hackers
a-t-elle été créée ? • 175

Quelques leçons
de Indie Hackers • 177

Quelques exemples
de communautés no-code • 179

Makerpad, une « communauté
apprenante » consacrée
au no-code • 179

Des communautés rattachées
à des outils no-code : les approches
Bubble et Webflow • 181

Des communautés secondaires
associées à des outils no-code • 185

No-Code France • 186

Que trouve-t-on dans les
communautés digitales ? • 191

Partage de templates • 192

Des questions posées
à la communauté • 193

Des success stories • 195

Des systèmes d'ambassadeurs • 198

Des tutoriels et des guides pratiques
accessibles • 199

6 **Une variété d'approches et d'attitudes** 203

Influences culturelles
du no-code • 203

Existe-t-il des valeurs
no-code ? • 207

Trois profils de no-codeurs • 213

Lise, « hackeuse en mode
no-code ops » • 214

Julien, « craftsman, artisan
du no-code » • 224

Naye, « makeuse lanceuse
de projets » • 228

PARTIE 3

COMMENT BIEN PRATIQUER

LE NO-CODE ? 237

7 **Lancer des projets numériques sans coder** 239

Bien démarrer en no-code • 240

Ce qui ne sera pas abordé ici • 240

Quelques conseils généraux • 242

À chacun son parcours
de progression • 243

Une approche holistique
de vos projets • 245

L'importance de la pratique • 247

La question des prérequis
théoriques • 248

FOMO, FOBO
et procrastination • 249

Bien débiter votre projet
no-code • 251

No-code

- Qu'entend-on par projet ? • 251
- Le principe du fail fast • 253
- L'objectif d'une adoption durable • 255
- Le lean start-up comme pierre angulaire • 257**
- Une méthodologie venant de l'industrie • 258
- L'approche lean start-up • 258
- La notion de produit minimum viable (MVP) • 260
- Les cycles d'apprentissage lean • 263
- La solution n'est jamais que technique • 265
- Cadrer un projet • 267**
- Pourquoi cadrer ? • 267
- Simplicité et efficacité • 269
- Les ateliers de conception • 270
- La feuille de route produit • 276
- La gestion de votre backlog • 277
- 8 Organiser, collaborer et automatiser en mode no-code ops 281**
- Des opérations aux no-code ops • 282
- Qu'appelle-t-on « opérations » ? • 283
- L'ère des ops, dans la lignée des DevOps • 284
- Quelques cas de no-code ops • 287
- Le cas de la société Tiller • 287
- Pourquoi automatiser ? • 289**
- Gagner du temps • 290
- Améliorer l'efficacité des processus • 291
- Sécuriser le bon fonctionnement de votre entreprise • 291
- Motiver les équipes • 292
- Comment bien no-coder vos opérations ? • 292**
- Cartographier ses processus • 293
- Exemple de processus et d'automatisation • 293
- 9 Implémenter 299**
- Faire soi-même • 300
- Choisir des outils adaptés • 303
- Ne pas se précipiter sur les outils • 306
- Se former • 307
- Documenter • 309
- Collaborer • 310
- Passage à l'échelle et sécurité • 311
- Et le code dans tout ça ? • 312
- RGPD • 314
- Conclusion**
- Au-delà du no-code 317**
- Au fil des cinq dernières décennies • 317
- Le futur du no-code • 323
- Quel impact sur l'emploi ? • 324
- Favoriser les projets locaux, efficaces et responsables • 326
- Présentation des auteurs • 329

Avant-propos

Lorsque les éditions Eyrolles nous ont proposé d'écrire un livre sur le no-code, notre première réaction a été la réflexion suivante : « le » no-code est une expression qu'on entend souvent, mais en soi, ça n'existe pas vraiment en tant que tel. Il existe clairement un « mouvement no-code ». Cependant, celui-ci se fonde sur une multiplicité d'outils, tous étiquetés « no-code », mais en réalité très variés. Déjà, ils ne servent pas à faire les mêmes choses. Ensuite, ce ne sont pas les mêmes profils qui les utilisent, ne serait-ce que parce que certains sont très accessibles techniquement et d'autres moins. Enfin, l'éventail de leurs contextes d'utilisation est immense : créer soi-même, après quelques jours de formation, des outils internes en no-code pour sa petite PME ou dans un grand groupe, cela n'a rien à voir avec la conception d'une app mobile qu'un porteur de projet confie à une agence no-code !

Rapidement, nous nous sommes dit que si on écrivait ce livre, il faudrait vraiment bien cartographier toutes ces configurations afin de ne pas rester sur une définition un peu flottante et sans relief de notre sujet. Il ne faudrait pas non plus véhiculer une vision trop focalisée sur les lancements d'activités : « je veux lancer ma start-up : soit je fais mon app moi-même, soit je la fais faire par une agence, pour plus vite et moins cher ». C'est souvent ce cas d'usage qui est placé sous le feu des projecteurs, mais comme nous aimons le répéter, cela ne représente que la partie émergée de l'iceberg no-code.

Nous nous sommes aussi immédiatement interrogés sur les **destinataires de cet ouvrage**. Le no-code veut ouvrir l'accès à la programmation à toutes et à tous : voilà une « cible » qui a le mérite d'être aussi précise qu'imprécise, et surtout parfaitement « inactionnable »... Quid de notre livre ? Qui pourrait s'intéresser au no-code ? Nous avons listé des hypothèses variées :

- des **curieux** qui ont été confrontés à ce drôle de mot, « no-code », que l'on voit et entend de plus en plus ;

No-code

- des **utilisateurs d'outils no-code** qui se font déjà une idée du sujet et souhaitent l'approfondir ;
- les **développeurs**, également, qui, dans leur veille permanente, ont envie de se faire leur propre idée du phénomène et de découvrir les outils, en dépassant les seuls discours publicitaires ;
- tous les autres **métiers du Web** (ex. product managers, UX et UI designers, UX writers, data scientists, experts en référencement, en marketing ou en growth hacking). Chacune de ces spécialités réfléchit déjà constamment à ses outils de travail et méthodes de collaboration. Il est clair que le no-code regorge de promesses pouvant les inspirer ;
- les **solopreneurs**, dont le cœur d'activité n'est pas forcément lié au numérique. Ils manquent souvent de temps pour s'informer sur les nouveaux outils, qui pourraient les seconder efficacement dans leurs tâches quotidiennes ;
- des **entrepreneurs** découvrant des solutions d'une efficacité redoutable et utilisables sans être développeur. Ces derniers étant si difficiles à recruter...
- les **décideurs de PME et TPE** qui veulent optimiser leurs processus et outillage interne afin d'accroître leur productivité.
- toutes les personnes souhaitant s'orienter vers des métiers du numérique.

Alexandre, notre interlocuteur chez Eyrolles, était déjà convaincu par cette approche nuancée. Il a immédiatement rebondi sur le fait qu'il ne fallait pas que notre ouvrage ait uniquement une vocation pratique, qu'il soit une sorte de « no-code pour les nuls ». Pour nous, c'était déjà clair. Car nous nous efforçons constamment de prendre du recul sur ce qui se passe et se dit, aussi bien dans les écosystèmes du no-code (ce qu'on appelle « le no-code game ») que dans la vulgarisation des concepts techniques cachés dans les profondeurs de ces outils visuels. Car même si le no-code repose sur la « programmation visuelle », cela demeure de la programmation. Sous le capot de ces outils qui dispensent de savoir lire et écrire du code informatique, on retrouve des notions techniques avec leurs bonnes pratiques associées. Nous voulions aussi proposer des analyses d'ordre plus sociologique, et une étude des avancées technologiques qui ont rendu l'émergence des outils no-code possible.

Au fil de nos échanges, est alors rapidement apparu un point déterminant pour attaquer ensemble cet ambitieux projet d'écriture. Alexandre a soulevé qu'il faudrait faire attention à ne pas verser dans une apologie du no-code, malgré notre enthousiasme pour le sujet. Il faudrait savoir nuancer et proposer un peu d'objectivité.

C'était pour nous une évidence. Car s'il y a bien un parti pris que nous avons dans tout ce que nous produisons au sujet du no-code, c'est celle de toujours avoir une posture enthousiaste, mais critique. Incitatrice mais réaliste. D'une part parce que le discernement est une valeur cardinale de Contournement et d'autre part parce que nous venons du code, parce que nous avons fait de la formation, de la production et du conseil dans le domaine du Web depuis des années. Ne vous inquiétez donc pas : à aucun moment ce livre ne portera de techno-angélisme ou d'idées préconçues dogmatiques, affirmant par exemple que le no-code permettrait, avec ses pouvoirs magiques, de tout faire par soi-même, sans même avoir à se former. Et encore moins des prophéties opportunistes et fallacieuses annonçant que nous n'aurions bientôt plus besoin de professionnels du code ! Nous entendons trop souvent ce genre de raccourcis à l'emporte-pièce, et ce livre représente aussi pour nous une opportunité de remettre les pendules à l'heure.

Afin d'être transparent dans notre démarche, nous avons aussi tenu à éviter certains lieux communs, à abattre quelques clichés et, finalement, à exposer certains de nos partis pris et convictions. Par exemple, sur la bonne approche pour aborder un projet numérique, nous croyons beaucoup aux bénéfices obtenus en croisant les grandes leçons des méthodes agiles, de la méthodologie *lean start-up*, de la culture *maker*, du mouvement des « artisans logiciels » (*software craftsmanship*) et de l'éthique des *hackers*. Il faut du temps pour s'approprier leurs philosophies et savoir faire bon usage de leurs recommandations concrètes. Nous avons voulu, à notre tour, transmettre de bonnes pratiques de conception et de cadrage, pour vous éviter de vous jeter, tête baissée, dans les outils no-code : cette attitude nous paraît vraiment essentielle, voire incontournable !

Après l'élaboration de la structure de l'ouvrage, s'est posée la question de l'écriture du livre en tant que telle : nous devons en effet gérer Contournement et son développement. La rédaction d'un livre de 300 pages en moins de 10 mois nous paraissait difficile sans la collaboration d'un ou une professionnel(le) à qui nous fournirions tout le fond et le contenu afin de le mettre en mots. Ayant remarqué quelques articles que Florian avait écrits en 2021, nous nous sommes tournés vers lui et il s'est révélé bien plus qu'un simple scribe, mais un véritable co-auteur, qui a également contribué sur le fond du livre et sur sa matière.

Le trio était donc constitué, et c'est à partir de notre plan détaillé que le travail a pu se poursuivre, jusqu'à parvenir à un ouvrage structuré en trois temps. Dans une première partie (chapitres 1 à 4), il nous a paru important de nous attarder sur le contexte général de l'émergence du no-code et d'éclairer ce phénomène à travers plusieurs prismes. Nous commençons ainsi par défricher les nombreuses prises de parole provenant des éditeurs d'outils, communautés et médias, afin d'en démêler le vrai du faux ou d'y apporter des nuances qui leur manqueraient

(chapitre 1). Alors seulement exposons-nous le no-code à travers des cas concrets, en racontant « de l'intérieur » les histoires d'une sélection de projets (chapitre 2). À la suite de ce parcours, nous proposons une définition clairement caractérisée de ce que l'expression « outil no-code » délimite, ce qui n'a rien de vraiment *évident* (chapitre 3). Enfin, en s'intéressant à ses origines techniques (chapitre 4), nous inscrivons les outils no-code dans une histoire ancienne, faites de progrès et d'avancées dont il bénéficie à plein. Nous tuons ainsi dans l'œuf l'illusion selon laquelle le no-code aurait surgi *ex nihilo* de manière complètement révolutionnaire.

Dans une seconde partie (chapitres 5 à 6), nous nous sommes concentrés sur les bénéficiaires des technologies no-code, afin de comprendre leurs intérêts et leur engouement pour ces outils. Le phénomène no-code est en effet également porté par de nombreux écosystèmes et communautés d'êtres humains qui échangent avec enthousiasme autour des outils (chapitre 5). No-codeuses et no-codeurs manifestent un esprit d'entraide, un goût pour l'action concrète et un sens du partage. Nous avons voulu présenter ces « valeurs » et réfléchir à une « culture » du no-code en mettant en avant trois personnages (fictifs mais inspirés de personnes réelles) et, à travers eux, certaines dispositions qui nous paraissent souhaitables pour pratiquer le no-code (chapitre 6).

La troisième et dernière partie (chapitres 7 à 9) est justement orientée vers cette pratique du no-code. Le no-code est aussi (et surtout !) une forme d'art ou d'artisanat : pour en prendre le coup de main, il faut passer à l'action et s'y exercer. Nous y formulerons nos recommandations pour vous guider sur différents thèmes : comment bien débiter en no-code et comment lancer son produit ou service (chapitre 7), comment bien mener ses opérations (ses « no-code ops ») en concevant des outils de collaboration efficaces (chapitre 8) et comment implémenter efficacement tout type de projet, tant sur un plan technique que méthodologique (chapitre 9).

Le cadre étant posé, voici quelques points supplémentaires que l'on aimerait spécifier ici.

Le no-code est un sujet gigantesque par ses implications : au cours de l'écriture du livre, nous avons renoncé à en développer certains aspects, afin de garantir une bonne lisibilité.

Ainsi, nous avons choisi de ne pas détailler l'utilisation d'une sélection d'outils et de leurs fonctionnalités. L'écosystème qu'ils constituent évolue très vite et nous avons dû éviter les sujets, références et données susceptibles de rapidement devenir obsolètes. Par exemple, en raison de leurs mises à jour fréquentes. Toutefois, l'objet de ce livre n'est pas de former aux outils no-code. Pour s'initier concrètement, il y a quantité de contenus (dont de nombreux gratuits)

disponibles sur le Web. Pour aller plus loin dans leur maîtrise, rien ne vaut une formation.

Bien que cela constitue une partie du champ d'activité de Contournement, nous avons décidé de ne pas beaucoup développer le recours au no-code par les grandes entreprises, pour plusieurs raisons. Difficile, déjà, d'exposer leurs stratégies d'implémentation sans détailler leurs historiques et leurs cadres particuliers (organisation opérationnelle, stack technique, contexte juridique, gouvernance). Raconter quelques cas typiques nous aurait peut-être donné bonne conscience, mais qu'auraient-ils réellement représenté de l'immensité des configurations variées des grandes structures ? Il faudrait consacrer un livre entier à ce sujet ! Cependant, comme souvent en matière de technologies, les start-up et petites structures sont à l'avant-garde des usages, ayant une plus grande marge de manœuvre pour bâtir leurs systèmes et choisir leurs outils. Ainsi, nous ne parlons que peu de certains outils no-code plus orientés « grandes entreprises » (comme Power Apps de Microsoft, ou Unqork, et toute la famille des outils low-code). Mais peut-être une version future de l'ouvrage sera-t-elle l'occasion d'approfondir ces points ?

Notre conviction chez Contournement est que la grande nouveauté induite par le no-code est la chance pour des profils non-techniques de pouvoir construire eux-mêmes leurs propres solutions numériques, notamment en se formant et en étant un minimum accompagné. Nous avons toutefois tâché de ne pas donner une place trop importante à la formation dans cet ouvrage, et de représenter également le champ de la production en no-code, qui constitue une grande partie des professionnels.

Nous souhaitons enfin mentionner la question de l'écriture inclusive, d'autant plus importante que le no-code, comme le code (un peu moins peut-être), est encore occupé par une majorité d'hommes. Chez Contournement, nous avons décidé d'utiliser au maximum l'écriture inclusive, notamment pour désigner les fonctions où les stéréotypes de genre sont les plus excluants (ex. no-codeur·se, développeur·se, expert·e). Nous avons réfléchi avec Eyrolles à la meilleure option pour un ouvrage long de plusieurs centaines de pages, et finalement opté pour des tournures de phrases qui matérialisent cette dimension inclusive.

Le contexte et la posture des auteurs étant ainsi établis, il ne nous reste qu'à vous souhaiter une bonne entrée dans l'univers des outils no-code – qui pourrait bien vous amener à envisager de manière complètement différente votre rapport au numérique, comme cela l'a fait à beaucoup d'autres ces dernières années.