

PAR
MARIE-LAURE
CAHIER
ET
SUZY
CANIVENC

COMMENT LES USAGES DU NO CODE IMPACTENT LE TRAVAIL

SOMMAIRE

P. 2 Le No Code, c'est quoi ?

P. 3 La promesse technophile
du No Code

P. 4 Les bénéfices attendus du No Code
pour les organisations

P. 5 L'appropriation du No Code :
un processus largement informel et
moins simple qu'annoncé

P. 7 Des précautions à prendre face à
des réticences légitimes

P. 10 Des remaniements substantiels
dans l'organisation du travail

P. 14 Conclusion : des potentialités
réelles mais une extension sans doute
limitée

Les outils no-code portent la promesse d'une démocratisation de la création de produits numériques à travers toutes les organisations, y compris chez des personnels non techniques. Aussi séduisante qu'elle apparaisse, cette promesse paraît nettement surestimée, car de nombreuses barrières à l'appropriation existent chez les utilisateurs potentiels. En revanche, une étude de terrain menée par la chaire FIT² montre que l'usage de ces outils a la capacité d'entraîner des reconfigurations substantielles de l'organisation du travail vers plus d'agilité, d'autonomie et de coopération, à condition de s'ancrer dans un terrain social favorable et de prendre en compte les réticences « métiers » des acteurs de l'IT. Comme toute technologie, le No Code n'actualisera ses potentialités que dans les configurations organisationnelles préalablement préparées à le recevoir.

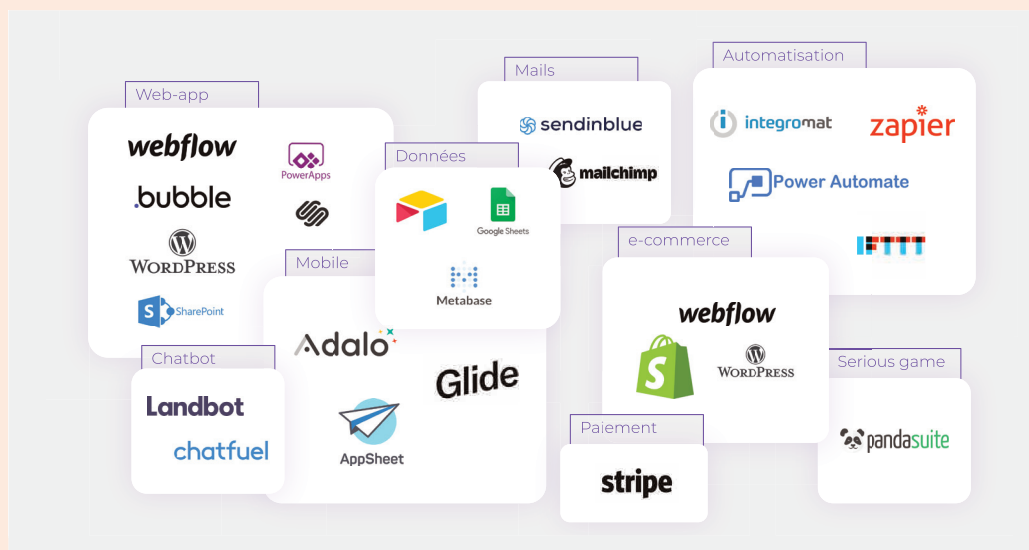
LE NO CODE, C'EST QUOI ?

Le No Code renvoie à une méthode de réalisation de produits numériques (sites web, applications, jeux vidéo, chatbots, automatismes, etc.) ne nécessitant aucune connaissance poussée en programmation informatique grâce à un environnement de développement basé non plus sur l'écriture de lignes de code en langage expert (Java, C++, Ruby, Python, etc.) mais sur des interfaces visuelles simplifiées : les lignes de codes sont déjà préprogrammées dans des composants (qui ont chacun une fonction) qu'il suffit de « glisser-déposer » (*drag and drop*) ou de relier entre eux, un peu comme des briques de Lego¹. Le No Code a vocation à être pris en main par des non-développeurs, contrairement au Low Code qui, lui, nécessite des connaissances en programmation. Notons que, dans de nombreux médias, low-code et no-code sont traités de façon globale et indifférenciée, rendant parfois peu lisible le potentiel respectif de ces deux approches.

Le No Code renvoie à une méthode de réalisation de produits numériques ne nécessitant aucune connaissance poussée en programmation informatique grâce à un environnement de développement basé sur des interfaces visuelles simplifiées.

Aux outils précurseurs du No Code comme WordPress, Wix ou Dreamweaver pour la création de sites web s'ajoutent aujourd'hui de multiples plateformes no-code (voir figure 1) qui permettent chacune de réaliser certains types d'applicatifs – un chatbot, un site e-commerce, un serious game, une web-application, mais aussi des automatisations de workflow, etc. – et dont les solutions peuvent aussi être combinées entre elles. Sur un plan technique, la sophistication des fonctionnalités que ces plateformes

FIGURE 1 - PANORAMA DE QUELQUES PLATEFORMES NO CODE



1. Antoine Crochet-Damais, « No code / low code : c'est quoi ? », *JDN*, 15 décembre 2021.

peuvent permettre d'atteindre fait encore largement débat entre partisans et détracteurs du No Code. Certains professionnels du code contestent la capacité de ces solutions à produire des applications qui soient à la fois « maintenables, sécurisées, performantes et scalables² ». De leur côté, les zélotes du No Code n'hésitent pas à affirmer qu'avec de la pratique, « *on peut aller aussi loin qu'avec le code*³ ». Le présent article n'a pas vocation à prendre parti dans ce débat technologique, mais à exposer les arguments des uns et des autres, dans un contexte où les pratiques no-code tendent à s'inscrire dans les organisations.

De multiples plateformes no-code permettent chacune de réaliser certains types d'applicatifs dont les solutions peuvent aussi être combinées entre elles.

Les plateformes no-code peuvent être propriétaires et payantes, ou encore en open source. Elles peuvent être le fait de start-up mais aussi de géants du numérique.

LA PROMESSE TECHNOPHILE DU NO CODE

Le No Code porte en lui une promesse de démocratisation du numérique qui l'apparente à un mouvement social en émergence⁴. Son fonctionnement n'est pas sans rappeler celui des communautés autour des logiciels libres, de l'open source ou des *makers*⁵.

Le No Code s'inscrit ainsi dans un long mouvement historique de simplification et d'accessibilité du numérique, commencé dans les années

1970-80 avec la micro-informatique, puis l'amélioration de l'interface utilisateur (clavier, souris, bureau, logiciels de textes et de calculs), enfin avec l'Internet et le « Web participatif » (wikis, blogs, réseaux sociaux). Après l'usage des outils informatiques et la création de contenus sur Internet, c'est aujourd'hui au tour de la création de logiciels d'être concernée par ce mouvement de démocratisation qui culmine avec l'émergence de la figure du « citoyen développeur ». Ce n'est alors rien de moins qu'une véritable révolution sociopolitique qui est annoncée : « *la no code permet de démocratiser et rendre accessible le développement et la création au plus grand nombre, et nous sommes convaincus que cette tendance apporte un réel bénéfice à la société, à la liberté d'entreprendre et à l'égalité des chances* »⁶. La rhétorique de l'émancipation sociale est une figure classique qui accompagne le développement des TIC depuis son origine⁷. « Tous makers ! », tel est l'horizon proposé par les promoteurs du No Code.

Nous allons montrer que les usages professionnels réels et leurs effets sur les organisations sont plus nuancés que ne le laissent entendre les discours promotionnels enthousiastes qui accompagnent le développement de ce mouvement.

Le No Code s'inscrit ainsi dans un long mouvement historique de simplification et d'accessibilité du numérique.

2. Blog de bluetouff <https://bluetouff.com/2022/01/06/nocode-ou-la-promotion-de-lilletterisme-numerique/>

3. Les verbatim issus de notre étude de terrain figurent en italique.

4. Parmi les « communautés no-code », on peut citer des plateformes ou associations comme Ottho, No Code France, NoCodeStation, Contournement, ou encore NoCode for Good au service de projets solidaires et environnementaux. Voir <https://usbeketrica.com/fr/article/le-no-code-nouveau-sesame-pour-les-non-inities-a-la-programmation>

5. Isabelle Berrebi-Hoffmann, Marie-Christine Bureau, Michel Lallement, *Makers : Enquête sur les laboratoires du changement social*, Seuil, 2018.

6. <https://ottho.fr/blog/le-no-code#wysiwyg>, Consulté le 15 octobre 2021.

7. La première techno-utopie émerge dès les années 1940, avec la « cybernétique » de Norbert Wiener (1948).

LES BÉNÉFICES ATTENDUS DU NO CODE POUR LES ORGANISATIONS

D'un point de vue technique et économique, le No Code offre aux entreprises la promesse de baisser les coûts de développement et d'accélérer les temps de production (et donc la livraison/mise en marché) de prototypes, produits numériques simples ou encore automatismes de process, dans un contexte de transformation digitale accélérée. Qui n'a pas, en effet, été confronté à des situations où, pour la réalisation d'une application métier, les coûts, les délais annoncés et les objections de conception soulevées par les DSI ou les fournisseurs de solutions, sont tels qu'ils finissent par décourager toute velléité d'action ? Comme l'indique un partisan du No Code : « *Arrive un moment où on a imaginé un super concept, et maintenant il faut le créer. On va voir nos équipes Tech et là, la première chose qu'elles nous disent c'est : "ça va coûter 100 000 euros et ça va prendre 6 mois". En no-code, c'est fonctionnel, ça marche, ça ne prend pas 6 mois et ça ne coûte pas 100 000 euros.* » L'alternative no-code permettrait donc de reconfigurer les rapports de pouvoir entre les métiers et les experts de l'IT, en redonnant aux premiers des marges de manœuvre dans la négociation avec les seconds, et en faisant reculer « l'effet boîte noire » dont jouissent habituellement les détenteurs du pouvoir informatique.

En outre, les outils no-code représenteraient une solution face à la pénurie de développeurs, métier qui reste en forte tension⁸. Milind Govekar, vice-président chez Gartner, a ainsi prédit, lors de l'IT Symposium Xpo 2021, que d'ici à 2025, 70 % de nouvelles applications seront développées à partir de low-code ou no-code : « *Applications of the future will be assembled and composed by the people that actually use them.*⁹ »

Le No Code est donc vu au premier chef comme un booster de productivité, d'agilité et d'efficacité. Une étude américaine de Salesforce¹⁰, portant sur la perception des utilisateurs d'automatismes, pointe également un autre avantage : ces outils représenteraient un accélérateur de collaboration inter-équipes mais aussi inter-départements (voir figure 2). En décloisonnant la création logicielle hors des différents silos où elle est habituellement cantonnée (DSI, Digital Factories ou ESN¹¹), les outils no-code favoriseraient une montée en compétences des collaborateurs et leur autonomisation, en même temps qu'une meilleure coopération autour des enjeux IT. Par exemple : dans le cadre de sa transformation digitale de grande ampleur, une multinationale d'origine française dans le domaine des produits de grande consommation, a pris le parti d'initier massivement ses salariés aux solutions no-code, afin qu'ils puissent monter en compétences, automatiser certaines tâches et créer eux-mêmes des applications pour leur propre usage. Le projet est porté à la fois par la Digital Factory (côté IT) et par les Ressources humaines en charge de « l'expérience collaborateurs ». L'ambition affichée est énorme : en 2023, 20 % des applications-maison devraient ainsi être réalisées en no-code, soit 1 outil sur 5.

Les outils no-code favoriseraient une montée en compétences des collaborateurs et leur autonomisation, en même temps qu'une meilleure coopération autour des enjeux IT.

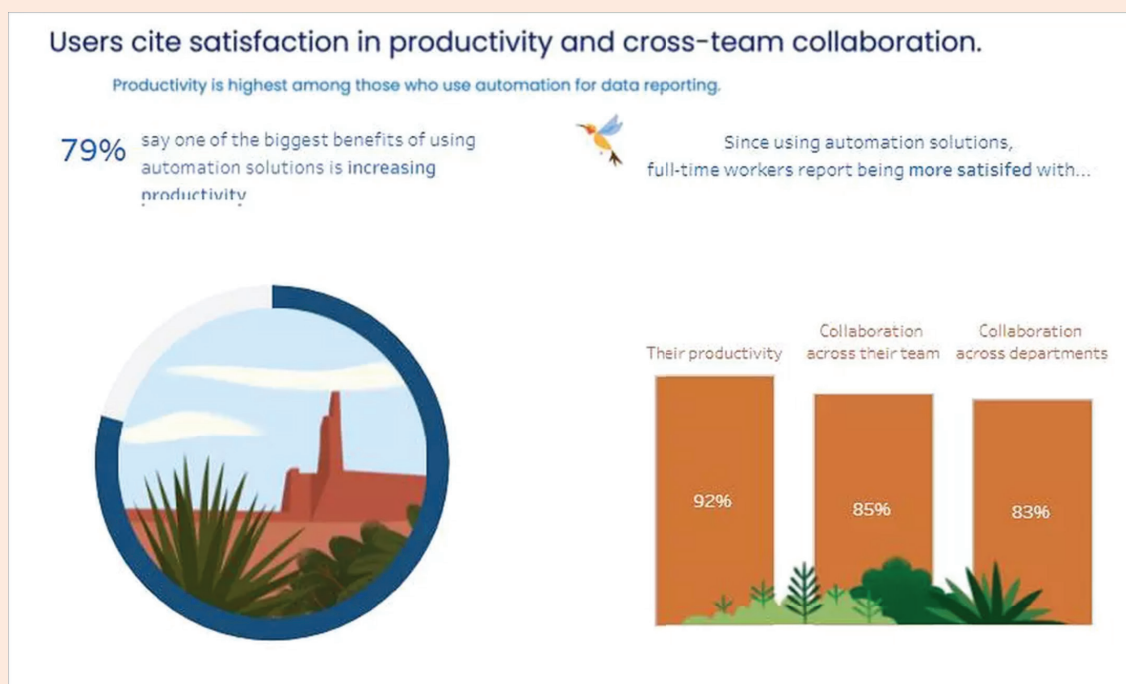
8. Selon l'enquête BMO 2021, Pôle Emploi-CREDOC, la part des recrutements de techniciens de développement informatique jugés difficiles s'élève à 60,5 % <https://statistiques.pole-emploi.org/bmo/bmo?fe=M1Z80&la=0&pp=2021&ss=1> et à 65,6 % pour les ingénieurs informatiques.

9. <https://www.gartner.com/en/newsroom/press-releases/2021-11-10-gartner-says-cloud-will-be-the-centerpiece-of-new-digital-experiences> et https://www.theregister.com/2021/11/10/gartner_it_symposium/ Consultés le 16 janvier 2021.

10. Étude menée par Salesforce en octobre 2021 auprès de 773 usagers d'automatismes aux États-Unis. <https://www.salesforce.com/news/stories/new-salesforce-research-links-lower-stress-levels-and-business-automation/>

11. Entreprises de services du numérique.

FIGURE 2 - INDICATEURS ISSUS DE L'ÉTUDE SALESFORCE



Déterminer si les outils no-code produiront les bénéfices escomptés nécessite de mieux cerner les processus sociotechniques sous-jacents qui accompagnent l'appropriation de ces outils et leurs effets sur l'organisation du travail. Pour ce faire, nous nous appuyons sur une étude de terrain menée par la chaire FIT² au sein d'une entreprise de conseil spécialisée dans la transformation numérique des organisations. Des entretiens semi-directifs ont été réalisés auprès d'un panel diversifié de salariés utilisateurs et non utilisateurs de no-code (designer, développeurs, consultant-chef de projet, directeur des opérations, responsable technique), puis complétés par des entretiens hors de cette organisation auprès de quelques développeurs. Cette étude révèle : i) que l'appropriation du No Code est moins simple qu'il n'y paraît et assez limitée, même dans une entreprise très sensibilisée au numérique ; ii) que des précautions sont à prendre lors de l'implantation de ces outils, car

ils peuvent venir heurter des fiertés métiers et des pouvoirs en place ; iii) que ces outils peuvent, à certaines conditions, favoriser l'émergence de nouvelles pratiques de travail et d'organisation.

L'APPROPRIATION DU NO CODE : UN PROCESSUS LARGEMENT INFORMEL ET MOINS SIMPLE QU'ANNONCÉ

Cette entreprise, fortement orientée vers le numérique, compte 350 salariés, dont des designers, des créatifs, des consultants, des marketeurs, des analystes et data scientists, des fonctions supports ainsi que des développeurs (*les dev* dans le jargon de l'organisation), ces derniers représentant près d'un tiers de l'effectif.

Elle présente des traits d'organisation qui l'apparentent à la fois à une configuration « professionnelle » et « innovatrice »¹², les deux se conjuguant pour réduire le besoin d'une ligne hiérarchique longue, pour favoriser l'autonomie des salariés et les ajustements mutuels informels. Toutefois, cette forte autonomie des acteurs se traduit aussi par un certain « vide » de process et d'organisation, qui contraint les individus et les équipes à « se débrouiller » pour trouver eux-mêmes les ressources et les moyens permettant d'atteindre les objectifs qui leur sont assignés (i.e. livrer vite et bien les projets commandés par les clients). Cette capacité à « se débrouiller » est valorisée et assumée par l'organisation.

Dans ce contexte, les outils no-code ont tout d'abord été investis par quelques salariés de manière officieuse, grâce à la large part d'autonomie dont ils jouissent. Ces salariés défricheurs de No Code ont un profil assez précis : sans être des experts du développement, ils ont une grande appétence pour le numérique et ont déjà mis « les mains dans le cambouis », souvent via une précédente expérience entrepreneuriale ou associative, afin de créer des sites web ou de petites applications. La plupart d'entre eux ont donc pratiqué des outils no-code bien avant que ce terme n'existe et ne devienne à la mode : « À l'époque, on ne parlait pas de no-code, ni nous, ni le marché. » Les utilisateurs de No Code semblent ainsi présenter un profil intermédiaire entre des « non techniciens » qui n'ont pas d'appétence particulière pour ces outils numériques et les développeurs pour qui le développement de logiciels est un métier nécessitant des compétences spécialisées. Ils ont en général un tempérament curieux et tenace : ils aiment chercher, tâtonner, expérimenter, car c'est ainsi que ces outils livrent tout leur potentiel.

Les utilisateurs de No Code semblent présenter un profil intermédiaire entre des « non techniciens » qui n'ont pas d'appétence particulière pour ces outils numériques et les développeurs pour qui le développement de logiciels est un métier.

Dans un premier temps, la diffusion au sein de l'organisation s'est faite de façon informelle et virale. Elle semble avoir été facilitée par la préexistence de relations fortes entre certains individus et par la fréquence de leur présence sur site (versus ceux qui télétravaillent parmi lesquels la sérendipité semble moindre) : « On geek chacun dans notre coin, chacun trouve de nouvelles options et va le dire à voix haute dans l'open space, ce qui va attiser la curiosité de quelqu'un d'autre qui va venir un peu regarder : on est un peu dans cette émulation-là, encore un peu physique, un peu ancienne école, en travaillant les uns à côté des autres. » Petit à petit, les réseaux interpersonnels commencent à s'inscrire dans une logique organisationnelle : « Il y a une rencontre qui s'opère entre une curiosité naturelle et un besoin potentiel pour un projet, puis deux, puis trois. Ça a fait comme un effet boule de neige. » Découvert par des « pionniers » souvent dans un cadre extérieur à l'entreprise et sur la base de leur propre curiosité, le No Code va être investi progressivement à des fins professionnelles liées à l'organisation, jusqu'à culminer par la création d'une équipe dédiée au No Code, structurée en centre de profit au service des clients mais aussi des besoins internes.

Pour autant, même dans cette organisation fortement orientée numérique, et malgré une volonté et des efforts d'évangélisation, tous les collaborateurs n'ont pas envie de se saisir de ces solutions, principalement par manque d'intérêt ou de temps à y consacrer : « On dit que c'est simple, qu'on l'a tout de suite en mains, mais c'est faux. Il faut passer pas mal de temps sur l'outil

12. Ces termes correspondent à la classification de Henry Mintzberg in *Structure et dynamique des organisations*, Editions d'Organisation, 1986.

pour apprendre à le maîtriser (...) ». Un de nos enquêtés indique, par exemple, que l'équipe RH sollicite souvent l'équipe no-code pour la réalisation d'applications internes, mais préfère se faire livrer les solutions clés en main plutôt que de s'emparer des plateformes permettant de les créer.

Tous les collaborateurs n'ont pas envie de se saisir de ces solutions, principalement par manque d'intérêt ou de temps à y consacrer.

Si certaines plateformes peuvent permettre d'obtenir en quelques heures des résultats jugés gratifiants, d'autres en revanche nécessitent un apprentissage plus long, souvent parce qu'elles permettent de développer des fonctionnalités plus complexes. Plus la plateforme permet de produire des résultats puissants, plus elle va nécessiter un investissement important de la part de l'utilisateur. L'appropriation des outils no-code implique donc une motivation et de la patience pour y consacrer de longues heures d'apprentissage. De plus, si les plateformes no-code sont conçues pour s'adresser prioritairement aux non-développeurs, elles restent développées par des codeurs et sont donc imprégnées de leur logique particulière : l'algorithmie¹³. Disposer de bases techniques et de réflexes de programmation représente donc un atout non négligeable pour accélérer l'appropriation de ces outils – un fait qui est souvent minimisé par les évangélistes du No Code.

Plus la plateforme permet de produire des résultats puissants, plus elle va nécessiter un investissement important de la part de l'utilisateur.

Au vu des barrières techniques et de temps d'apprentissage qui peuvent exister même dans une entreprise fortement orientée vers le numérique, la promesse du « tous no-codeurs » mérite d'être fortement nuancée.

DES PRÉCAUTIONS À PRENDRE FACE À DES RÉTICENCES LÉGITIMES

La promesse de démocratisation de la création logicielle au sein des entreprises s'attaque à deux bastions : les développeurs qui y voient une menace pour leur métier, et les DSI pour leur pouvoir institutionnel.

Ces deux populations doivent donc être particulièrement prises en compte, lors de l'implantation et du déploiement d'outils no-code. Leurs objections méritent d'être écoutées et traitées ; une attitude ouverte et bienveillante permettra d'amoindrir les inquiétudes et les attitudes de rejet, voire d'enrichir la démarche via des formes de « dispute professionnelle¹⁴ » : « *Je trouve ça bien aussi qu'il y ait des personnes un peu méfiantes parce que ça ouvre les discussions* » indique le chef du projet no-code.

LES RÉTICENCES ET INQUIÉTUDES DES DÉVELOPPEURS

Dans notre entreprise-terrain, la philosophie promue consiste à ne pas opposer code et no code, et à souligner au contraire leur complémentarité. Selon les salariés interrogés, les outils no-code servent les intérêts des non-développeurs autant que ceux des développeurs. Non seulement ces derniers sont les mieux placés techniquement pour tirer parti de ces outils, mais en outre le No Code vient faciliter leur travail et l'accélérer (certains l'utilisent déjà sans forcément le crier sur les toits). En réduisant la part de réalisation de tâches ingrates et répétitives qui alourdissent et allongent les projets, le No Code

13. Ensemble des règles mobilisées pour la conception d'algorithmes de programmation.

14. Clot, Yves (2014). « Réhabiliter la dispute professionnelle », *Le Journal de l'École de Paris du management*, 105 (1), 9-16.

permet aux développeurs de se concentrer sur les activités les plus enrichissantes, complexes et créatives : « *Je constate, personnellement, que je ne traite plus aucun sujet répétitif. Il y a tout un tas de fonctionnalités qui sont très bien prises en charge par la plateforme et je ne me concentre plus que sur la valeur ajoutée qui n'est pas accessible en no-code* » dit l'un d'entre eux. Le No Code permet ainsi de recentrer les développeurs sur le but ultime de leur métier qui n'est pas forcément de faire du code mais de livrer des produits fonctionnels répondant aux besoins des clients : « *On ne va plus cramer du budget et du temps de développeur pour faire des trucs standardisés et pas très intéressants, on va utiliser les développeurs sur ce qui est différenciant.* ».

Les réticences et inquiétudes de nombreux développeurs semblent concerner leur fierté « métier » associée à la détention de compétences spécifiques et non substituables.

Toutefois si cette entreprise s'inscrit dans une perspective de conciliation des pratiques, il ne faut pas sous-estimer, là comme ailleurs, les réticences et inquiétudes de nombreux développeurs. Celles-ci semblent moins concerner l'avenir de leur emploi que leur fierté « métier » associée à la détention de compétences spécifiques et non substituables¹⁵. Parmi six développeurs interrogés à titre de groupe de contrôle hors de cette organisation, cinq disent soit ne pas connaître les outils no-code, soit ne pas vouloir les utiliser, l'un d'eux affirmant même les « fuir ». Un seul déclare les utiliser parce qu'ils lui sont imposés par son employeur dans un cadre précis (des automatisations de tests). Ils arguent que ces outils leur paraissent soit trop limités techniquement pour répondre à leurs besoins d'expert, soit que leurs routines profes-

sionnelles leur permettent déjà de gagner du temps (par ex. réutiliser un ancien morceau de code ou aller chercher celui dont on a besoin sur le net). Ces outils ne font pas « sens¹⁶ » pour eux et ils ne souhaitent pas y recourir, allant même jusqu'à « *recoder des choses qui peuvent être [réalisées] automatiquement* ». Loin de voir dans le No Code un enrichissement via l'automatisation des tâches rébarbatives, ils y voient un appauvrissement de leur métier et une négation de leur plus-value.

Dans notre entreprise-témoin, d'autres objections s'expriment : le manque de fiabilité, de robustesse et de sécurité des outils créés en no-code, la minimisation de ce qu'il est possible de réaliser par ce moyen (selon certains, le no-code peut servir au mieux à prototyper avant de passer à l'échelle en code), ou encore le rôle de pompier que les développeurs auront à assumer pour réparer les dégâts créés par le No Code. En effet, lorsque les applications créées en no code finissent par concaténer une multitude d'outils différents, les professionnels du code sont souvent les seuls à pouvoir démêler les fils lorsqu'un problème survient. Ce sont eux aussi qui seront sollicités pour intégrer du code quand on voudra pousser plus loin les fonctionnalités de l'outil initial. Ils devront alors réparer la maison ou imaginer son extension, mais sans disposer du plan de construction, car les no-codeurs n'ont souvent pas le réflexe de créer une documentation qui aiderait les développeurs à comprendre comment le produit a été fait. « *Je pense que la défiance vient beaucoup de là*, explique un ingénieur informatique plutôt favorable au No Code. *Les développeurs ont cette arrière-pensée qu'à un moment ou un autre, il y aura quelqu'un qui devra remettre au propre tout ça et que ce sera sûrement pour leur pomme. Si, quand on ouvre le capot de la boîte noire, on constate que le truc est un fatras de tout un tas de choses empilées impossibles à maintenir en l'état, alors le risque est que le bénéfice du temps gagné au départ se révèle être un jeu de dupes.* »

15. Philippe d'Iribarne, *La Logique de l'honneur*, Seuil, 1989.

16. Marie Benedetto-Meyer, Anca Boboc, *Sociologie du numérique au travail*. Paris : Armand Colin, 2021.

C'est toute la promesse de départ (réduction des délais et des coûts) qui pourrait alors être remise en cause.

« Les développeurs ont cette arrière-pensée qu'à un moment ou un autre, il y aura quelqu'un qui devra remettre au propre tout ça et que ce sera sûrement pour leur pomme. »

Une manière de rassurer les développeurs sera d'accompagner les « no-codeurs » par des formations à l'algorithmie et à la culture informatique. Parmi les compétences qui faciliteront un usage raisonné et professionnel du no-code : « la logique des interfaces graphiques, les grands types de traitement (gestion de dates, de listes, de variables, de conditions, de boucles, etc.), le fonctionnement d'une base de données, le principe de gestion de versions, la notion d'API pour gérer les échanges avec d'autres applications...¹⁷ »

Il pourra être aussi utile de mixer des équipes code et no-code sur les projets, afin qu'elles s'acculturent mutuellement, plutôt que de les séparer. Pour chaque projet précis, on s'interrogera sur ce qui peut être fait en code et en no-code et les bénéfices que chaque option procure au projet. Associer des *dev* à un projet réalisé en no-code permettra en outre d'assurer un cadrage technique à la gestion du projet.

Il pourra être utile de mixer des équipes code et no-code sur les projets, afin qu'elles s'acculturent mutuellement, plutôt que de les séparer.

LA MÉFIANCE DES DSI

Les DSI mettent en avant des risques recevables liés au No Code, en termes de cohérence d'infrastructure numérique, de coûts d'intégration, de sécurité, etc. Pour eux, la balance coûts-bénéfices n'apparaît pas forcément évidente. Mais ces arguments pourraient aussi dissimuler d'autres inquiétudes, plus directement liées à la mise en cause de leurs prérogatives.

Les DSI mettent en avant des risques en termes de cohérence d'infrastructure numérique, de coûts d'intégration, de sécurité.

Le No Code n'est en fait qu'un nouvel épisode de la querelle des anciens et des modernes qui se joue entre deux logiques orthogonales : d'une part, l'informatique historique, dite aussi héritée (*legacy*), verticale, centralisée et à juste titre obsédée par la protection et la continuité des systèmes et, d'autre part, l'informatique digitale, orientée réseau, horizontale et décentralisée. De nombreux dirigeants d'entreprise ont réagi à cette situation en créant deux structures parallèles : la DSI historique pour l'exploitation et une structure dédiée à l'innovation digitale pour l'exploration¹⁸. Dans ce contexte déjà compliqué, le No Code viendrait ajouter une couche de difficultés supplémentaire, en incitant à la prolifération incontrôlée de « petits » systèmes d'origines variées. On comprend que les DSI ne se réjouissent guère de voir arriver ce qui s'apparente à une nouvelle menace. « Pour schématiser, j'imagine les sueurs froides des gars quand ils pensent à toute la donnée de l'entreprise qui part sur tout un tas de plateformes hébergées on ne sait trop où. » Non seulement le No Code pourrait entamer encore davantage leurs prérogatives, mais plus encore les DSI pourraient être tenues pour responsables de ses éventuels méfaits.

17. Antoine Crochet-Damais, « Six conseils pour déployer une stratégie no code », *JDN*, 10 janvier 2022.

18. Christophe Deshayes, *La transformation numérique et les patrons*, Les Docs de La Fabrique, Presses des Mines, 2019, p. 98 ss.

Toutefois, plutôt que de se battre contre un mouvement qui va prendre de l'ampleur – au risque de se faire dépasser par une concurrence moins frileuse ou du *shadow IT*¹⁹ non maîtrisé – les DSI auront intérêt à prendre les devants pour en piloter le déploiement, en se positionnant comme des facilitateurs : conseiller les meilleures plateformes à adopter en fonction des cas d'usage, mettre en place des formations ou des communautés d'experts qui seront autant de ressources auxquels les no-codeurs pourront faire appel. « *Il y a des personnes qui sont habituées historiquement à jauger les technologies et à les acheter. Il faut mettre cette capacité au service de l'environnement no-code*, indique un directeur technique. *Par exemple, ce sont les professionnels des SI qui vont s'intéresser au lieu d'hébergement des données ou à la pérennité de la boîte qui propose la plateforme no-code.* » Une approche facilitatrice de la part des DSI pourrait leur permettre de redorer leur image et de créer une véritable logique partenariale avec les métiers. Il faut donc veiller à les embarquer dans le mouvement, car il sera plus vertueux de faire avec elles que sans.

« Il y a des personnes qui sont habituées historiquement à jauger les technologies et à les acheter. Il faut mettre cette capacité au service de l'environnement no-code. »

DES REMANIEMENTS SUBSTANTIELS DANS L'ORGANISATION DU TRAVAIL

Le No Code impacte les frontières de certains métiers mais également les relations que ceux-ci entretiennent entre eux, pouvant provoquer des remaniements substantiels dans l'organisation du travail.

HYBRIDATION DES MÉTIERS, RÉDUCTION DES DIVISIONS HORIZONTALE ET VERTICALE DU TRAVAIL

Le No Code provoque une hybridation de métiers auparavant distincts : les métiers non techniques intègrent désormais une dimension technique et inversement, permettant au passage d'enrichir les tâches de chacun et d'accroître aussi potentiellement les capacités de création et d'innovation. Cet enrichissement des tâches est promu par tous les nouveaux modes de management (de l'école de relations humaines presque centenaire au jeune mouvement des entreprises libérées).

Le No Code provoque une hybridation de métiers auparavant distincts : les métiers non techniques intègrent désormais une dimension technique et inversement, permettant d'enrichir les tâches de chacun.

Dans notre entreprise-témoin, cette hybridation est perçue positivement chez certains designers comme chez certains développeurs. Côté design, « *avec le No Code, je peux me positionner dans des secteurs créatifs avec un peu une étiquette de technicien* » : cette acquisition de compétences est considérée comme attractive, car elle élargit le champ des opportunités professionnelles. Côté développeur, « *Je passe beaucoup plus de temps à discuter véritablement du produit et de ses enjeux avec des designers, avec le client, avec des contributeurs, avec des populations qui ont des sensibilités vraiment différentes. Il y a plus de variété de réponses que quand on a une équipe orientée uniquement développeurs qui pense tout de suite fonctionnalités et moyens de les réaliser, et qui est bridée par la faisabilité* ».

19. Cette expression renvoie à des systèmes informatiques créés de manière artisanale par des salariés en utilisant des logiciels et techniques non conventionnels.

Cette nouvelle façon de travailler prend ainsi le contre-pied de la division horizontale des tâches (consistant à décomposer l'activité productive en multiples activités simples réparties sur les différents salariés) qui crée des silos. « Hier, le designer créait une maquette, puis la passait à une autre personne en charge de la transformer en produit numérique, qui la modifiait sans toujours expliquer pourquoi. C'était des étapes assez étanches, et finalement ça amenait tout un tas de petits problèmes. Maintenant, le designer crée la maquette et puis, il va plus loin et il crée aussi tout le site Internet. Il n'a même plus forcément besoin des développeurs sauf pour des fonctionnalités vraiment complexes ». Cette situation pourrait cependant entraîner un risque d'individualisation à outrance des tâches au détriment du sentiment d'interdépendance, si des salariés détiennent la capacité à prendre intégralement en charge l'ensemble d'un processus.

Le No Code est aussi susceptible de faire bouger les lignes de la division verticale du travail, c'est-à-dire la distinction entre ceux qui conçoivent et ceux qui exécutent. La conception des produits numériques devient ainsi plus collégiale et pluridisciplinaire.

Si nous n'avons mentionné ici que quelques métiers au cœur de l'activité de notre entreprise-témoin, le No Code a la capacité d'impacter tous les métiers, particulièrement ceux qui sont de gros utilisateurs de tableurs intelligents et de données : contrôleurs de gestion, comptables, marketeurs et commerciaux, services RH, car ils sont très concernés par les « workflow automation », ceux-ci étant un champ privilégié pour les applications en no-code.

AMÉLIORATION DES COOPÉRATIONS

La capacité qu'ont les outils no-code d'hybrider les activités facilite la compréhension et le dialogue entre les métiers et l'IT au profit d'une meilleure ambiance de travail et d'un travail de meilleure qualité.

Le No Code aide, en effet, chaque métier à mieux comprendre les préoccupations et les contraintes de ceux avec lesquels il est amené à collaborer. Cela développe des capacités de coopération, réduit la tentation de se rejeter mutuellement les fautes des dysfonctionnements, et permet *in fine* de réaliser de meilleurs produits numériques. Côté développeur, « Tout d'un coup, le design arrive à comprendre pourquoi le développeur parle de la maintenance de demain. Alors qu'avant, il n'en avait pas conscience. Donc, ça augmente le plus grand dénominateur commun entre les différents métiers ». Côté design, « la casquette de "bébé développeur" nous rend meilleur quand les projets sont gros et qu'on a des rôles vraiment distincts : on comprend mieux pourquoi ils [les devs] nous demandent de faire des choses, ce n'est plus quelque chose d'abstrait ».

Le No Code aide chaque métier à mieux comprendre les préoccupations et les contraintes de ceux avec lesquels il est amené à collaborer.

Quelques-uns de nos interviewés entrevoyent cependant un petit risque de scission progressive entre les utilisateurs de No Code et les autres, susceptible de créer de nouvelles hiérarchies d'expertises : « Il y a des petites écoles qui se créent. Maintenant que je suis passé au No Code, j'aurais du mal à refaire un site web sur d'autres plateformes, c'est tellement moins gratifiant d'un point de vue design. Et je pense que pour ceux qui n'ont pas cette curiosité envers le No Code, ça peut être un peu frustrant d'entendre parler de ça ».

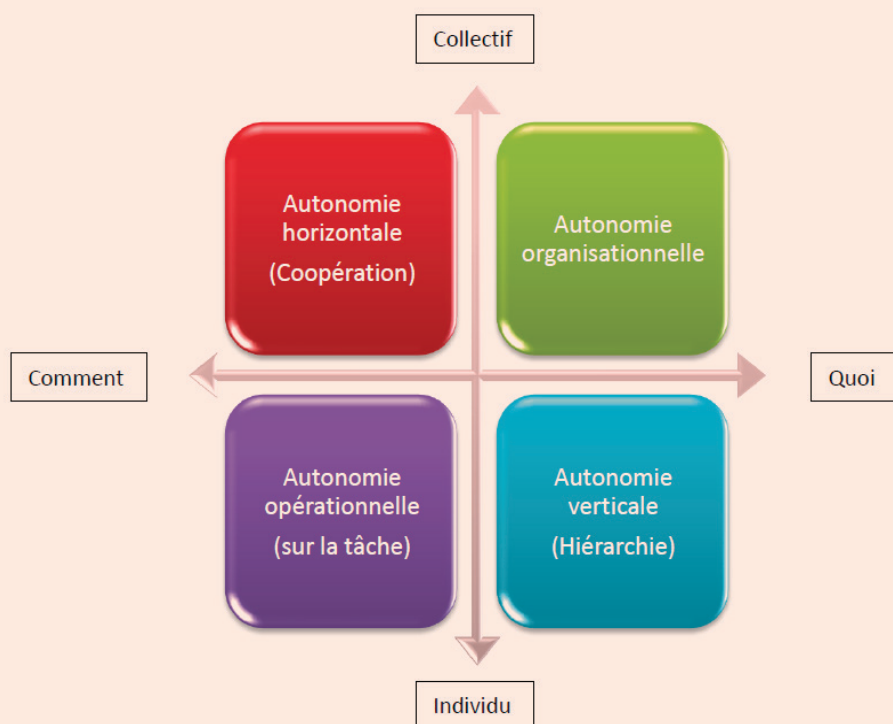
DES EFFETS FAVORABLES SUR L'AUTONOMIE AU TRAVAIL

Il existe de nombreuses grilles d'analyse des champs de l'autonomie au travail, généralement bâties de façon ternaire : par exemple la segmentation entre autonomie sur la tâche, sur l'environnement organisationnel du collectif (« coopération ») et sur la gouvernance²⁰ ; ou sur le comment, le quoi et le pourquoi²¹. Nous inspirant de ces précédentes recherches, nous proposons ici une matrice de l'autonomie au travail organisée autour de deux axes : autonomie individuelle / collective d'une part, autonomie sur le comment (tâche) / le quoi (activité), d'autre part. La matrice permet de dégager quatre dimensions d'autonomie (figure 3) sur lesquelles les TIC en général²², et les plateformes no-code en particulier, peuvent

venir agir : autonomie opérationnelle (manières de réaliser la tâche), autonomie horizontale (coopération intra et inter-équipes), autonomie verticale (hiérarchie entre conception/exécution ou décision/exécution), autonomie organisationnelle (pouvoir d'influencer l'environnement organisationnel et collectif). Le niveau concernant « la gouvernance » a été ici évacué de l'analyse, car les technologies ne semblent pas entraîner d'effets notables sur ce plan (du moins à ce stade de la recherche).

L'appropriation et l'usage des technologies no-code dans l'entreprise observée semblent produire des effets favorables sur au moins trois des quatre niveaux d'autonomie au travail.

FIGURE 3 - MATRICE DE L'AUTONOMIE AU TRAVAIL À 4 DIMENSIONS



20. Bourdu E., Péretié M.-M., Richer M. (2016), *La qualité de vie au travail : un levier de compétitivité. Refonder les organisations du travail*, Les Notes de La Fabrique, Presses des Mines, p. 137.

21. Weil T., Dubey A.-S. (2019), *Au-delà de l'entreprise libérée : Enquête sur l'autonomie et ses contraintes*, Les Notes de La Fabrique, Presses des Mines, p. 76-79.

22. Cette recherche sur le No Code s'inscrit dans un programme plus large visant à décrypter les effets de plusieurs catégories de TIC sur l'autonomie et le travail.

Concernant l'**autonomie opérationnelle** : l'usage de ces technologies n'étant pas prescrit par l'organisation mais seulement favorisé, il n'y a pas de réduction des marges de manœuvre des individus sur la manière d'exercer leurs tâches, mais au contraire l'ouverture d'un champ des possibles. Les tâches s'élargissent, l'autocontrôle sur les processus augmente, et les personnes montent en compétences et en maîtrise par rapport à leur métier initial.

Concernant l'**autonomie verticale** : le No Code agit comme facteur de rapprochement entre les tâches de conception et d'exécution. Aucun élément n'a pu en revanche être mis en évidence concernant des marges de manœuvre supplémentaires que le No code ouvrirait aux collaborateurs vis-à-vis de leur hiérarchie, mais rappelons qu'il s'agit d'une entreprise faiblement hiérarchisée.

Concernant l'**autonomie horizontale** : ce qui est constaté est l'augmentation de la coopération par une meilleure compréhension réciproque du métier et des contraintes de chacun. Toutefois, il faut rester vigilant sur l'émergence de comportements individualistes qui pourraient apparaître via la réduction du sentiment d'interdépendance collective, de même que sur l'existence de poches de résistance (des développeurs refusant de coopérer par exemple).

Concernant l'**autonomie organisationnelle** : celle-ci ne semble pas être directement influencée par la diffusion des technologies no-code, mais au contraire représenter la condition préalable de leur émergence et appropriation dans l'organisation. C'est parce que de grandes marges de manœuvre étaient laissées aux équipes pour s'auto-organiser en vue de satisfaire les objectifs de production (au prix d'un certain « laisser-faire » générateur de stress) que les pratiques no-code ont pu émerger dans l'entreprise. C'est aussi cette autonomie organisationnelle qui a permis de passer de pratiques no-code infor-

melles à la structuration d'un centre de profit basé sur une offre de no-code pour les clients : les salariés peuvent en effet proposer des idées qui sont écoutées par le dirigeant, celui-ci prenant ensuite les décisions finales. En l'occurrence, c'est un salarié impliqué dans le No Code qui a fait la proposition au dirigeant, puis a pris en charge la direction de ce nouveau service. Toutefois, dans cette organisation de type familial, l'écoute dont bénéficient les propositions des collaborateurs ne se traduit pas par un processus de délibération collective qui dépasserait les frontières du Comex. Pour reprendre la segmentation proposée par Weil et Dubey entre les zones rouges et les zones bleues²³ de l'autonomie organisationnelle, la décision stratégique (ouvrir le centre de profit) est une zone rouge réservée au dirigeant, tandis que les modalités de mise en œuvre relèvent de la zone bleue (comment mettre en œuvre le No Code dans l'organisation ; qui évangéliser ; sous quelle forme ; comment associer les développeurs). Dans cette zone bleue, il y a beaucoup de consultation, de discussions et d'écoute au sein de la nouvelle équipe No Code ainsi qu'entre elle et les autres services.

C'est parce que de grandes marges de manœuvre étaient laissées aux équipes pour s'auto-organiser que les pratiques no-code ont pu émerger dans l'entreprise.

23. Thierry Weil et Anne-Sophie Dubey, *op cit.* p. 77. Dans le cadre d'un processus d'autonomisation des équipes, voire de l'entreprise, les zones rouges définissent ce qui n'est pas ouvert à la concertation des personnels ou des équipes, et les zones bleues, les champs qui sont ouverts à la consultation ou à la participation des salariés. Une telle distinction permet d'éviter l'hypocrisie organisationnelle.

CONCLUSION : DES POTENTIALITÉS RÉELLES MAIS UNE EXTENSION SANS DOUTE LIMITÉE

Bien que cette étude ne porte que sur des observations en nombre limité au sein d'une entreprise singulière et sur un champ encore en pleine structuration, la diffusion du No Code dans les entreprises semble avoir la capacité de favoriser des remaniements de l'organisation du travail entre certains métiers, allant vers plus d'agilité, d'autonomie opérationnelle, de décloisonnement et de dialogue-coopération. Il faudra toutefois veiller à déminer des réticences pouvant émaner des DSI et des développeurs pour lesquels la balance coûts-bénéfices de ces outils n'est pas évidente et qui sentent leur pouvoir menacé. Des précautions sont donc à prendre pour embarquer ensemble tous les métiers, plutôt que de les jouer les uns contre les autres.

Les apports du No Code doivent aussi être nuancés face à la promesse de « tous nos codeurs ». Au vu des barrières techniques ou de temps d'apprentissage qui viennent réduire l'accessibilité du No Code à des profils particuliers susceptibles de s'en saisir, il est probable que ces outils se déploieront en premier lieu chez les fournisseurs de solutions numériques pour réduire les coûts et les délais de mise au point des logiciels et applications. Les développeurs *full-stack*²⁴ qui sauront outrepasser la fierté d'être « codeurs » et y voir leur intérêt, s'en empareront probablement aussi pour faciliter et accélérer leur travail, de même que les start-uppers ou entrepreneurs pour « sortir » leurs preuves de concept rapidement et à moindre coût. De là toutefois à imaginer une démocratisation de ces outils dans toute l'entreprise auprès des personnels non techniques dans de larges pans de métiers et services supports, il y a un grand pas à franchir.

L'hypothèse la plus probable pourrait être l'apparition d'une nouvelle couche de « techniciens no-codeurs », autrement dit l'émergence d'un nouveau métier expert inséré dans les *digital factories* chargées d'industrialiser les produits numériques. Les compétences no-code pourraient ainsi rester concentrées dans des centres d'expertise plutôt que largement décloisonnées à travers les métiers. Une distinction pourrait aussi s'établir entre des *power makers*, capables de devenir complètement autonomes sur la réalisation de produits numériques avec le no-code, et des *power users* qui se limiteront à des rôles d'édition de contenus, de mise à jour ou de petite maintenance des outils réalisés en no-code, contribuant à créer de nouvelles hiérarchies. Même si la démocratisation partielle des pratiques no-code peut contribuer à élever le niveau de maîtrise numérique au sein des organisations, son impact pourrait donc rester plus limité que cela n'est fantasmé.

Les approches prescriptives imposées de façon hiérarchique peuvent aboutir à des effets positifs à court terme en matière de coûts et délais de création de logiciels, mais elles risquent de venir contrecarrer la potentialité qu'ont ces outils à créer des affinités et de l'émulation collective.

Enfin, ces pratiques gagneront à être favorisées plutôt qu'imposées via des programmes de formation massifs ou l'établissement de nouvelles prescriptions. Les approches prescriptives imposées de façon hiérarchique peuvent en effet aboutir à des effets positifs à court terme en matière de coûts et délais de création de logiciels, mais elles risquent de venir contrecarrer

24. Un développeur généraliste, polyvalent ou plus familièrement appelé « développeur à tout faire », est un codeur capable de réaliser la programmation d'un site ou d'une application web à la fois en front-end et back-end. C'est ce profil qui pourrait à la fois être le plus affecté par les plateformes no-code, mais aussi en tirer le plus de bénéfices pour réduire les tâches répétitives et peu intéressantes.

la potentialité qu'ont ces outils de créer des affinités et de l'émulation collective, et *in fine* de l'autonomie et de la coopération dans le travail. Ni des bénéfices techniques durables, ni des bénéfices organisationnels solides ne pourront alors véritablement se déployer à partir de ces outils. Il serait illusoire de penser que les outils sont intrinsèquement en mesure de façonner les structures humaines dans le sens du changement souhaité. *A contrario*, ceux-ci auront beaucoup plus de chance d'être adoptés dans des environnements déjà ouverts et fluides, stimulant la curiosité et l'entraide, avec un management de soutien professionnel et des salariés en autonomie. Dans un tel contexte, les usages du no-code pourront alors venir renforcer les dispositions déjà présentes à la fois au sein du dispositif technique et dans l'organisation.

À cet égard, l'un de nos interlocuteurs souligne : « *Le No Code, ça va avec un état d'esprit. Toutes les organisations ne peuvent pas mettre ça en place.* » Dès lors, technique et organisation doivent être pensées ensemble, les deux se façonnant mutuellement : l'outil comporte certes des potentialités mais qui ne peuvent être activées que par des usages qui sont eux-mêmes à la fois permis et contraints par les caractéristiques du contexte organisationnel dans lequel il est déployé.

« *Le No Code, ça va avec un état d'esprit. Toutes les organisations ne peuvent pas mettre ça en place.* »

LES PUBLICATIONS DE LA CHAIRE FIT²



COLLECTION REPÈRE FUTURS DU TRAVAIL

- N°1 - Proximité et distance, communication synchrone et asynchrone
- N°2 - Travail hybride : plus facile à dire qu'à faire
- N°3 - « Être un leader, c'est donner le pouvoir à ses équipes »
- N°4 - Comment les usages du no code impactent le travail



COLLECTION LES NOTES DE LA FABRIQUE

- Design du travail en action (note 37)
- Travail à distance (note 35)
- Au-delà de l'entreprise libérée (note 32)
- Organisation et compétences dans l'usine du futur (note 28)
- Tempête sur les représentations du travail (note 25)



LES MÉCÈNES DE LA CHAIRE FIT²



GRUPE RENAULT



CONTACT

CHAIRE FIT²
Futurs de l'Industrie et du Travail :
Formation - Innovation - Territoires
Mines ParisTech

60 bd Saint-Michel, 75006 Paris
 chaire-fit2@mines-paristech.fr

CHAIRE
FIT²
FUTURS
DE L'INDUSTRIE
ET DU TRAVAIL

MINES ParisTech | PSL