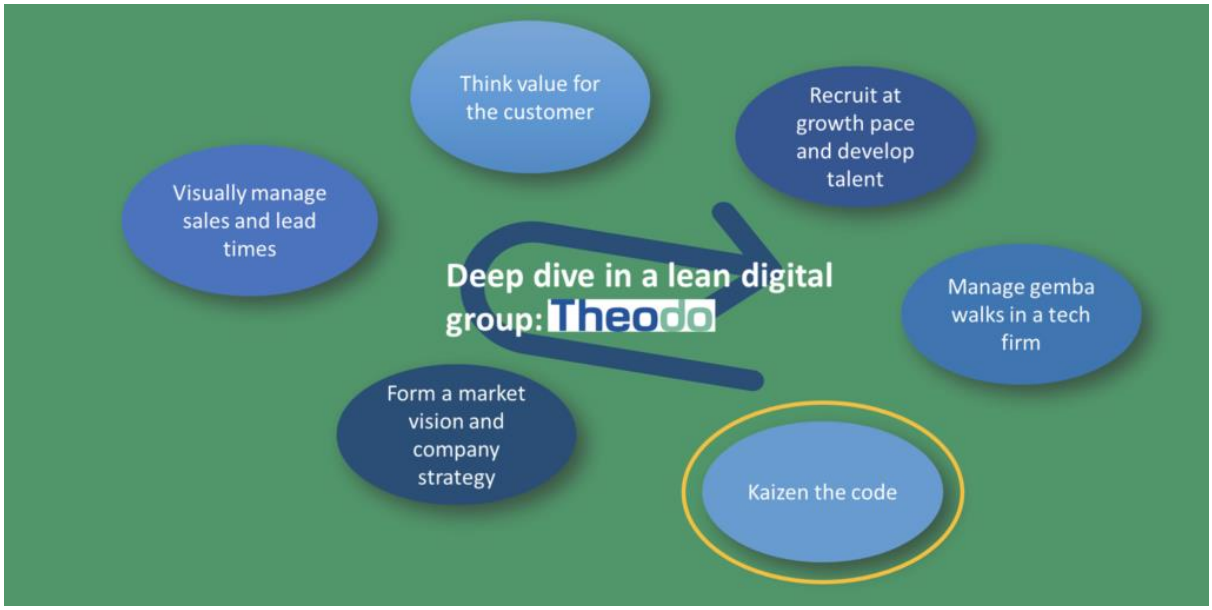
Source : [Deep dive in a lean digital company #5 - Planet Lean (planet-lean.com)]

Immersion dans une entreprise Lean du monde numérique – Chapitre 5



RÉSUMÉ - Depuis maintenant quelques années, Theodo a fait de la qualité et de la satisfaction des clients l'axe principal de son activité, et cela porte ses fruits. Mais que signifie promouvoir et améliorer la qualité dans une entreprise numérique ?

Par Catherine Chabiron, *auteure Lean et membre de l'Institut Lean France*

Bienvenue à la cinquième immersion sur le gemba du groupe Theodo. Dans cette série, nous apprenons comment la pensée lean peut aider une start-up du numérique à mûrir et à prospérer. Dans les quatre premiers articles, nous avons vu comment l'entreprise améliore ses délais de livraison aux clients, comment elle réfléchit efficacement à la valeur pour le client, comment elle recrute et fidélise les talents, et comment elle pratique le gemba walk dans un monde technologique. Aujourd'hui, je me consacre à la qualité.

2012 : UN CAP SALUTAIRE VERS LA SATISFACTION DU CLIENT

Fabrice Bernhard, l'un des deux cofondateurs du groupe, est assis en face de moi, et réfléchit à la manière dont l'importance accordée par Theodo à la qualité est réellement née. « En 2012, nous avons eu notre première inspiration: nous devions nous concentrer beaucoup plus sur la satisfaction des clients afin qu'ils reviennent et achètent davantage », explique-t-il. Inspiré par l'approche « Client un jour, client toujours » de Toyota, nous avons conçu des questionnaires hebdomadaires pour comprendre ce que le client pensait de la livraison de Theodo. Les équipes projet envoient désormais ces questionanires à leurs clients dès que l'affaire est conclue et les interrogent chaque semaine sur leur satisfaction quant à la rapidité de la livraison et au service de l'équipe.

Cela a radicalement changé la donne. Ce type d' *andon* client oblige le développeur à écouter attentivement le client au moins une fois par semaine et à s'en tenir à ce qu'il veut réellement. Ce n'est peut-être pas une confrontation agréable, mais c'est certainement une excellente occasion d'apprendre. Le résultat de cette bascule vers la satisfaction du client a été impressionnant : en 2016, Theodo a décuplé son chiffre d'affaires grâce, du moins en partie, aux clients qui revenaient.

LA QUALITE EST-ELLE LE PARENT PAUVRE DE LA *DELIVERY* ?

Malheureusement, il ne suffit pas d'être disponible et de livrer rapidement pour satisfaire pleinement un client. Aujourd'hui, je rencontre également Rémy Luciani, connu chez Theodo sous le nom de Mr Kaizen. Rémy explique comment la pression de la *delivery* a pu parfois éclipser le besoin de construire la qualité à chaque étape : « Dans notre métier, nous avons une pression constante sur la livraison. Nos bonus, tant au sein de l'entreprise que dans nos relations avec les clients, ont tendance à être basés sur les délais. C'est probablement parce que nous pouvons facilement mesurer le résultat par la livraison des tâches prévues, alors que l'évaluation de la qualité n'est pas aussi simple. »

Les deux fondateurs de Theodo – Fabrice Bernhard et Benoît Charles-Lavauzelle – se sont rendus au Japon en 2017 et 2019 pour voir le Toyota Production System en action chez les fournisseurs de Toyota. « Cela a été un choc », raconte Fabrice. « Nous avons appris que si nous voulions réduire sérieusement les retouches et les retards, il nous fallait être présents sur le *gemba* bien plus que nous ne l'étions, pour développer une bonne réflexion et un bon apprentissage en matière de conception. »

L'histoire de la visite chez Mifune, un fournisseur Toyota de rang 2, est l'une des préférées de l'équipe de management de Theodo. Pendant la visite du fournisseur, Fabrice et Benoît ont demandé combien de fois le président de Mifune était sur le *gemba*, s'attendant à une réponse de l'ordre de quelques heures par semaine ou par mois. On leur a répondu qu'il y était en moyenne 10 minutes par heure ! Non seulement cela a révélé la réalité de la présence de la direction dans l'atelier, mais cela a également mis en lumière le choix puissant d'effectuer des *gemba walks* par petits lots pour observer différentes opérations ou activités à différents moments de la journée. Ils ont également découvert que l'exigence qualité de haut niveau attendue par le client n'était pas seulement mesurée juste avant la livraison, mais qu'il s'agissait d'un effort continu à toutes les étapes de la production.

Cela leur a donné matière à réflexion. Lorsque vous développez une application, vous pouvez choisir de la tester à la fin (l'équivalent d'une inspection finale sur une chaîne de montage) ou au fur et à mesure du codage (autocontrôles à chaque étape de l' « assemblage »). Si vous optez pour la première solution – test d'acceptation par l'utilisateur – vous vous préparez une longue période d'évaluation qui débouche souvent sur une large liste de modifications de dernière minute et d'éléments à retravailler. Il faut en effet superviser des tests de robustesse ou de capacité (le produit peut-il supporter les connexions simultanées de la part d'un grand nombre d'utilisateurs ?), des tests de non-régression (un changement dans une fonction crée-t-il un défaut dans une autre fonction ?) et des tests d'intégration (les données circulent-elles facilement de bout en bout, avec les transformations prévues ?). Construire la qualité à chaque étape prévient les défauts et évite de découvrir les problèmes à la dernière minute.

SE RECENTRER SUR LA QUALITÉ

En 2019, Fabrice a commencé à arpenter lui-même le *gemba* pour regarder le code et apprendre des problèmes rencontrés par l'équipe. Un référentiel qualité, appelé les 3S, a rapidement été mis en place pour le groupe Theodo:

- Stabilité – pas de bugs
- Vitesse – temps de réponse rapide
- Sécurité – pas de vulnérabilité

Le temps de réponse a fait l'objet d'un vaste débat. S'agissait-il du temps de réponse perçu par un utilisateur individuel ou du temps de réponse lorsqu'un grand groupe d'utilisateurs étaient connectés ? L'obsession de la satisfaction client, mantra de la pensée lean, a finalement prévalu : ce qu'on voulait vérifier ici, ce n'était pas l'évolutivité de l'application, mais le temps de réponse perçu par *un* utilisateur individuel, qu'il se soit connecté seul ou en même temps qu'un grand groupe de personnes.

Lorsqu'il se rend sur le *gemba*, Fabrice a trois objectifs clairs : vérifier la réalité de « l'atelier » par rapport à ce qu'il a en tête en tant que dirigeant et cofondateur ; mettre en valeur les réalisations des techniciens qu'il rencontre ; et faciliter le développement d'une stratégie qualité parmi les *team leaders*.

Les deux *gemba walks* hebdomadaires qu'il effectue depuis deux ans ont révélé des choses intéressantes sur les équipes. « Nous avons des recrues brillantes qui créent des choses ingénieuses, me dit-il, mais si elles peuvent

sortir des sentiers battus et maîtriser des technologies complexes, le *kaizen* est pour elles l'occasion de soulever des questions intéressantes sur le travail et la manière de répondre à une demande du client. » Rendre visite aux équipes deux fois par semaine permet de comprendre qui fait quoi et qui est particulièrement compétent sur un sujet donné, et Fabrice réoriente parfois les Techs qu'il a rencontrés vers des équipes ou des personnes qui pourraient les aider et leur faire gagner du temps.

Le code est également une source importante d'améliorations potentielles. « Je vois parfois des bugs, avérés ou potentiels. Et lorsque je demande à voir le dernier segment de code que les développeurs ont écrit, je suis surpris de voir qu'il s'agit souvent de la reprise d'un précédent code défectueux », poursuit-il. Les bibliothèques de code sont de bonnes idées, mais copier-coller un segment de code sans comprendre clairement l'intention qui se cache derrière peut mener à une mauvaise utilisation. L'intention de Fabrice est d'encourager les gens à réfléchir avant d'utiliser un code standard.

Curieusement, cela me rappelle les *best practices* que les grandes entreprises aiment promouvoir : copier-coller quelque chose qui a été conçu par une équipe pour résoudre un problème spécifique ailleurs n'est pas forcément une bonne idée. D'un autre côté, énoncer le problème initial auquel l'équipe s'est attaquée peut être un excellent moyen d'apprendre des autres et de développer votre propre réponse. En d'autres termes, ne partagez pas la solution mais la question initiale pour inciter les équipes à penser « lean ».

La nécessité de travailler sur la qualité du code a donc été confirmée et, en septembre 2020, Rémy a été sollicité pour aider les équipes sur le kaizen.

APPRENDRE PAR LA RESOLUTION DE PROBLEMES

Fabrice confirme que c'est en 2016 qu'a eu lieu la première tentative de kaizen menée chez Theodo, axée sur le code et le développement. Bien qu'intéressante, il confirme qu'elle s'est attaquée à des processus longs, et qu'aborder des questions aussi complexes a pu parfois rebuter les équipes.

C'est pourquoi Rémy a été chargé d'apporter son soutien sur ces *kaizen* l'année dernière. Il est chez Theodo depuis 10 ans et est pleinement convaincu de la nécessité d'utiliser une approche scientifique dans la *delivery*. Il a trop souvent constaté que les équipes prenaient des raccourcis pour éviter de chercher à comprendre en profondeur pourquoi un problème se posait. Selon Rémy, une partie du temps passé à débattre des mérites comparatifs des outils disponibles sur le marché du numérique devrait être consacrée au développement d'une compréhension fondamentale du fonctionnement réel des choses.

Pleinement conscient du fait qu'il avait devant lui un long parcours semé d'embûches pour essayer de convaincre les équipes de trouver du temps pour des *kaizen* réguliers, Rémy a décidé de leur simplifier la tâche. « Je pousse les équipes à choisir des sujets ciblés et collectifs. Essayer d'aborder des projets vastes et transversaux, comme nous le faisions autrefois, est un moyen sûr de les noyer dans de vastes plans d'actions correctifs, sur lesquels elles n'ont pour la plupart aucun levier », explique Rémy.

Il sait également que, par le passé, les résultats de *kaizen* ont pu être enjolivés pour passer un cap de revue par des pairs, dans le cadre d'une évolution de carrière. Si vous vous souvenez de ce que Marie, la RH du groupe Theodo, a dit dans le troisième article de la série, un modèle de maturité *kaizen* était autrefois une étape obligatoire dans le développement de carrière. Ils ont depuis renoncé à cette pratique.

Rémy prend donc les équipes projet une par une et les aide à mettre en évidence un problème de performance qui peut être traité en un ou deux jours maximum. « Nous devons instaurer la confiance dans l'approche, et l'assurance viendra au fur et à mesure des résultats et/ou des apprentissages », me dit-il.

Rémy adhère pleinement à l'apprentissage par la résolution de problèmes sur le code : « Favoriser la collaboration sur un problème de qualité est un excellent moyen de développer vos compétences tant sur le code que sur votre capacité à écouter activement tout en influençant les autres. »

Une fois le problème de performance défini, et si l'équipe est dans des conditions raisonnablement stables (la base du TPS), Rémy propose de programmer un atelier *kaizen* d'une journée. Avec une nouvelle difficulté : l'accent étant mis sur la *delivery* et la facturation au client étant basée sur le temps passé sur l'application dans le monde numérique, plutôt que sur le résultat, quelqu'un doit dire au client qu'une journée entière consacrée à l'amélioration de la qualité sera facturée. Si certains clients peuvent penser que c'est une excellente idée, d'autres fronceront sans doute les sourcils et auront besoin d'être convaincus. Attention, ce n'est pas parce qu'ils ne se soucient pas de la qualité, mais parce qu'ils pensaient que c'était une partie intrinsèque du travail au départ !

Après un démarrage difficile fin 2020, Rémy a désormais atteint une vitesse de croisière d'un à deux ateliers *kaizen* par semaine.

LE *KAIZEN* EN SIX ÉTAPES

Rémy applique une approche *kaizen* classique en 6 étapes.

Tout d'abord, vous vous mettez d'accord sur la performance à améliorer. Ensuite, vous passez du temps à observer la façon dont le travail est effectué aujourd'hui et effectuez des mesures en temps réel au fil de l'eau, en les répétant en cas de forte variabilité. Rémy me montre un exemple où ils ont chronométré la mise en production complète, jusqu'à l'Apple Store. Il s'est avéré que l'équipe ne connaissait pas bien les outils, perdait du temps à les configurer ou à attendre une notification qui ne venait pas. Elle considérait l'ensemble du processus comme complexe et peu familier, et était donc tentée de regrouper de gros blocs de construction avant d'essayer de les diffuser dans l'Apple Store. Cela augmentait le temps de mise sur le marché des nouvelles fonctionnalités.

Cette phase d'observation de la méthode actuelle est cruciale : vous pourriez être rapidement tenté de sauter sur une solution qui vous traverse l'esprit sans approfondir suffisamment la réflexion, et le rôle de Rémy est essentiel pour éviter cela. « Ma journée est réussie si je les vois ouvrir les yeux et s'émerveiller de leurs apprentissages. Ils me disent souvent que c'est formidable de pouvoir prendre le temps d'observer, d'analyser et de réfléchir », dit-il.

La troisième étape consiste à réfléchir à de nouvelles idées. Dans cet exemple, l'équipe a trouvé 16 idées pour alléger la charge et réduire le risque d'erreurs. Ils en ont sélectionné une à ce stade – automatiser le téléchargement des *builds*.

Cette étape idées nouvelles est l'occasion d'expérimenter individuellement quelques-unes de ces idées afin d'être en mesure de les comparer et de sélectionner la meilleure. « En général, nous observons et analysons le matin et nous jouons avec les nouvelles idées l'après-midi. L'intérêt est vraiment d'essayer quelque chose immédiatement, car cela peut révéler des obstacles inattendus. En outre, chaque membre de l'équipe apprend en s'exerçant, seul ou en binôme », précise Rémy. L'apprentissage ne se fait pas collectivement. Vous apprenez lorsque vous explorez une idée et expérimentez une solution possible, et vous apprenez seul. Le temps alloué par Rémy à l'apprentissage individuel est crucial.

Rémy insiste alors : « C'est un atelier très orienté vers la pratique. Parfois, nous commençons sur un point de performance et le *kaizen* est terminé à midi. Nous pouvons alors en commencer un nouveau dans l'après-midi. Dans d'autres cas, l'analyse prend du temps, mais les enseignements sont utiles même si les résultats ne sont pas atteints à la fin de la journée, tant sur la façon dont les gens voient leur travail que sur la démarche *kaizen* elle-

même. » L'objectif de Rémy pour la journée est d'atteindre au moins l'étape 4 – définir le plan de mise en œuvre de l'idée sélectionnée.

Il revient aux équipes de mettre en œuvre le plan (étape 5) et d'évaluer la nouvelle méthode (étape 6).

Rémy me montre un autre *kaizen* sur les *Data Transfer Objects*. Il me dit : « Nos développeurs passent en fait plus de temps à lire du code qu'à en écrire. Ils se basent sur des applications existantes (développées en interne ou fournies par le client) pour en créer une nouvelle. Un code de mauvaise qualité ne tarde pas à être la source d'une charge mentale inutile. Et les DTO sont l'une de nos cibles car ils sont fréquemment utilisés. »

Il me montre le cas d'un DTO d'identification (par nom d'utilisateur ou mail et mot de passe), typiquement conçu pour traiter les droits d'accès à une page ou une application. Mais en y regardant de plus près, il y a en fait deux cas d'utilisation différents pour ce DTO : une création de compte ou une autorisation d'accès une fois le compte créé. Et si l'utilisation du DTO que l'équipe a sélectionné dans ce *kaizen* serait pertinente dans le premier cas, elle ne l'est pas dans le second. Ils ont donc trouvé une duplication de code qui n'a pas tenu compte du contexte d'utilisation.

```
15  @Data
16  @Builder
17  @NoArgsConstructor
18  @AllArgsConstructor
19  @EmailOrUserAccountIdRequired
20  public class PermissionDto {
21
22      @NotNull
23      private PermissionEntityType entityType;
24
25      @NotNull
26      @NotEmpty
27      private String entityId;
28
29      @NotNull
30      private List<PermissionType> permissionTypes;
31
32      @Nullable
33      private String permissionName;
34
35      @Nullable
36      private String userAccountId;
37
38      @Nullable
39      private String email;
40  }
41
```

Parmi les autres problèmes fréquemment rencontrés sur ces DTO, citons le problème des différents types de données au sein d'une même application (une chaîne de caractères dans un cas, un nom de permission dans un autre).

En fait, le nombre d'activités qui peuvent faire l'objet d'un *kaizen* est stupéfiant. Fabrice peut en citer un certain nombre : « Les techniciens prennent l'initiative de choisir leurs outils et le flux. Et ces choix ne sont pas toujours les meilleurs. Certains *kaizen* sont donc dédiés au *lead-time* ou au *touch time* dans la construction de nos produits, mais il y a également de nombreuses améliorations possibles sur l'architecture du logiciel, sur les problèmes de duplication, les interfaces, les bases de données. Ou encore sur les tests. Il est clair que nous manquons de standards dans ce domaine. »

Il ajoute ensuite avec un sourire : « On peut rajouter aussi tout ce qui empêche les Techs de travailler de manière fluide, dans le flux, avec le sentiment que tout se met en place quand il le faut ». Rémy souligne : « Je pense à une application dont l'environnement de développement met huit minutes à démarrer le matin – assez pénible. »

CÉLÉBRER LES APPRENTISSAGES

Rémy est persuadé qu'à ce stade, seulement 5 à 10% des équipes qui ont participé à un atelier *kaizen* avec lui ont poursuivi les efforts de *kaizen* par elles-mêmes. La route sera longue et tous ceux qui ont essayé de promouvoir le fameux mantra *JOB = WORK + KAIZEN* ont dû, à un moment ou à un autre, faire face à cette réalité décourageante. Mais l'enjeu est de taille et Rémy est déterminé.

Une façon d'encourager le *kaizen* est de communiquer abondamment sur les enseignements qu'on en tire. Fabrice intervient : « Au Royaume-Uni, nous faisons un débriefing sur les enseignements des *kaizen* une fois par mois. Et en France, la TV Theodo et les réunions *Asakai* sont utilisées pour présenter les résultats et remercier les techniciens qui se sont impliqués. »

L'objectif de ces débriefings est de valoriser le temps que les gens passent à apprendre. « Seuls les meilleurs geeks font cet apprentissage continu de manière spontanée », explique Fabrice. « Nous voulons montrer à tous les membres du groupe la valeur que nous accordons à l'effort de réflexion avant d'agir et à quel point cela peut être fun. *Une bonne réflexion pour de bons produits*, comme nous l'enseigne Toyota. J'ai souvent été très impressionné par l'investigation et les apprentissages résultant d'un problème, et c'est le genre d'exemples que nous devons continuer à diffuser dans l'entreprise.

L'AUTEURE

Catherine Chabiron est auteure Lean et membre de l'Institut Lean France

Traduction par Marc-Antoine Guichard, Nicolas Villemain et François Lopez