Ségolène Porot - licence CC 4.0

Le guide du product owner

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Introduction

Ce guide pour product owner s'adresse plus particulièrement aux nouveaux product owner. Néanmoins, les curieux et les expérimentés en recherche de nouvelles perspectives dans ce domaine sont les bienvenus.

Le concept est simple, il s'agit de résumer pour un nouveau product owner les éléments clés pour réaliser un produit tout au long de son cycle de vie à travers le rappel de ses responsabilités, les outils à disposition et des astuces pour remplir ce rôle crucial et stratégique pour la mise en place d'un produit à haute valeur technologique.

Ce guide a été conçu à l'origine pour obtenir une vue d'ensemble dans un seul et unique ouvrage les objectifs, les outils, les bonnes pratiques pour être un product owner expert et accompli.

En effet, le constat est le suivant, la plupart des informations sont dispersées dans des articles de blogs, des meet-up... au final très peu de livres ou de guides spécialisés existent dans ce domaine pour les débutants en quête d'une synthèse de qualité.

C'est pour cela aussi que j'ai pris l'initiative de créer ce document : simplicité et efficacité seraient les mots pour décrire au mieux l'ambition de cet ouvrage.

Cet ouvrage a un but avant tout méthodologique en vous présentant les outils à votre disposition pour chaque étape clé du produit,.

Je lisais encore les différentes types de product management, aussi, il en existerait un certain nombre : le Business owner (appelé aussi product manager) , le Product owner, le Product marketing manager.

A mon sens, ce n'est qu'un seul et même métier celui de « product owner », il ne s'agit ni plus ni moins d'une spécialisation ensuite ou tout simplement d'une opération marketing pour les cabinets de conseil ?

Quoiqu'il en soit, le fait est, que le métier de product owner est à l'origine un rôle inventé par des développeurs pour des développeurs dans un contexte de start-up.

Typiquement, le modèle et le rôle de product owner s'adapte en fonction de l'entreprise dans lequel il s'applique, c'est-à-dire : de la mise en œuvre de la méthode agile et enfin des compétences et du savoir-être de la personne qualifiée de product owner.

Je traiterais toutes les facettes plus ou moins longuement. Certaines seront plus détaillées dans un prochain guide que je vous communiquerai prochainement.

Bonne lecture

Ségolène POROT

beNext Company

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Sommaire

- **Partie 1 : Qu'est-ce qu'un product owner ?**
  - A) Le rôle du product owner
    1) Description du rôle du product owner
    2) Le product owner : des tâches diversifiées au quotidien
    3) Les règles du product owner expert
  - B) Rappel des cérémonies SCRUM
    4) Les principes de scrum
    5) Les rituels scrum
    6) Quelques illustrations
  - C) Quel est le déroulement d'un sprint type en tant que product owner ?
    1) Illustration d'un sprint type pour un product owner
    2) Explication de l'illustration
    3) Les astuces pour optimiser votre temps

- **Partie 2 : Définir du sens et de de la valeur tout au long du cycle de vie du produit**
  - A) Phase préalable : définir notre marché et nos utilisateurs
    1) Identification du marché du produit numérique
    2) Identification de la cible du produit numérique
    3) Les outils pour bien démarrer votre produit numérique
  - B) Phase 1 : comment lancer un produit ?
    1) Qu'est-ce que la vision d'un produit ?
    2) Qu'est-ce qu'une User story ?
    3) Les outils consacrés à votre produit
  - C) Phase 2 : comment assurer une croissance continue du produit ?
    1) Planifier des réunions clés de suivi de votre product backlog
    2) Des actions spécifiques pour votre product backlog
    3) Dynamiser vos ateliers

- **Partie 3 : Savoir, savoir-être, intention et attitude du product owner**
  - A) Les compétences générales et détaillées du product owner
    1) Le référentiel de compétences du product owner
    2) Comment acquérir ces compétences
    3) Les conseils des experts pour les nouveaux
  - B) Le savoir-être du product owner
    1) Management, organisation des RH
    2) Cultiver l'intention bienveillante
    3) Attitude : le product owner caméléon
  - C) Proposition de perspective sur le rôle de PO
    1) Gérer le change management
    2) Un rôle de coach
    3) La pédagogie : l'outil inattendu du PO

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

PARTIE 1

QU'EST-CE QU'UN PRODUCT OWNER ?

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Il me semble important avant de rentrer dans le vif du sujet de présenter le rôle du product owner. En commençant dans ce métier, je me suis posée des questions simples mais structurantes, telles que : Qu'attendons-nous d'un product owner ? Quelles sont ses priorités ? Quelles sont ses tâches et responsabilités au quotidien ? Quelle posture doit-il adopter ? Quelles sont ses interactions avec l'équipe de développement ? Avec le Scrum Master ? Avec les parties prenantes de l'entreprise ? Quels sont ses outils et ses facteurs clés de succès ? A quoi reconnaît-on un "bon" Product Owner ?

Certains termes pour les nouveaux du métier seront évoqués mais ne vous inquiétez pas l'explication et la mise en œuvre seront décrites dans la partie 2 de cet ouvrage.

Le but est dans un premier temps de bien comprendre et assimiler les bases du métier de product owner.

A) Le rôle de product owner

1) Description du rôle du product owner

Son rôle est de maximiser la valeur du produit et du travail de l'équipe de développement, pour cela plusieurs domaines d'activités sont à la charge du product owner telles que :

Marketing stratégique	Observer et analyser la concurrence (veille) Observer le marché Définir la cible Observer ses clients Définir les KPI Evaluer Taux de satisfaction client du produit, Connaître les enjeux Mettre en corrélation le quoi et le pourquoi Faire évoluer sa stratégie Définir la roadmap pour atteindre l'objectif stratégique Donner du sens, une perspective Échanger sur une vision et non sur le détail
Marketing opérationnel	Garantir l'explication sur le quoi et sur le pourquoi Gérer l'optimisation et la maintenance des sites internet, extranet, portail... Améliorer la visibilité des sites : acquisition de trafic et de référencement Piloter la création, la mise en place et le suivi de nouveaux supports digitaux dans un environnement multicanal

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Communication	Énoncer des éléments de valeur Développer une bonne approche Exprimer la valeur, ses enjeux, son agenda, les choix qui ont été fait, pourquoi, les résultats Intégrer les bonnes personnes dans sa réflexion Répondre aux questions pendant l'action Conduire le changement
Innovation	Construire et activer son plan de veille multicanaux Utiliser les leviers de l'innovation ouverte pour détecter les besoins et technologies émergentes. Repérer les discontinuités de l'environnement. S'inspirer des insights consommateurs ou clients. Surveiller les nouveaux comportements lors du parcours client, vis-à-vis du produit global. Évaluer les compétences clés de l'entreprise. Identifier les nouvelles aires de business. Construire la "roadmap" et le business model de l'innovation.
Gestion de projet IT	Piloter, suivre et coordonner les actions pour concevoir le produit Etre tactique pour découper, organiser, prioriser les US Clarifier les impacts, les risques, coût et délai Réaliser les tests et recette fonctionnelle
Organisation	Découper les produits Reformuler, anticiper, faciliter l'appropriation pour l'équipe Définir les processus et procédure métier

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

2) Le product owner : des tâches diversifiées au quotidien

Le product owner est le seul responsable du product backlog, la plupart de ces tâches sont liées à celui-ci :

1) Décrire de façon clair les éléments du product backlog
2) Prioriser le product backlog
3) Valider/Rejeter une fonctionnalité « terminée » dans le product backlog en cours
4) S'assurer que le product backlog est accessible et compréhensible de tous
5) Faire le product backlog grooming
6) Tester les fonctionnalités du produit : remonter les anomalies et assurer la qualité des développements
7) Réaliser les atelier métiers pour définir le besoin
8) Définir la vision du produit
9) Faire la roadmap : plan de release, but/objectif des sprints
10) Piloter les sprints
11) S'assurer du bon déroulement des cérémonies (en collaboration avec le Scrum Master)
12) Faire la veille du marché et des technologies (arrivée nouveaux concurrents, ruptures technologiques, évolution des tendances clients)
13) Agréger, analyser et communiquer les indicateurs d'avancement et budget

3) Les règles du product owner expert

Les règles du product owner sont une synthèse des accomplissements et résultats nécessaires pour réaliser au mieux votre travail :

1. J'ai la vision du produit (créée avec les clients, les utilisateurs, les investisseurs)
2. Je peux répondre aux questions sur la vision du produit d'une manière claire et concise et motiver mes choix
3. J'ai une phrase courte de synthèse pour la vision du produit pour communiquer l'essence de la valeur du produit pour la release
4. Je comprends les besoins de mes parties prenantes (et plus particulièrement les clients, les utilisateurs et les investisseurs)
5. Je communique régulièrement avec les parties prenantes pour comprendre leurs besoins et gérer leurs attentes
6. J'ai un product backlog
7. Les parties prenantes légitimes ont confiance dans le product backlog
8. Je suis légitime pour prendre des décisions sur le product backlog
9. Je mets à jour le product backlog avant chaque sprint planning meeting
10. Je m'assure que le product backlog est accessible à tous les membres de l'équipe scrum
11. Le product backlog est priorisé (basé sur la valeur, le risque, le travail estimé, les dépendances)
12. Le product backlog est clairement exprimé et plus détaillé au début de product backlog

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

13. Je détaille le product backlog régulièrement avec au moins un développeur pour avoir le product backlog « ready » pour le prochain sprint planning meeting
14. Je suis disponible pour les développeurs pendant les sprints pour clarifier les spécifications et les besoins
15. Je protège mon équipe de développement de quiconque qui essaie d'interférer avec le sprint en cours
16. Mon équipe Scrum me fait confiance dans mon domaine et ma connaissance des utilisateurs finaux si ce n'est pas le cas, j'essaye de créer cette confiance en améliorant ma propre connaissance business et des utilisateurs
17. J'ai la même compréhension de « definition of done » que mon équipe Scrum et mon Scrum Master.
18. J'ai confiance et j'ai une bonne communication avec mon Scrum Master et l'équipe de développement. Si non, on travaille ensemble pour améliorer notre coopération
19. Je participe au sprint planning meeting pour sélectionner les éléments du product backlog avec l'équipe Scrum
20. Je participe (j'anime et/ou facilite) au sprint review meeting pour donner des retours constructifs et présenter les éléments du product backlog qui ont rempli leur unique critère d'acceptance et aussi la definition of done générale.
21. Je participe à la rétrospective meeting pour observer et améliorer mon propre travail en tant que product owner
22. Je travaille avec mon équipe de développement lors du DSM lorsque cela est nécessaire, pour clarifier les besoins, le design, et optimiser les retours du sprint.
23. J'ai planifié les évènements du sprint avec le scrum master (pour les événements répétitifs du calendrier)

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

B) Rappel des cérémonies SCRUM

Comme cité précédemment, le product owner doit être présent aux différentes cérémonies Scrum, je vous propose de vous rappeler brièvement les objectifs de chacune d'entre-elles.

1) Les principes de l'agilité et plus particulièrement de scrum

Les principes de l'agilité se manifestent à travers un certain nombre de valeur :

- Satisfaire le client est la priorité
- Accueillir les demandes de changement « à bras ouverts »
- Livrer le plus souvent possible des versions opérationnelles de l'application
- Assurer une coopération permanente entre Client et Equipe projet
- Construire des projets autour d'individus motivés
- Privilégier la conversation en face à face
- Mesurer l'avancement du projet en termes de fonctionnalités de l'application
- Faire avancer le projet à un rythme soutenable et constant
- Porter une attention continue à l'excellence technique et à la conception
- Favoriser la simplicité
- Responsabiliser les équipes: les meilleures architectures, spécifications et conceptions émergent d'équipes auto-organisées.
- Ajuster, à intervalles réguliers, son comportement, ses processus pour être plus efficace

D'un point de vue humain, c'est aussi un changement important par rapport aux méthodes traditionnelles telles que le cycle en V à travers les thématiques suivantes :

- Le respect pour la valeur de chaque personne
- La vérité dans chaque communication
- La transparence de toutes les données, actions et décisions
- La confiance pour que chaque personne soutienne l'équipe
- L'adhésion à l'équipe et aux objectifs de l'équipe

2) Les rituels scrum

Sprint Planning

Il s'agit de la réunion de lancement de l'itération d'environ 3 ou 4 heures pour un sprint de 2 semaines. Le sprint planning se définit à travers deux étapes : définir le contenu de l'itération à venir (avec l'estimation* et l'engagement). Cette définition est réalisée une par une : dans l'ordre du backlog définit par le product owner pendant 2 heures pour découper en tâche les US pendant 1h30. Par conséquent, le product backlog avec les user stories au début et assez détaillées sont nécessaires et à l'issue on obtient l'engagement de l'équipe sur un ensemble d'user stories.

Ainsi concrètement, Le product owner n'estime pas, ni le scrummaster qui est neutre : c'est l'équipe

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Scrum qui s'engage et estime la complexité.

*Pour traiter la question de l'estimation & l'engagement, qui reste un sujet délicat même pour les équipes chevronnées. Je vous propose de mettre en place « le fameux **Planning Poker**.

Il permet de déterminer collectivement les pondérations en « points » de valeur ajoutée.

On peut également utiliser des échelles de valeur plus ou moins fine (exemple : « faible », « moyenne », « haute » ou encore les tailles de tee-shirt : XS, S, M, L, XL, XXL).

Concrètement, l'ensemble des fonctionnalités de la liste doivent être estimées par l'équipe de développement afin de permettre les futurs engagements de cette dernière. Impliquant plusieurs développeurs, le Planning Poker permet de mettre à profit les expériences de chacun et de parvenir rapidement à une estimation optimale et objective. Avant ou pendant les estimations, le Product Owner pourra être sollicité afin de répondre aux questions de l'équipe de développement. A ce stade, le besoin pourra être approfondi, mais sans aller trop loin (il s'agit simplement d'estimer le coût de chaque exigence). La conception détaillée se fera pendant les itérations (sprints).

Daily Standup :

Il s'agit de la réunion quotidienne de l'équipe à la même heure pendant 15 minutes maximum avec l'équipe uniquement. Le scrummaster est garant de la mise en œuvre pratique. Le public est silencieux mais autorisé. C'est un moment où l'équipe partage l'état des lieux du projet au sein de l'équipe. En pratique, chacun présente le travail effectué depuis le dernier daily standup, le travail sur lequel il travaillera jusqu'au suivant, et si s'il rencontre des difficultés. L'objectif est de partager l'avancement du projet à au sein de l'équipe afin que de garantir un niveau d'information commun.

Review :

Il s'agit du passage en revue des réalisations de l'itération pendant 30/40 min. C'est le lieu où l'on montre les réalisations de l'équipe lors de la dernière itération, mais uniquement ce qui est réellement fini. C'est aussi le lieu où l'on reçoit le feedback des parties prenantes. Tout le monde participe. Ainsi l'équipe présente le résultat de l'itération (fini et validé par le product owner), des éléments d'informations, et une projection (release plan) sur la suite. A l'issue, les parties prenantes connaissent l'état d'avancement du projet et peuvent donner leur feedback.

Rétrospective

Il s'agit du moment où l'on s'interroge sur « comment s'améliorer ? » entre le product owner, scrummaster, l'équipe pendant 2 heures. Ce rituel se déroule en 3 étapes : tout d'abord, on observe le déroulement de l'itération, on définit ce qui s'est bien passé, ce qui s'est mal passé et les axes d'améliorations. Enfin on sélectionne un (et un seul) élément à améliorer et comment y parvenir.

3) Quelques illustrations de rétrospectives

Pour les rétrospectives, je vous conseille :

- Le livre « Agile Retrospectives – Making Good Teams Great »

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

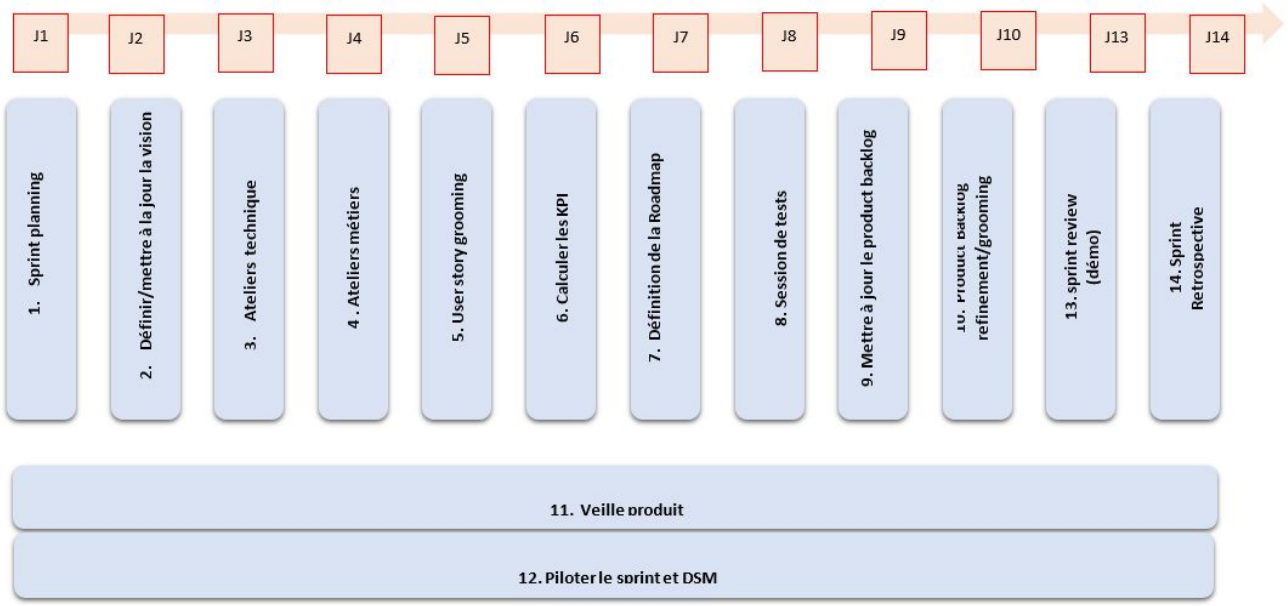
- Le site Fun Retrospectives
- Le site Plans for retrospectives qui vous génère des déroulés de rétrospectives lorsque vous devez aller vite.
- La rétrospective a aussi son wiki dédié.
- La section jeux de Tastycupcakes.

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

C) Quel le déroulement d'un sprint type en tant que product owner ?

1) Illustration d'un sprint type pour un product owner



(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

2) Explication de l'illustration

Cette illustration a pour objectif de présenter un sprint type pour un product owner.

On constate que le product owner a des journées constituées de deux axes principaux : les réunions et la gestion du product backlog. Je considère que les éléments présentés ci-dessus nous permettent tout d'abord d'identifier un point de départ, c'est-à-dire lorsque le sprint débute à travers le sprint planning, explicité ci-dessus. C'est le moment où l'équipe s'engage et pondère les US ainsi que la définition des tâches pour chaque US.

Selon les échanges, la vision doit être mise à jour voir ou être ajustée avec de nouveaux points, ce qui impactera plus ou moins la roadmap qui sera donc à mettre à jour ou à redéfinir.

Enfin, le product owner avance sur ces US en cours, pour cela il va devoir planifier (au préalable du lancement du sprint) les ateliers métiers et techniques nécessaires pour ensuite rédiger ces User Story à travers l'organisation du user story grooming.

Par ailleurs, selon l'avancement des US, le product backlog doit être mis à jour.

C'est aussi dans ce cadre que le product owner peut identifier les risques grâce au suivi des KPI (l'avancement du projet à travers le burndown chart, KPI budget).

Au fil de l'eau, les US seront réceptionnées et c'est au product owner de valider ou non les US à travers des sessions de tests pour vérifier que les critères d'acceptation sont remplis ainsi que la definition of done.

Puis, avant de lancer un nouveau sprint, le product owner planifie le Product backlog refinement pour détailler les US et la priorisation de celles-ci qui seront à présenter pour le prochain sprint planning.

La fin de sprint se caractérise par le Sprint review, où l'équipe ou le product owner montre les US terminées et validées par le product owner tout au long du sprint.

Enfin, la rétrospective est un moment privilégié pour améliorer le produit, s'améliorer en tant que product owner sur la mission donnée pour le prochain sprint et apporter toujours plus de valeur au produit et à l'équipe élargie.

En parallèle de ces actions en continue, le product owner réalise une veille produit qui consiste à observer le marché, les produits concurrents, les clients afin de récolter de nouvelles informations (insights) pour développer, mettre à jour des US.

Il doit également piloter le sprint en participant au DSM si cela a de la valeur ou à travers le suivi du sprint via les fonctionnalités à réaliser, en cours, terminées pour le sprint en cours et à venir.

Enfin, je vous propose en annexe (annexe 1), une histoire pour vous illustrer les tâches du product owner dans un cas concret.

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

3) Les astuces pour optimiser votre temps

Comme nous pouvons le constater, les journées du product owner sont chargées de tâches diverses. Je vous propose ci-dessous quelques astuces pour optimiser votre temps et gagner en efficacité et en sérénité pour remplir votre fonction au mieux. Pour cela je vous propose 3 axes : préparer une liste, définir un agenda et déléguer à un bras droit.

Faites des listes

Une des meilleures façons pour bien gérer son temps, dans le meilleur équilibre personnel et professionnel possible, c'est de dresser une liste des choses à faire, chaque semaine. Ensuite, triez-les selon la matrice d'Eisenhower (voir note 1) :

- Ce qui est important et urgent (en fonction de vos fonctions, de votre poste, de vos priorités et de vos obligations)
- Urgent et non important
- Non urgent et important
- Non urgent et non important

Préparer votre agenda à l'avance

N'hésitez pas à y inscrire, en rouge ou soulignées, les tâches importantes à accomplir. Inscrivez-y aussi les activités que vous aimeriez faire pour vous détendre et respectez-les!

Votre agenda doit refléter l'équilibre entre votre vie personnelle et votre vie professionnelle. Cet équilibre contribuera également à augmenter votre sentiment de satisfaction.

Pour avoir un agenda bien organisé, il faut respecter certaines étapes :

- Sélectionnez vos priorités professionnelles et personnelles.
- Énoncez vos buts en rapport avec vos rôles au travail, à la maison et en société. Inscrivez-y les tâches qui s'y rattachent.
- Faites la liste des activités en rapport avec chaque but énoncé à l'étape précédente et accordez un numéro et un moment de réalisation à chaque activité ainsi planifiée.
- Choisissez, chaque semaine, un moment dédié à la rédaction de vos rôles, vos buts et vos activités. Vous jetterez alors un regard critique sur vos priorités.
- Gardez du temps pour les urgences. Vous en aurez, croyez-moi. En fait, il est préférable d'y consacrer 30 % de votre horaire quotidien...
- Ajoutez-y votre liste des choses à faire...

Faîtes vous aider !

Vous pouvez faire appel à votre équipe pour vous aider, sinon pendant un temps défini car ce n'est pas intégré en que tel à la méthode vous pouvez faire appel à un proxy PO. C'est le bras droit du PO, il va pouvoir s'occuper des espaces vacants que vous ne pouvez pas occuper par manque de temps le plus souvent. N'oubliez pas de lui donner des responsabilités, de lui accorder de l'autorité, de faire des points d'alignement ou de synchronisation et enfin de féliciter lorsque cela est possible !

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

PARTIE 2

Définir du sens et de de la valeur

tout au long du cycle de vie du produit

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

A) Phase préalable : définir notre marché et utilisateur

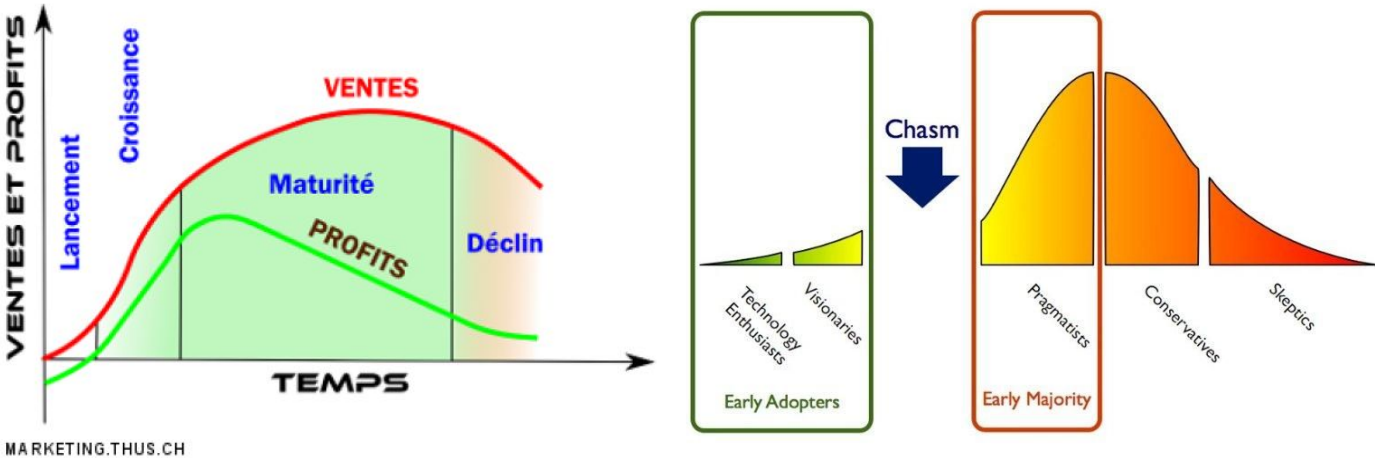
1) Identification du marché du produit numérique

En guise d'introduction de cette partie, je vous propose d'abord de nous rafraichir la mémoire du cycle de vie d'un produit technologique. Cette partie traite essentiellement de la partie amont relative au domaine marketing stratégique du rôle de product owner. Elle sera volontairement succincte et fera l'objet d'un guide plus spécifique pour aborder d'autres thématiques marketings utiles au product owner.

Il existe 4 phases majeurs dans le cadre d'un cycle de vie d'un produit :

- Phase 1 : le lancement

Le besoin de profit n'est pas immédiat. Le produit est communiqué aux clients. Si le produit ne connaît que peu ou pas de concurrence, une stratégie de prix d'écrémage est utilisée. Le produit est fabriqué à travers des petits volumes.



- Phase 2 : Croissance

La concurrence grandit sur le marché. Par ailleurs, le produit atteint son seuil de rentabilisation. Le produit est complété par d'autres offres. En général, on se focalise sur le produit et la marque. Les parts de marché tendent à se stabiliser et les bénéfices s'accroissent de manière substantielle.

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

- Phase 3 : la maturité

Les ventes augmentent plus lentement et finissent par se stabiliser. On opte pour une différentiation produit à travers les marques. Une concurrence intensive est en marche et c'est la guerre des prix. Le marché est saturé. Certains produits quittent le marché à cause de leur trop faible marge. La promotion est très répandue et une plus grande panoplie de média est utilisée.

- Phase 4 : le déclin (ou le re-lancement)

Le marché est ici en ralentissement. Des produits novateurs sont lancés ou les goûts de la clientèle se modifient par exemple. Les prix doivent être rabaisser. Par conséquent, beaucoup de produits doivent être retirés du marché. Les profits peuvent être accru par une réduction des dépenses marketing et autres coûts.

Ces étapes du cycle de vie doivent être en adéquation avec le type d'utilisateur, c'est-à-dire, nos futurs personas que nous verrons plus loin.

2) Identification de la cible du produit numérique

En effet, les 5 types de consommateurs selon le modèle de Rogers sont :

- Les innovateurs : à l'affût des nouveautés, ils sont d'un niveau social plus élevé que la moyenne, et veulent être les premiers à adopter un nouveau produit pour lequel le prix importe peu. Ils représentent autour de 2,5% de la population.
- Les précurseurs : Proches des innovateurs, mais plus frileux, c'est le cœur de cible pour le succès d'une innovation. Entre 13 et 15 % de la population, non seulement ils devraient permettre d'atteindre la masse critique de consommateurs, mais surtout ils vont critiquer le produit, propager cette critique et se valoriser socialement de la diffusion du produit.
- La majorité précoce : Les prix ont baissé, le produit a fait ses preuves, l'influence publicitaire fait son chemin, et ces 34 % permettent la diffusion de masse.
- La majorité tardive : La majorité de la population en est équipée, à force, ils s'y mettent...
- Les retardataires : Averses aux risques et aux modes, d'un niveau social et culturel plus modeste, leur budget leur permet d'accéder au produit quand les prix sont vraiment plus bas : ils représentent environ 12-18 % de la population.

Les démarches d'innovation pour créer son produit et amorcer le démarrage de son produit peuvent être définies à travers le lean start-up et l'innovation frugale.

3) Les outils pour bien démarrer votre produit numérique

- **Le lean startup**

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

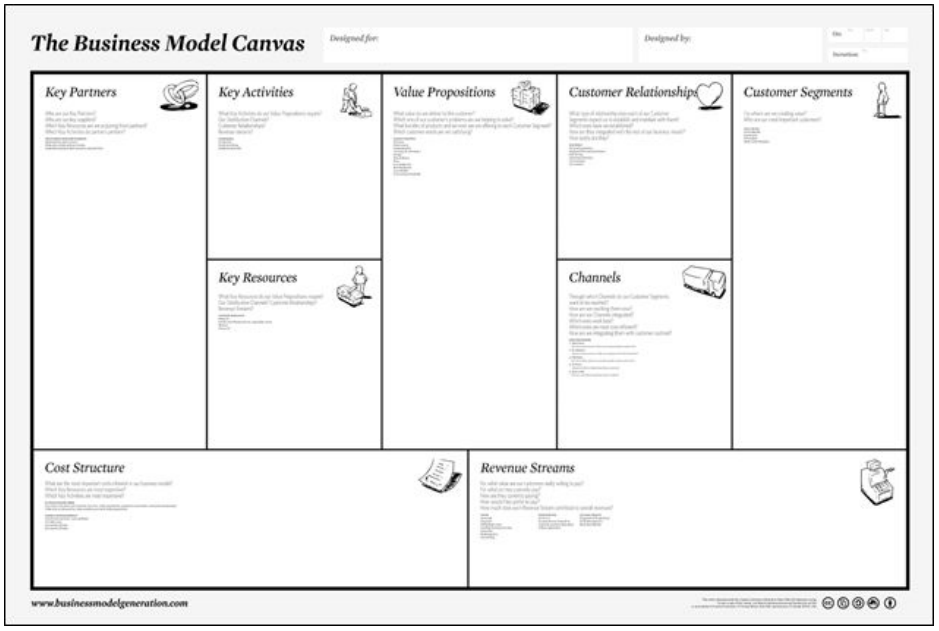
La démarche du lean start-up repose sur le triptyque : Build- Mesure – Learn

Vous trouverez ci-dessous les outils du lean startup :

Pitch elevator : L'objectif est de raconter le projet en 5 min. Je vous propose le pictch canva pour vous aider à le réaliser :

Le PITCH CANVA	
Simple phrase des changements grâce au produit sur le monde	
Douleurs & gains	Produit
La démo du produit	L'unicité du produit
L'attraction des clients	Business model
Investissement	Equipe
Fin de pitch avec une phrase de type « call to action »	
Pourquoi pas vous ?	

Business model canvas



MVP

Un MVP (Minimum Viable Product), Tout dépend du sujet auquel il s'applique. Pour autant tous les MVP ont une particularité commune : être focus sur un but unique. Un produit MVP doit avoir une fonction unique et spécifique. Un service MVP doit offrir seulement une prestation particulière et simplifiée. Une interface MVP doit être axée sur une fonctionnalité du site/application.

BY-NC-ND

Ségolène Porot - licence CC 4.0

- **Frugal innovation**

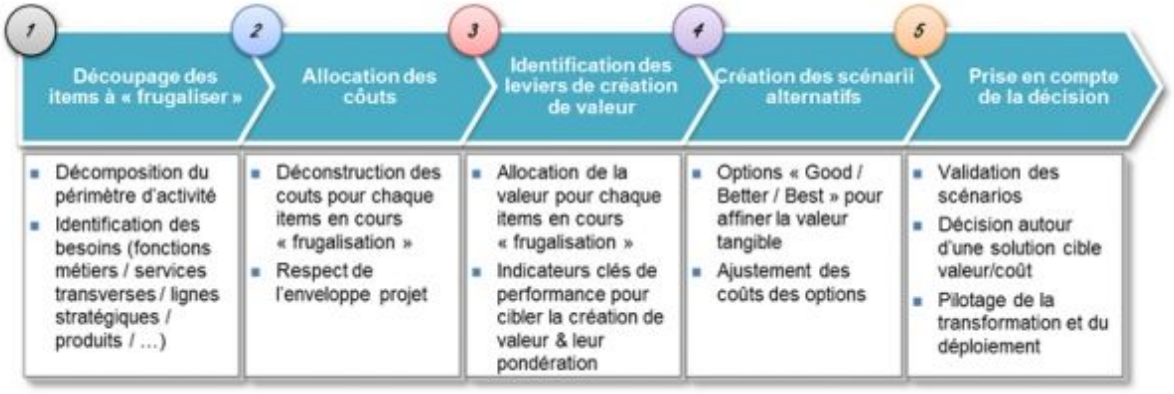
La Frugal Innovation est une méthodologie de priorisation par la valeur. Voici la démarche d'innovation frugale à travers 5 étapes :

- Découpage des items à « frugaliser »,
- Allocation des coûts,
- Identification des leviers de création de valeur,
- Création des scénarii alternatifs,
- Prise en compte de la décision

Un des livrables clés de l'approche "Frugal Innovation" est un outil de priorisation par la valeur.

Cette matrice de décision doit présenter a minima :

- Les options,
- Leur coût,
- Et leur contribution à la valeur
- La réalisation d'un arbre de valeur est également une force pour décomposer la création de valeur tangible.



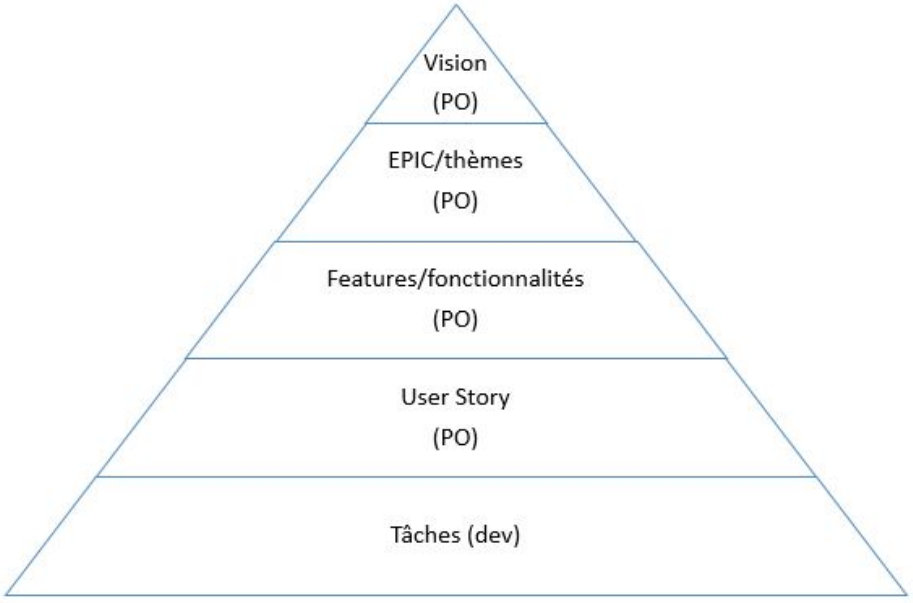
(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

B) Phase 1 : comment lancer un produit ?

Lorsque l'on souhaite lancer un produit, nous avons besoins de sens, de perspective pour dérouler les phases du produit. La phase de lancement est une des phases, à mon sens, les plus enrichissante et « enchantée » mais aussi c'est aussi une phase de doute, d'incertitude et de rapport au risque plus élevé.

Pour vous aider à mieux assimiler la construction d'un produit, la pyramide ci-dessous vise à hiérarchiser les axes et actions depuis la vision de votre produit (l'aspect conceptuel, abstraction) jusqu'aux tâches quotidiennes concrète de développement pour mettre en œuvre (l'aspect concret, pratique)



Pyramide de la vison aux tâches

Légende

PO : Sous la responsabilité du product owner

Dev : Sous la responsabilité du développeur

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

La première étape et non des moindre est de définir la vision de votre produit.

Je souhaite tout d'abord préciser que la vision du produit n'est pas spécifique à l'agilité. Cette dernière s'inscrit à part entière dans la vision stratégique de l'entreprise.

1) Qu'est-ce que la vision d'un produit ?

Il s'agit de l'ambition que vous vous donnez, là où vous voulez aller, ce que ce produit va faire ou apporter comme valeur lorsqu'il sera achevé.

La vision produit est la réponse opérationnelle à la stratégie globale de l'entreprise.

La vision produit est primordiale car elle permet ensuite l'illustration de cette stratégie à travers la roadmap du produit (et autres outils relatifs à la constitution du produit).

En pratique, pour définir la vision produit, vous pouvez dans un premier temps réaliser une étude préalable qui consiste à répondre aux points suivants :

- Quel problème essayons-nous de résoudre ?
- Pour qui résolvons-nous ce problème ?
- Quelle solution apportons-nous à ce problème ?
- Comment allons-nous lancer ce produit ? (go-to-market)
- Quel est le modèle économique ?
- Comment allons-nous mesurer le succès de ce produit (revenu, métriques) ?
- Comment avoir une vision produit claire ?
- Comment accorder les parties prenantes (marketing, finance, équipes techniques).
- Comment garantir la motivation des équipes ?
- Comment anticiper les choix technologiques en fonction de cette vision ?

Dans une équipe agile, la vision produit est encore plus importante que pour une équipe classique : avant chaque sprint ou avant le développement de chaque fonctionnalité, les membres de l'équipe doivent se demander si leur travail répond bien au "Product Statement". Si ce n'est pas le cas, c'est que l'énergie n'est pas utilisée à bon escient.

La vision pour un product owner est essentiel car :

- C'est l'essence du futur produit
- Cela permet d'exprimer de manière concise une idée forte
- Cela permet d'atteindre un but commun
- Cela donne une direction
- Cela facilite la créativité

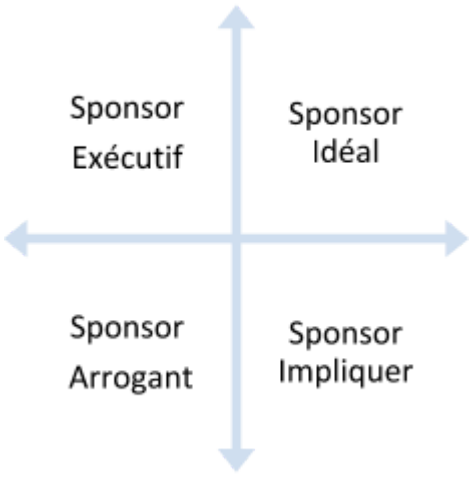
La vision produit doit également décrire la cible des utilisateurs (ou clients).

La vision produit doit être définit par le product owner et dans certains contextes le « Commanditaire » ou le « sponsor », et tout autres parties prenantes concernées (scrum master et développeur) par les objectifs stratégiques du produit. C'est dans ce contexte que

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

je souhaite vous proposer une cartographie des différents types de sponsor et les actions à entreprendre pour aboutir à une vision consolidée de votre produit :



La grille Intérêt-Pouvoir

Type de sponsor	Définir la vision Définir les enjeux Définir le périmètre du produit	Disponibilité
Le sponsor idéal	x	x
Le sponsor exécutif	x	x
Le sponsor impliqué		
Le sponsor arrogant	-	x
Le sponsor « sans pouvoir »	Résignation – Désengagement Sortie de projet- Changement de sponsor	

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Nous pouvons identifier 4 sponsors type et un sponsor à éviter si possible soit par la fuite ou par la lutte (changer de sponsor). Pour chacun de ces sponsors, vos pouvez identifier grpace à la grille intérêt pouvoir l'objectif et par conséquent le plan à mettre en œuvre :

- Le sponsor idéal : il faut le manager
- Le sponsor exécutif : il faut le satisfaire
- Le sponsor impliquer : il faut l'informer
- Le sponsor arrogant : il faut contrôler

Lorsque vous avez définit le type de sponsor, vous pouvez vous focaliser sur les différents enjeux de votre produit qui feront l'objet de votre communication aux parties-prenantes. La liste ci-dessous est exhaustive mais elle aborde les points classiques en stratégie.

<u>Formaliser les enjeux :</u>

- Stratégiques : chiffre d'affaires, reach, performance opérationnelle, ROI, position sur le marché…
- Usages : accès à l'information, expérience utilisateur, multi-distribution…
- Contenus : nouveauté du contenu, enrichissement, recommandation, croisement ou agrégation d'informations…
- Techniques : innovations ou évolutions technologiques…
- Internes à l'entreprise : amélioration de processus, de la qualité, de la productivité, économies…
- Internes à l'équipe projet : capitalisation sur une connaissance, une méthodologie, une technologie…

Ces enjeux permettent de clarifier l'écosystème à travers différentes thématiques dans lequel votre produit évolue.

Néanmoins, lorsque vous êtes en mission, vous pouvez vous retrouver face à 2 cas possibles:

Dans un premier cas, la vision du produit a déjà été définie par ailleurs, il suffit de la partager et de s'assurer qu'elle est claire pour tous.

Par ailleurs, je vous conseille de vérifier les éléments ci-après ont été défini et partagé :

- Les facteurs clé de succès ;
- Les KPIs qui permettront de mesurer l'atteinte des objectifs pour le produit ;
- Les principaux processus métiers intégré dans le product backlog ;
- Un premier backlog « macro » et l'ordonnancement des fonctionnalités du produit à mettre en place. Ce backlog « Produit » sera alimenté ensuite en continue durant toute la vie du projet.

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Je vous propose par ailleurs, un premier aperçu des prochaines thématiques pour gérer ce type de situation grâce à une liste de points à vérifier, à récolter des informations et analyser.

En effet, vous devez tout d'abord réaliser une phase d'audit à travers l'analyse d'objectifs & vision du projet, les objectifs explicites et implicites du client, l'étude du contexte, l'organisation, les RH, les SI, tester les IHM, les bases de données (saisir des données, vérifier leur stockage, le processus de traitement). La seconde phase est dédiée à l'analyse et à l'émergence des problématiques apparents et profonds.

Dans un second cas, la vision produit n'a pas encore été définie, il convient de la formaliser à travers les outils ci-dessous. En général, plus d'un seul atelier Vision & Enjeux est nécessaire obtenir une vision du produit claire.

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Lean Canvas :

- Segment utilisateur : qui sera la cible de notre produit ? Qui sont nos potentiels early adopters ?
- Problème : quel est le top des problèmes que nous cherchons à résoudre pour nous, early adopters supposés. Par exemple : structure de coût, de revenue
- Proposition unique de valeur : c'est l' "Unique Value Proposition Unfair Advantage + Customer Segments: la proposition de valeur est la raison qui doit pousser vos prospects à devenir des utilisateurs. En quoi résolvez-vous leurs problèmes ?
- Produit : Imaginer le produit à travers le « product market lean canvas »
- Solution : quelles sont les 3 grandes fonctionnalités qui vont répondre aux problèmes de nos early adopters ?
- Canal : quels sont les canaux gratuits et payants que vous pouvez utiliser pour atteindre vos futurs utilisateurs ?
- Métriques : quelles sont les métriques que vous allez mesurer pour valider ou invalider vos hypothèses ? (NPS, AARRR, chiffre d'affaires...)
- Coûts : quels sont vos coûts fixes et variables ?
- Revenu : quel est le modèle économique ? Comment allez-vous gagner de l'argent ?
- Avantages concurrentiels : qu'est-ce qui vous rend plus performant que vos concurrents pour traiter les problèmes identifiés ? (Connaissance technique, carnet d'adresse, marque).

Je vous propose également plusieurs outils et plus particulièrement des outils de type « Innovation games » pour définir la vision de votre produit.

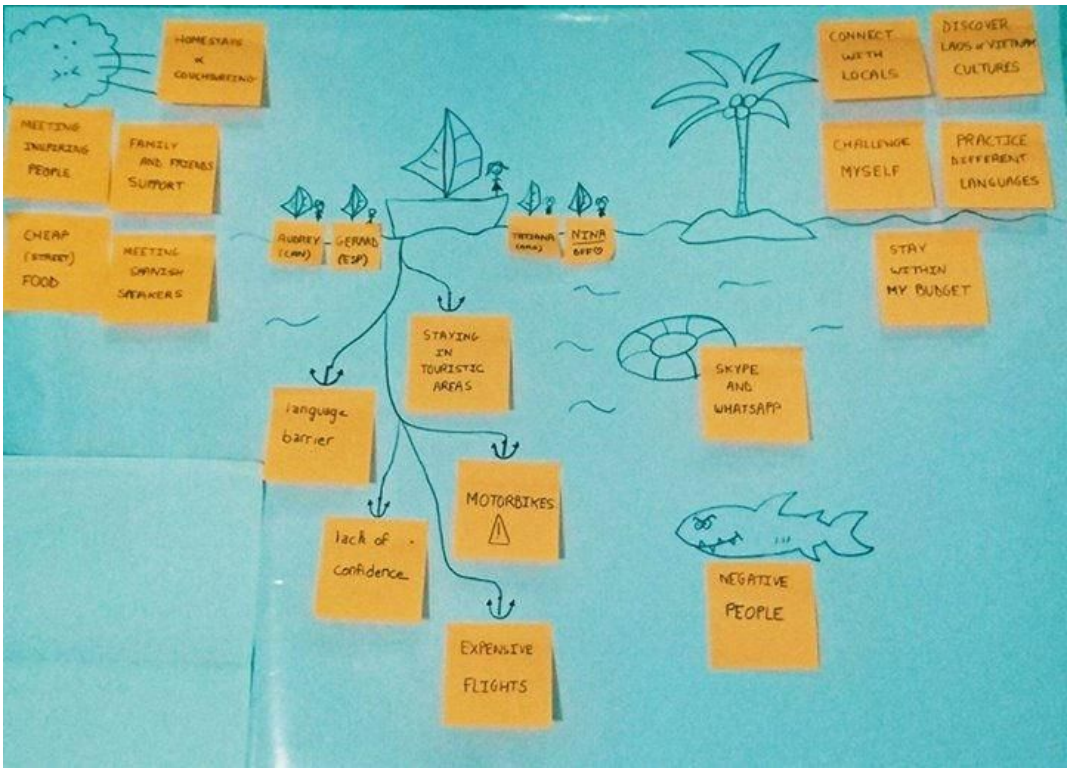
Product box : permet d'identifier les caractéristiques les plus excitantes d'un produit. Demandez à l'équipe de concevoir une boite qui va représenter votre produit comme s'il devait être vendu et apparaitre sur les rayonnages d'un magasin. Ils vont pouvoir habiller la boite avec des slogans, des coupures de magazines, des prix... Une fois réalisée, demandez à un membre de l'équipe de la présenter, de la vendre.

Remember the future : là, c'est très simple, il suffit de s'imaginer ce qu'il s'est passé dans le futur si tout s'est bien passé et que le produit a été un succès. Cela permet d'identifier les critères de succès, souvent vus des clients, des utilisateurs finaux.

Speed boat : Un bateau qui représente votre système, votre produit, votre projet ou votre équipe Le bateau peut se rendre au port ou sur son ile, il s'agit de dessiner le le but, l'état B dans une perspective de changement ou posséder telle ou telle caractéristique. Donner une position au bateau permet de présenter et peut le cas échéant nous donner une idée sur le chemin à parcourir Des ancres constituent les freins qui empêchent votre bateau d'avancer. Elle le ralentisse ou il s'agit des choses que vos utilisateurs n'aiment pas dans votre produit et nuisent à son fonctionnement optimal. Plus elles sont bas, plus elles sont fortes Le nuage/le vent qui représente les éléments qui le poussent et le font avancer, des éléments de motivation par exemple

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0



Prune the product tree: le principe est de définir un arbre dans lequel le tronc correspond au produit, les branches au fonctionnalités actuelles et les feuilles aux nouvelles fonctionnalités. L'objectif est de « tailler » le produit en fonction des besoins. Personnellement, je préfère un exercice de priorisations des fonctionnalités.

IN/OUT : une autre manière de déterminer le périmètre du projet est d'exprimer ce qui ni est pas : on peut donc construire la liste des fonctionnalités qui sont inclus dans le projet, et une autre liste des fonctionnalités qui n'y sont pas. Cet exercice est très efficace.

Vous pouvez compléter cette vision plutôt orientée stratégie par une vision orientée produit. Comme nous l'avons vu précédent, la stratégie impacte la vision produit et doit être intégrée dans celle-ci.

Le Produit CANVAS			
Nom	**But**	**Indicateurs**	
Nom du produit	*L'objectif global du produit*	**Qualité**	**Avancement**
Groupe Ciblé	**Image globale**	**Contraintes du produit**	**Détails du produit**

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Pour terminer, je vous propose ce tableau récapitulatif, et plus particulièrement le résumé en une phrase de votre vision produit.

Pour cela vous pouvez utiliser « **elevator speetch** », qui consiste à compléter le texte ci-dessus :

- Pour « les clients, utilisateurs »
- Qui ont « les besoins »
- Le « nom du produit »
- Une « catégorie produit »
- Pourront « bénéfice clé »
- Grâce à « les fonctionnalités/features clés »
- Contrairement « aux concurrents, au système legacy »
- Le produit pourra « élément clé de différentiation »

Enfin, vous pouvez compléter le tableau suivant qui intègre à la fois « l'elevator speech », la vision produit, le Product CANVA.

Phrase courte de résumée de la vision	Résumée de la vision		
Groupe ciblé	**Besoins**	**Produit**	**Valeur/ Objectifs business**
Quel est le segment marketing du marché sur lequel s'adresse le produit ?	Quel besoin le produit remplit-il ?	Quel est le top3, 5 des caractéristiques critique pour le succès du produit ?	Comment le produit va apporter de la valeur à l'entreprise ?
Qui sont les clients et les utilisateurs du produit ?	Comment le produit génère de la valeur ?	A quoi le produit ressemblera (macro) Quels sont ces points de vente unique ? Peut-on réaliser le produit ?	Quels sont les revenus du produit ? Quels sont ces coûts de structure ? Quels sont les canaux de distribution utilisés ? est-ce que cela va économiser de l'argent ?
Les utilisateurs et les clients avec leurs besoins Présentation des personas	L'expérience client souhaitée, le parcours utilisateur, les fonctionnalités du produit et les propriétés non-fonctionnel, Epic, stories, scénarios, workflow, design sketch, mock-up	Liste des freins (externalités) rencontrés, des contraintes du produit	L'objectif de la prochaine itération et éléments clés pour y parvenir. Les éléments sont priorisés et peuvent être définis en US

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Enfin vous pouvez préparer votre tableau de bord du produit à travers :

Le tableau de bord du produit	
Objectifs Les objectifs commerciaux du produit	**Clients** Les indicateurs relatifs aux clients (part de marché, taux d'adoption)
Financier Les indicateurs financiers pertinents (revenue, coût, profit)	**Equipe** Les indicateurs des parties prenantes (Engagement et motivation des parties prenantes)
Produit Les indicateurs relatifs au produit (variance de planning, état du produit)	**Processus** Les indicateurs processus (variance de planning, état du produit)
Contraintes fonctionnelles Les indicateurs du taux de couverture fonctionnel réalisé, en cours, à faire	**Contraintes non-fonctionnelles** Les indicateurs du taux de couverture non-fonctionnel réalisé, en cours, à faire Taux de risque

Même si elle peut évoluer dans le futur, bien cadrer la vision en amont reste le meilleur moyen de minimiser le risque d'un investissement humain et financier dans le développement d'un produit qui ne va pas trouver son marché.

Pour mener à bien votre vision, celle-ci est déterminée par un certain nombre de feature/fonctionnalité

Il s'agit d'une action particulière qui amène une possibilité supplémentaire. Fonction faisant partie d'un programme ou sous-programme ayant la capacité d'effectuer un calcul ou une tâche indépendamment du programme. Par exemple : Aujourd'hui, les smartphones sont les appareils proposant le plus de fonctionnalités. Ils peuvent notamment : Téléphoner, Envoyer des messages, Se connecter à Internet, Prendre des photos, Filmer des vidéos, Lire de la, musique, Lire des films, Vous guider via le GPS. Chaque élément est une fonctionnalité différente

L'outil principal que je vous propose pour prioriser vos features/fonctionnalités s'organise autour d'un atelier « **Buy a feature »**. Le but est de prioriser les fonctionnalités en demandant aux acteurs de les acheter. Une variante qui a ma préférence est de donner des gommettes (3 ou 5) à tout le monde et de demander de les placer sur les fonctionnalités qui ont leurs préférences. Ils ont le droit de poser plusieurs gommettes sur la même fonctionnalité.

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Vous pouvez ainsi vous créer la « **product roadmap** », ce tableau vous permet de lister et prioriser les fonctionnalités pour chaque release à venir.

Product roadmap			
Release n - Date	Release n+1 - Date	Release n+2 - Date	Release n+3 -Date

Vous pouvez détailler votre product roadmap à travers le tableau « **the go product roadmap** »

The GO Product Roadmap				
Date La date de release ou période	Date	Date	Date	Date
Nom Nom de la nouvelle release	Nom/version	Nom/version	Nom/version	Nom/version
But La raison pour laquelle la release est crée	But	But	But	But
Features/ Fonctionnalités Les fonctionnalités au niveau macro pour atteindre le but	Features/ Fonctionnalités	Features/ Fonctionnalités	Features/ Fonctionnalités	Features/ Fonctionnalités
Indicateurs Pour vérifier que le but a été atteint	Indicateurs	Indicateurs	Indicateurs	Indicateurs

2) Qu'est-ce qu'une User Story ?

Une User Story est une description simple et compréhensible d'une fonctionnalité à valeur

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

business, c'est-à-dire utilisateur d'un système.

La rédaction d'une User Story se fait en trois étapes

- Une description du besoin ;
- Une négociation en vue d'une définition du besoin ;
- Une définition des critères d'acceptabilité.

En pratique ces étapes se formalisent tout d'abord par l'approche des **3C**

- **C**arte : la user story tient sur une carte
- **C**onversation : discuter à l'oral
- **C**onfirmation : comment va-t-on valider l'US ? il s'agit des tests d'acceptante

Par ailleurs, je vous encourage à respecter la règle de qualité d'une User Story qui est définie par l'approche INVEST :

- Indépendante : celle-ci peut être implémentée dans n'importe quel ordre
- Négociable
- Avec de la valeur pour le client
- Estimable pour la prioriser et la planifier
- Suffisamment petite : avec une petite description
- Testable : on peut tester l'US

L'approche agile préconise la formalisation d'une User Story sous la forme :

- **En tant que** – Qui (utilisateur, client) ?
- **Je veux** – Quoi ?
- **De façon à**/afin de – Pourquoi ?

Je vous propose tout d'abord de répondre au « **Qui** » à travers l'identification des utilisateurs-types au début du projet et de les communiquer à tous les parties prenantes. On peut utiliser différents techniques en fonction de vos besoins tels que :

- Profil utilisateur : permet de définir les responsabilités et les tâches ;
- **Rôle utilisateur** : permet de décrire les contextes, les caractéristiques et les critères des rôles ;
- **Persona** : permet de décrire un utilisateur-type, une représentation fictive des utilisateurs cibles.

Dans le cadre de projets agiles, une combinaison des rôles utilisateurs et du Persona permet d'identifier les rôles existants et d'en faire des utilisateurs-types qui serviront de fil conducteur pour la rédaction des Users Stories. Vous pouvez utiliser le modèle de Persona ci-dessous :

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Photo et nom, prénom	Détails	But
A quoi ressemble la « persona » ? **Choisissez un prénom et un nom réaliste**	Quels sont ses caractéristiques et comportements ? Démographique : Age, sexe, job, revenu, Psychologique : style de vie, niveau social, personnalité Comportement : modèle d'usage, attitude, fidélité envers une marque Seulement une liste de détail pertinent	Quel problème la persona souhaite résoudre ? Quel est le bénéfice qu'il souhaite ? Pourquoi la persona souhaitera utilisé ou acheté ce produit ?

Une persona peut être également décrite par d'autres approches telles que :

Approche n°2	Approche n°3
▪ Archétype en % ▪ Démographique, social ▪ Habitudes (comportement, style de vie) ▪ Expertise (device, plateforme) ▪ Cuture (référence, wirframe) ▪ Motivation ▪ Must do – must never ▪ Les objectifs UX	▪ Nom ▪ Profession ▪ Age ▪ But personnel ▪ Intérêt : passe-temps, lire, écrire sur quel sujet ▪ Pouvoir : influence de la persona ▪ Besoins : requis, manquants ▪ Comportements : routines, free-time ▪ Valeur : Qu'est-ce que « prendre soin/le care » pour cette persona par exemple ? ▪ Aspiration : quel genre de personne voudrait-elle être ?

Par ailleurs, pour obtenir des personas complètes, voici « l'empathy map » qui permet de déceler les insights de vos personas :

Dit	Pensée
Fait	Ressentis (feels)

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Ainsi pour chaque persona vous pouvez définir un parcours avec :

Légende :

La spécification de la thématique traitée actions, émotion, thinking selon le timing de cette action pour atteindre le point B

Voici une seconde « empathy map » possible :

Pensée & ressenti	Senti (feel)
Vu & entendu	Dit & fait
Douleurs	Gain

Pour obtenir ces personas, je vous propose le processus suivant :

- Définir les objectifs, la cible, le périmètre des personas
- Collecte d'information de toutes sources
- Conduire des entretiens avec des utilisateurs
- Chercher des modèles, des tendances (des patterns)
- Créer des scénarios pour les personas
- Tester les personas, le produit-service, les opportunités, douleurs
- Faire le parcours utilisateurs avec les personas
- Analyser les opportunités, les problèmes, les recommandations, mesurer les progrès, créer une roadmap

Il faut toujours utiliser le minimum de persona pour un produit sinon vous pouvez définir « la primary persona » pour faciliter l'organisation, la priorisation.

Pour répondre au « **Quoi ?** », cela revient à formuler la question suivante : Comment exprimer le besoin de mes utilateurs pour rédiger mon User Story ?

Pour cela, voici quelques méthodes **de conduite d'entretien d'expression de besoin,** car ce n'est pas toujours si simple de savoir par où commencer ou de savoir comment s'y prendre.

L'objectif de ces entretiens est de faire émerger une problématique, des idées pour formaliser les US. En général, les personnes ne parlent pas d'emblée du vrai problème ou pense parler du problème ou du besoin mais il s'agit le plus souvent d'une perception erronée. Alors que faire pour définir un besoin. C'est pourquoi, plusieurs étapes sont

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

nécessaires pour étudier un besoin à travers plusieurs approches :

Approche « short »,

Vous pouvez utiliser cette démarche lorsque vous êtes poussés par des contraintes de délai, des problématiques trop complexes à définir au cours d'un seul atelier :

1. Réunir toutes les parties prenantes (quel que soit leurs intérêts, leurs contradictions)

2. S'assurer d'échanger sur le « bon problème » :
    1. Noter les idées des personnes : quel est votre pb ? autant de pb que de pers
    2. Centraliser les réponses par thème
    3. Définir le périmètre du besoin : périmètre du besoin (ajouter, modifier, supprimer ou uniquement supprimer des contenus.)
    4. Analyser les impacts du pb sur : les applications SI existantes et futur

3. Faire une représentation schématique : apporter de la valeur
    1. Réalisation de diagramme de séquence (transmission d'information), de diagramme d'état (principe du cycle de vie), diagramme de classe (modèle de donnée)
    2. Impact mapping (à réaliser pendant l'entretien à coréaliser lors de l'entretien. L'outil est expliqué **pageX**)

Approche « long » : besoin de synthétiser des éléments d'information pour une restitution

1. Travail préparatoire à réaliser
    a. Création de scénario (d'usage de bout en bout ex : créer une CVIV)
    b. Création de business case (cas d'usage et US associée)

2. Mettre en place un atelier (prévoir 1h30/2h – 8 personnes)

3. Poser les questions d'un parcours client (customer journey) selon le scénario et business case associé. Il s'agit de décrire le parcours utilisateur.

    - Il existe différents types de parcours selon les thématiques tels que : le parcours émotionne, le parcours de distribution, le parcours front/back, le parcours à travers le cycle de vie du client, le service bluesprint (parcours avec interaction sur le marché), le heat map (parcours avec les priorités à adresser)

4. Faire une représentation schématique : apporter de la valeur à travers le Story maps (outil présenté **page X**). Vous pouvez également présenter un tableau de synthèse tel que celui-ci :

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Persona		
	Etape 1 du parcours utilisateur de la persona	Etape 2 du parcours utilisateur de la persona
	Objectif de l'étape 1	Objectif de l'étape 2
	Pensée/feeling	Pensée/feeling
	Indicateur 1, 2,3	Indicateur 1, 2,3
	Opportunités	Opportunités

Approche atelier créatif

Vous pouvez opter pour des ateliers plus créatifs d'une heure en moyenne, selon le profil, les attentes et les besoins de vos clients.

L'idée est de miser sur l'approche « innovation game » pour faire émerger les idées ou mieux contextualiser la problématique et trouver une solution pertinente.

En général ces ateliers sont plutôt mis en œuvre dans la phase de croissance que l'on souhaite la plus continue et longue possible. Vous trouverez des outils pour les ateliers page **X**.

Enfin, la dernière étape consiste à **valider le besoin.** Il faut définir les **critères d'acceptabilités.** En effet**,** vous abordez ce sujet lors des négociations pour une User Story. Vous pouvez les déterminer à travers les questions : *Comment ? Que faire si ? Quand. Concrètement*.

Enfin, vous pouvez valider le besoin de la façon suivante (**Gherkin***) :

- **Etant donné** une situation *(votre jeu de données)*
- **Quand** se produit *(les étapes du cas de test)*
- **Alors** j'obtiens *(le résultat du test)*

Enfin, pour ceux qui ont des doutes, ces critères d'acceptation sont nécessaires car ils permettent de Savoir quand s'arrêter de développer, c'est aussi un outil de collaboration pour une compréhension globale et partagée enfin c'est une donnée réelle pour mesurer l'avancement du développement

Note : le Gherkin est une syntaxe utilisée à la base dans l'outil Cucumber (mais également dans JBehave) pour faire du BDD ("Behaviour-Driven Development"). Le BDD permet de décrire facilement des informations techniques une personne non-technique en intégrant une dimension comportementale, des phrases compréhensibles pour traiter l'aspect fonctionnel (ex : les cas de test)*

Par exemple les critères d'acceptation peuvent être rédigés de la façon suivante :

Exemple 1 :

BY-NC-ND

Ségolène Porot - licence CC 4.0

- SI « Le nom de l'utilisateur existe déjà »
- Lorsque « L'acheteur en ligne clique sur « Créer un compte »
- Alors « Faites apparaître un message d'erreur disant « Ce nom existe déjà »

Exemple 2 :

- SI « Faites apparaître La liste des éléments du panier sont affichés »
- Lorsque « L'acheteur en ligne clique sur « Supprimer du panier »
- Alors « Faites apparaître un message de confirmation disant « Etes-vous sur de vouloir supprimer cet article ? »

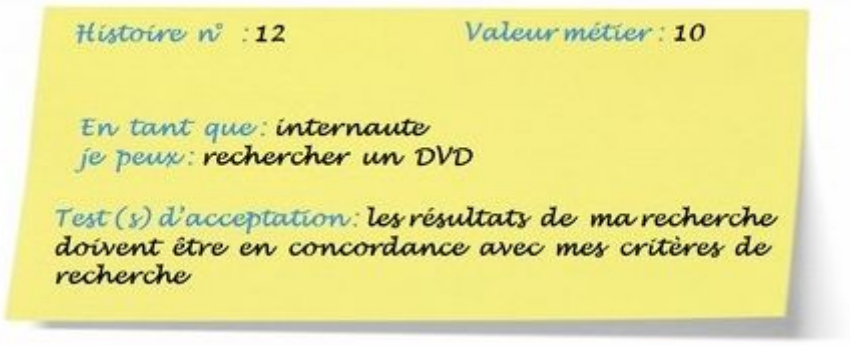
Maintenant que ces étapes ont été réalisées vous obtenez un besoin validé traduit en US comme celle ci-dessous :

Enfin pour **valider une US au niveau de l'équipe Scrum, du Scrum master** et dans une autre mesure des parties prenante, il faut que votre US remplisse les critères définis par la « **definition of done** ».

Il s'agit d'un ou plusieurs indicateurs de qualité qui permettent de valider ou d'invalider de façon objective les travaux produits par les équipes de développement.

Il s'agira pour valider une user story au sens DoD de valider les critères de type :

- Intégration avec succès dans la Plate d'Intégration Continue ;
- Existence de tests automatisés ;
- Validation de la qualité technique (relecture de code) ;
- Conformité vis-à-vis des conditions d'acceptation et des cas de tests définis dans la User Story.



Concrètement, la notion de "done" pour des travaux peut être très variable en fonction du contexte du projet, de la culture des équipes.

Pour terminer, je vous propose cet outil « user story check list » qui peut vous servir pour différents ateliers ou pour une fiche récapitulative de vos US pour votre product backlog.

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

USER STORY CHECK LIST

Titre	Créer une story		
ID	AD1		
Scénario	En tant que «rôle », je veux «but » pour « justification business »		
Estimation	Nombre de point	**Priorité**	Haute/1

Critère acceptation

Definition of Done
Ex :

Pre-requis

Exemple d'US
o Au format texte o Au format table de données

Règles métiers

Impacts sur fonctions existante/ architectures/Systèmes d'information

Référence à spécifications/maquette/diagramme
Ex : personna, diagramme de sequence (flux), mock-up

Cas de test détaillé

3)

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

3) Les outils consacrés à votre produit

- **Le product backlog**

Les User Stories sont priorisées dans un product backlog pour définir l'ordre de réalisation pour le développement.

Il est important de noter que plusieurs outils sur internet sont disponibles pour réaliser votre product backlog. Par ailleurs, le graphique ci-dessous vous permet de cibler l'outil qui correspond au mieux à vos besoins.

En agile, il n'y a pas de documentation, c'est-à-dire de fonctionnelles détaillées L'outil de collecte s'appelle le product backlog où figure les fonctionnalités (les features). Ce dernier est accessible à toutes les parties prenantes.

Le product backlog est donc une liste des fonctionnalités attendues d'un produit. Cependant, au-delà de cet aspect fonctionnel, il contient tous les éléments qui vont être nécessaire pour construire le produit. Ces deniers doivent être ordonnés par le product owner.

On peut prioriser les US du product backlog et par conséquent prioriser ce dernier grâce à la méthode **MoSoCoW**. Il s'agit d'un outil de priorisation simple et efficace

- **Must :** identifie ce qui est indispensable pour que l'application soit opérationnelle, indépendamment de tout confort.
- **Should** : identifie ce qu'il faudrait vraiment avoir pour que l'application soit utilisable dans de bonnes conditions
- **Could :** Introduit un confort d'utilisation pour faciliter l'expérience des utilisateurs finaux
- **Won't** : identifie des options qui seraient sympathiques, mais que, lucidement, on n'aura pas le temps/budget de réaliser !

Le product backlog est sous la responsabilité du Product Owner en revanche, toutes les parties prenantes peuvent collecter des éléments, mais c'est le product owner qui les valide pour intégrer le product backlog et prioriser.

Le backlog est élaboré avant le lancement des sprints, dans la phase de préparation (ou sprint 0). Sinon, lors du sprint planning, l'équipe, le product owner et le scrum master décident des US qui seront réalisés.

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Le product backlog est un pilier pour le product owner, ce dernier est utilisé pour construire :

- Le product backlog permet de produire le planning de release, c'est à dire le sous-ensemble des éléments du backlog qu'on estime finir pour la date prévue de release ;
- Mettre en place la réunion « product backlog refinement » ou « product grooming » pour affiner les US ;
- Le burndown chart de release, qui est un graphe, mis à jour à la fin de chaque sprint, montrant la tendance de l'avancement dans la réalisation des éléments du backlog de produit ;
- Le Backlog de sprint

En résumé, le product backlog permet de lister et hiérarchiser les fonctionnalités à produire et faciliter la planification des sprints et des releases.

Vous trouverez ci-dessous un exemple de product backlog :

ID	Priorité	US	Critère d'acceptation	Estimation	Release	Sprint	Statut
77	2	En tant que « » je veux que « » Afin de « »		5	4	6	terminé

Le product backlog peut avoir un cycle de vie similaire à celui du produit plus ou moins étalé dans le temps. C'est un document qui évolue et doit être mis à jour par le product owner.

Concrètement, le product owner raffine régulièrement ses User Stories au cours des sprints en découpant les stories de grandes tailles en stories de plus petite taille.

Puis, les US sont au sein du product backlog sont estimées lors du sprint planning (ou encore le planning poker). Il n'y a pas d'impératif d'unité de mesure pour estimer vos User Story. Néanmoins, certaines techniques se distingue.

La plus courante est l'estimation en points relatifs L'équipe choisit une User Story étalon à laquelle il affecte un nombre de points arbitraire. Les fonctionnalités suivantes seront estimées par rapport à cet étalon : une User story valant 2 points représente deux fois plus de travail qu'une valant 1 point. Sinon vous pouvez utiliser les premières valeurs de la suite de **Fibonacci (1,2,3,5,8,13,21)** pour éviter les valeurs proches et garder des fonctionnalités d'importance acceptable. Par ailleurs, vous pouvez utiliser les tailles de T-shirt : S, M, L, XL. A vous et à votre équipe de définir l'unité de mesure qui lui semble la plus pertinente.

Enfin, chaque fonctionnalité est alors découpée en tâches qui pourront être réalisées par

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

différents membres de l'équipe.

Les US sont ensuite associées à certain nombre de tâches par les développeurs :

US #	**Estimé**	**Dev :** **QA**
Description de la tâche		**Rev**

Pour finir, je vous propose une liste pour éviter les erreurs et surtout vous proposez des astuces pour réaliser et suivre votre product backlog.

Les erreurs à éviter pour votre product backlog	Astuces
Avoir plusieurs backlogs pour un seul produit (ou d'autres sources parasites) Ne pas partager le backlog avec toute l'équipe Ne pas faire vivre le backlog pendant le projet Confondre avec le backlog de sprint en y mettant des tâches Avoir plus de 50 éléments à faire dans le backlog Le responsable produit hiérarchise le backlog au début du projet, mais ne l'adapte pas en fonction du feedback des développeurs et des parties prenantes. L'équipe limite les tâches du backlog aux seules tâches qui concernent directement le client. Le backlog est conservé sous forme de document stocké en local et peu partagé	Conservez tout au sein d'un même outil pour garder trace des bogues, des exigences et des tâches y compris pour les tâches de développement, Lorsque le backlog dépasse les capacités à long terme de l'équipe, les tickets que l'équipe ne traitera jamais peuvent être clôturés (noter les spécifiquement). Un backlog = une équipe = un produit

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

- **Quelques outils pour créer votre product backlog :**

L'outil préconisé si vous êtes en sprint 0 ou si vous souhaitez tout simplement mettre en place l'outil. Il s'agit du Story mapping :

C'est l'objectif à l'issue de cet atelier où l'outil est utilisé.

Pour cela vous devez identifier :

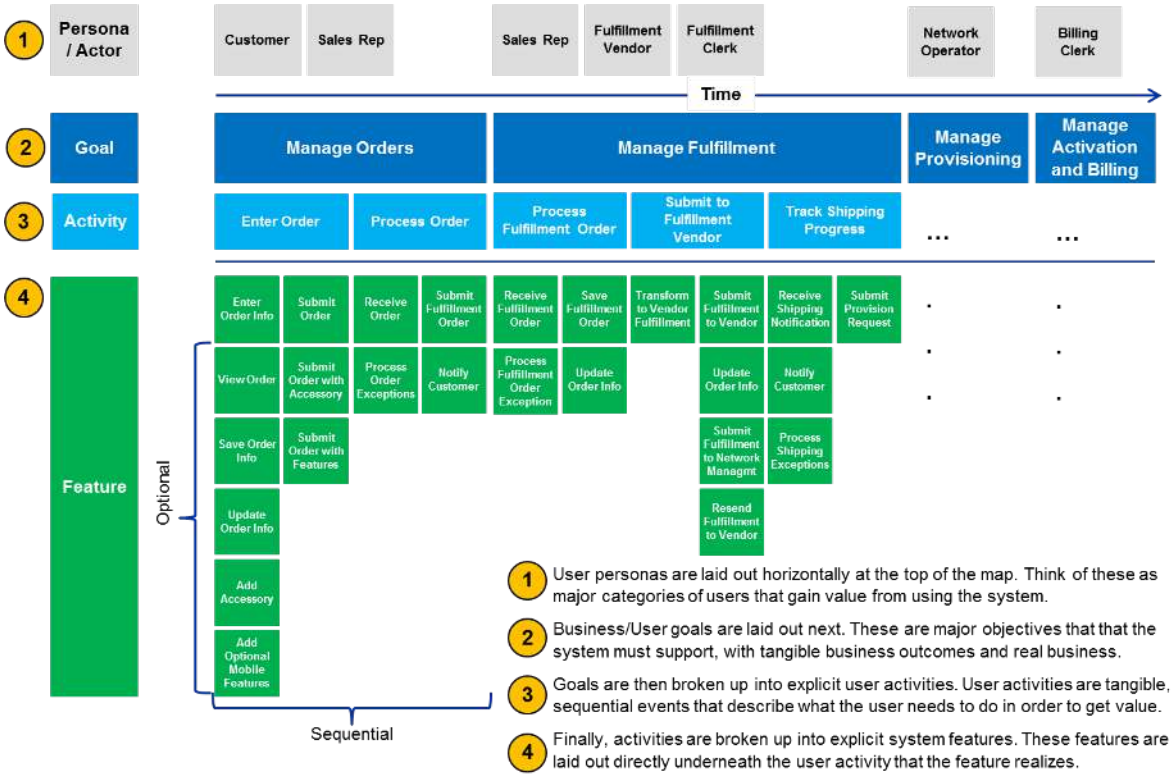
a) Définition des acteurs du Story Mapping
   - Les participants de la réunion ont chacun leur vision du produit à développer

b) Définition d'utilisateurs (persona) :
   - Qui va utiliser l'application ?
   - Quels sont leurs rôles ?
   - Quelle sera leur utilisation de l'application, leur approche fonctionnelle ?

c) Définition de scénariis utilisateur

d) Définition des priorités

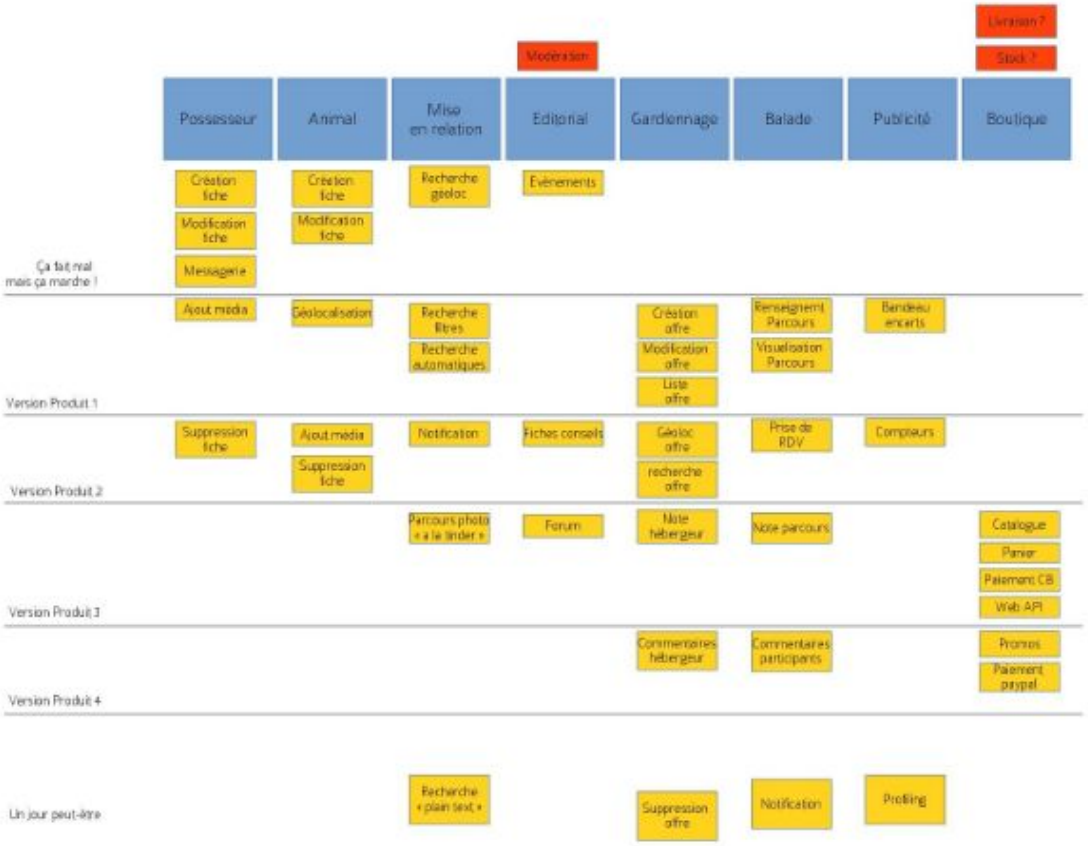
Le document ci-dessous permet une illustration plus visuelle des éléments caractéristiques de cet outil.



(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Exemple de story map (source Pablo Pernot – Are)



Source : Pablo Pernot

- **Burndown chart**

Le burndown chart montre ce qui reste à faire, mais on ne voit pas les évolutions de périmètre. Le responsable produit détermine la priorité des tâches dans le backlog, tandis que l'équipe de développement détermine la vélocité à chaque étape du backlog.

Pour construire ce graphique, il faut pour chaque sprint, collecter 2 valeurs pour construire ces courbes. Les valeurs sont estimées en points (les fameux story points), donc seules les stories qui ont été estimées sont prises en compte. Il permet d'estimer la vélocité, c'est-à-dire, combien de point ont été délivré sur les US au cours du psrnt. Cela varie d'un sprint à un autre. Cette variance peut être plus important lorsque l'équipe est nouvelle ou quand des changements sont réalisés. La vélocité la plus «réelle » correspond à la moyenne de la vélocité des 3 derniers sprint.

Concrètement vous pouvez obtenir le tableau suivant :

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

- L'axe X représente la ligne du temps du projet. L'unité dépend de la fréquence de ré-estimation du reste à faire. Pour un burndown d'itération, l'unité est souvent le jour, pour un burndown de release, l'unité est le sprint (le reste à faire pour compléter la release est recalculé à chaque début d'itération).
- L'axe Y Le travail qui doit être complété pour terminer le projet, représenté en temps ou en points, est introduit à travers l'axe Y.

Tableau de calcul à l'origine d'un burndown chart :

Itération	**Date de début**	**Back-log**		**Complexité**	**Vélocité**		**Reste à faire**	**Date prév. de fin (théorique)**
		Début d'itération	**Fin d'itération**	**Planifiée**	**Réalisée**	**Glissante**		
Numéro d'itération	Date	Points planifiés au total dans le product backlog au début sprint	Points réalisés au total dans le product backlog à la fin sprint	Points affecté complexité au total dans le product backlog pour le sprint	Points réalisés de complexité au total dans le product backlog à la fin du sprint	Moyenne (vélocité réalisé N/ vélocité réalisée N-1)	Points réalisés au total dans le product backlog à la fin sprint – SOMME (vélocité réalisée N/ vélocité réalisée N-1	Date de début+ *Durée du sprint (reste à faire N/ vélocité glissante N

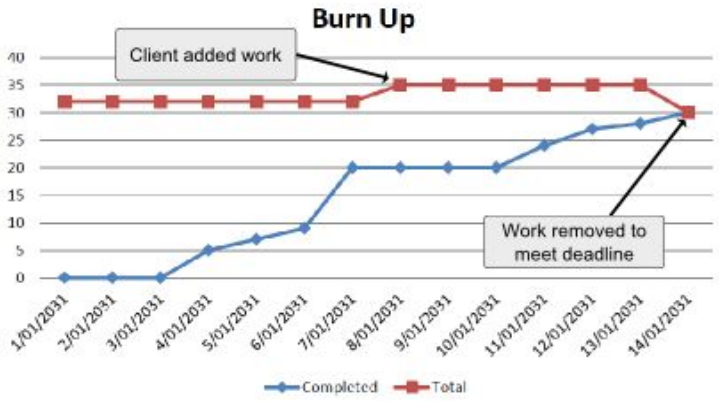
- **Burnup chart**

Il permet de suivre l'avancer d'un sprint sous le format suivant:

1. La ligne de travail total (le scope du projet)

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0



2. La ligne du travail réalisé

Le projet sera terminé une fois que les 2 ligne se rejoindront à un point commun. L'axe vertical est la quantité de travail (tâche, heure estimé, points de story). L'axe horizontal, c'est le temps (en jours).

C) Phase 2 : comment assurer une croissance continue du produit ?

Le contexte est le suivant, vous avez réussi à créer votre vision, votre product backlog et vous avez comme objectif de faire évoluer votre produit. La question est comment faire évoluer mon produit ? quelles sont les pièges à éviter ? quels sont les outils à ma disposition pour améliorer ma créativité en tant que product owner mais aussi de l'équipe Scrum et du Scrum master ?

1) Planifier des réunions clés de suivi de votre product backlog :

- **Planifier le Product Backlog Refinement**

Tout d'abord, je vous propose de miser sur le Product Backlog Refinement ou « product backlog grooming » pour une croissance continue de votre produit. Nous l'avons vu dans les tâches affectées au product owner.

En fait, cette tâche arrive plus rapidement que l'on peut l'imaginer. En effet, le product backlog refinement a pour objectif de :

- Permettre d'avoir les US "ready"
- Ajouter du détail au US
- Estimer les US
- Identifier les risques et problématiques avant le sprint planning
- Mettre à jour les priorités des users stories

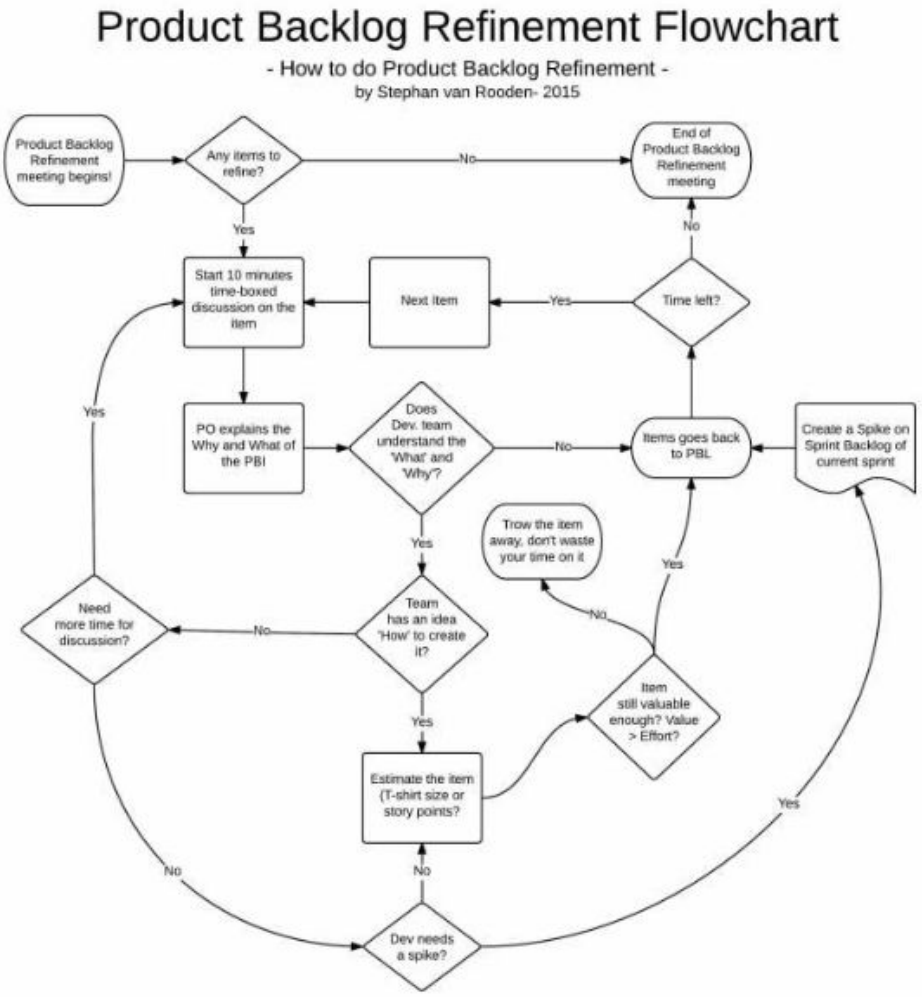
La fiche récapitulative du Product backlog Refinement

Product Backog Refinement Memo			
Quand :	durant le sprint – 1	**Durée** :	30 min

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

	fois par semaine		
Qui :	l'équipe de développement, le scrum master, le PO, les parties prenantes(option)	**Agenda type**	Identifier les US à affiner Clarifier les besoins pour les US Identifier les questions, pb, pour ces US et assigner un owner pour chaque Découper les US trop importante Echange sur les priorités des US et mettre à jour si nécessaire Estimer les US
Input	Le product backlog Les US check list	**Output**	Le product backlog détaillé



Source : van-rooden

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Pour réussir les différentes phases pour réaliser votre product backlog refinement, vous pouvez réaliser un KANBAN agile pour visualiser l'avancement de vos US pour permettre de contrôler l'avancement (le rythme) des US Ready

Cela consiste tout simplement à réaliser un tableau :

Planifié (to do)	**En progression (in progress)**	**Fini (done)**

Je vous propose quelques outils pour formaliser et conduire votre product backlog refinement: le Design studio, le Story mapping (déjà présenté à la page), l'impact mapping**,** le Découpage des US, le story-writing workshop appelé aussi « User story grooming session

2) Des actions spécifiques pour votre product backlog

- **Découper les US**

Lors du découpage, ne tenez pas compte des performances dans un premier temps, à la place préférer une pré-étude : comment marche le système ?

a. Par étape (je m'inscris, je me connecte)
b. Par règles métiers
c. Simple/complexe
d. Variation des données (en français/en anglais)
e. Méthode de saisie des données
f. Opération : création, mise à jour, suppression

On peut aussi opter pour le « Découpage vertical » est un travail (tâche) qui permet de délivrer un changement de valeur dans le comportement d'un système de manière à travailler à travers plusieurs couches d'architectures pour implémenter le changement. Quand le découpage est « done », le système permet d'apporter une valeur concrète à l'utilisateur. Cela est à contraster avec « le découpage horizontal », qui réfère au travail de tache à travers le changement

- **User story grooming session**

Je vous propose de planifier l'User Story Grooming Session. Cette réunion a pour objectif d'écrire les User story en cours et à venir

MEMO User story grooming session			
Quand :	1 fois par sprint (si toutes les 2 semaines)	**Durée** :	1 heure
Qui :	utilisateur, équipe de dev, scrum	**Agenda type**	Revoir la priorisation du product backlog

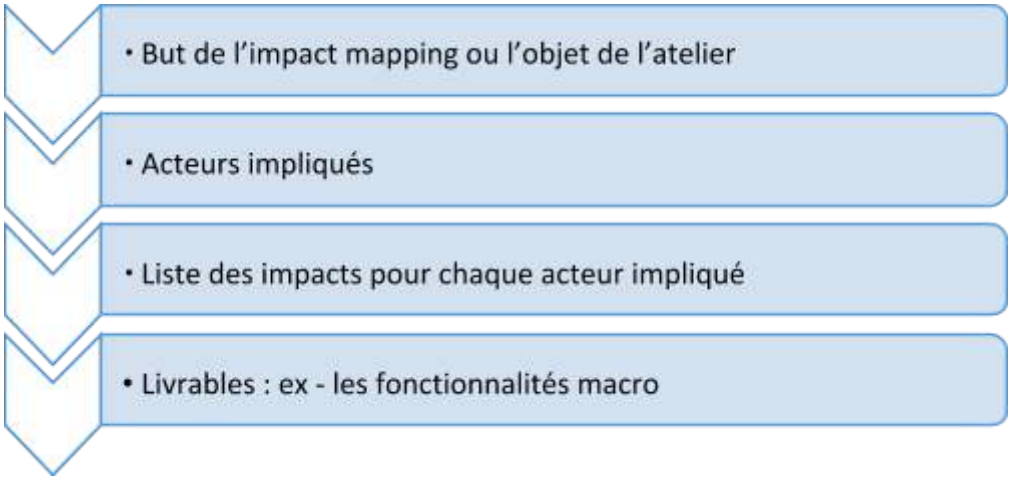
(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

	master		Passer en revue les US des utilisateurs en détail Finaliser les questions ouvertes Perfectionner les mokups Décrire et vérifier les flux d'utilisateurs
Input	Le product backlog priorisé US check-list préparé/pré-rempli	**Output**	US Check-list validé Compte-rendu (points d'action clairs)

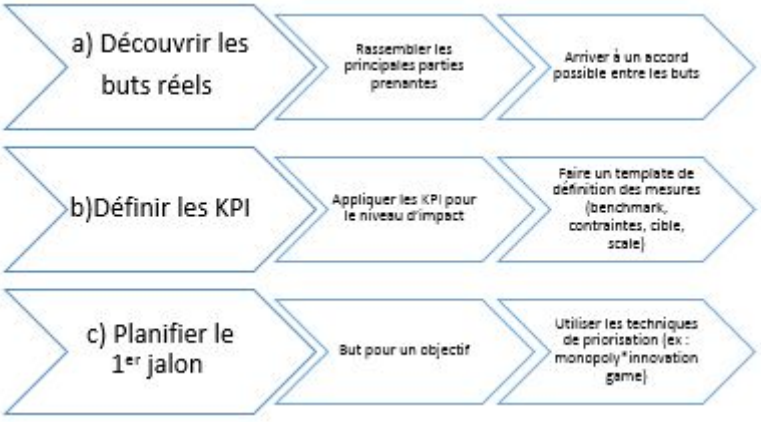
- **Impact mapping**

L'impact mapping se formalise de la manière suivante :



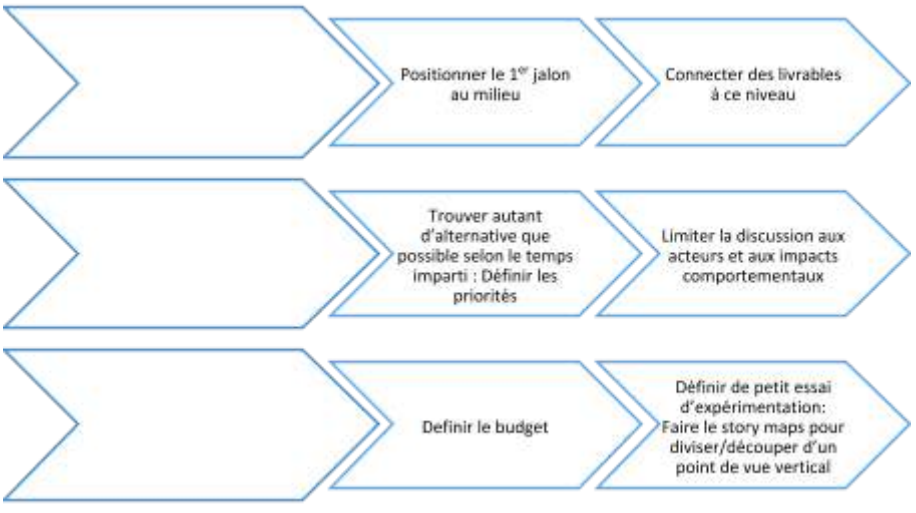
Le processus d'impact mapping se déroule de la façon suivante à travers 2 phases :

La phase 1 : Préparation



(cc) BY-NC-ND

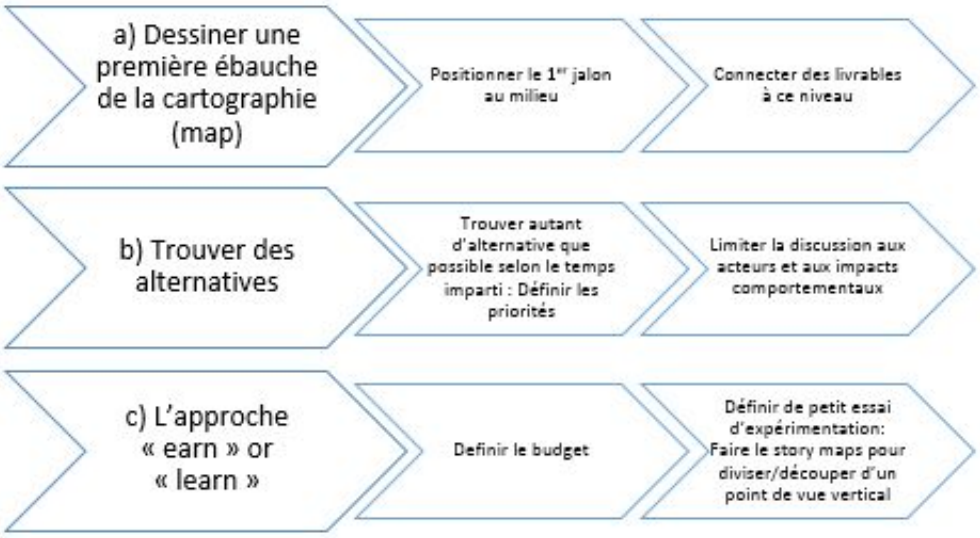
Ségolène Porot - licence CC 4.0



(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

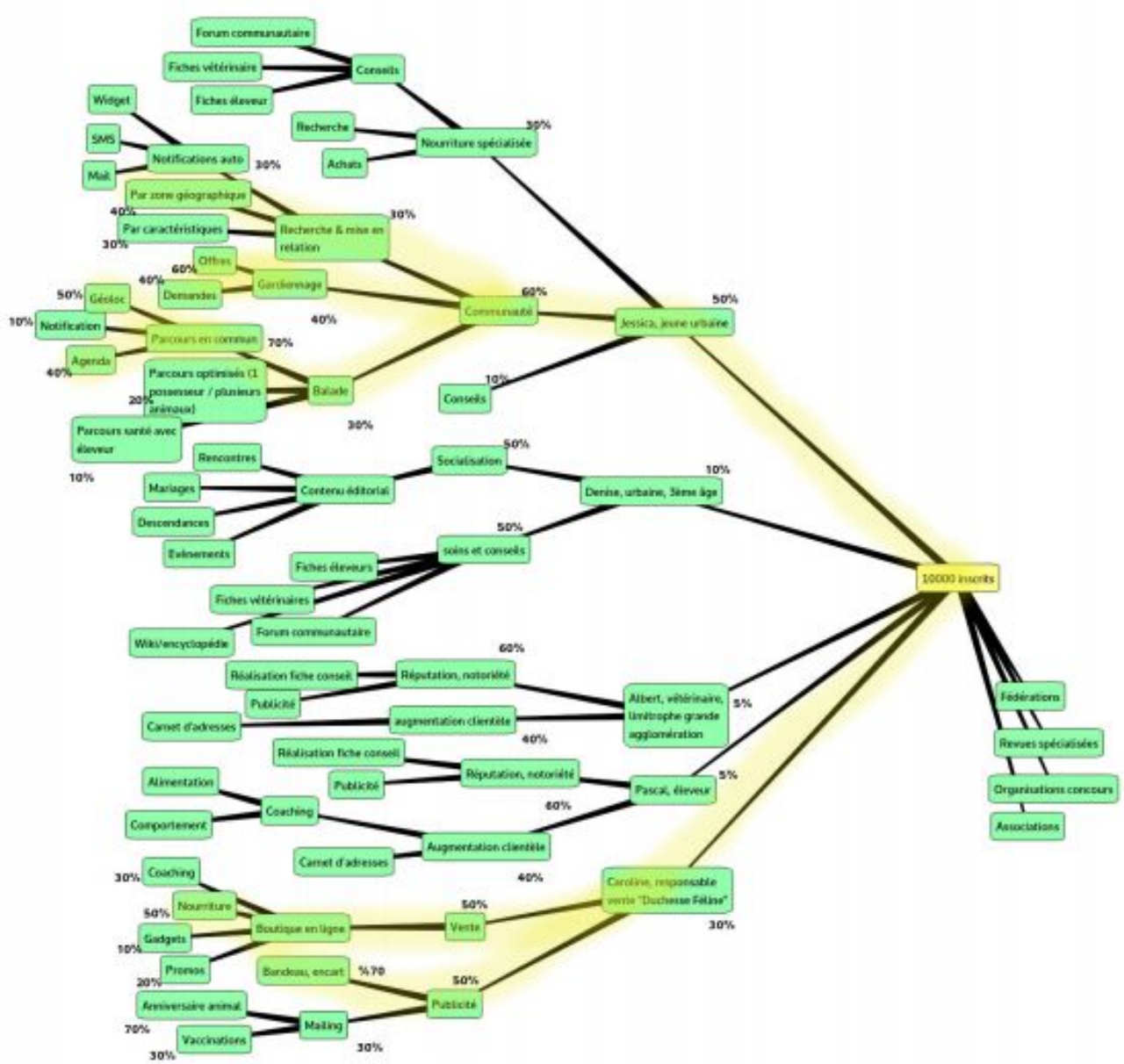
La phase 2 : Mapping



(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Exemple d'impact mapping



(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Vous pouvez utiliser en complément de l'impact mapping le **CQQCOQP** qui correspond aux questions Comment, Quoi/Qui, Combien, Où, Quand et Pourquoi. Utilisez-le pour vous assurer qu'un sujet a été bien couvert.

- **Design Studio**

Objectif :

- Design l'interface du site, application mobile à travers un exercice de sketching (dessin) ou posture ergonomique du produit
- Intégrer des mini design studio tout au long des développements, au cours des sprints, dans les Backlog Refinement ou des workshops distincts, plus courts sur des problématiques précises…

Processus de réalisation :

- **Temps 0** : Ouverture (10 – 30 min) :
  règle du jeu, contexte, Vision produit, Personas, Elements de Backlog ou Problème à résoudre

- **Temps 1 (30 – 40 min) :** 1er itération Design individuel : 8-10min
  Présentation + team critique : Itération 1 avec Sketching Solo (rapide, 6 écrans ou enchaînement en 8 minutes), puis PITCH de chaque personne à l'équipe (3 min) pour défendre et présenter ses solutions puis CRITIQUE (constructive) de l'équipe (2 min)

- **Temps 2 (30 – 40 min) :** 2ème itération Design individuel : 8-10min
  Présentation + team critique : Itération 2 (le REMIX), sur la base de la 1ere itération. On tient compte des options et critiques des autres avec Sketching Solo (rapide, 6 écrans ou enchaînement en 8 minutes), puis PITCH de chaque personne à l'équipe (3 min) pour défendre et présenter ses solutions puis CRITIQUE de l'équipe (2 min)

- **Temps 3 : (45-60min) : 3ème itération** Team design :
  Collaboration pour un design unifié, Présentation (inter-équipe **5 min)** Itération 3 : on construit collectivement la solution qui GAGNE :) puis on la présente à d'autres si nécessaire

- **Temps 4 : Clôture**
  Au démarrage des projets, on se lance dans des design studio de 3, 4 heures qui vont impliquer le plus de personnes possibles (spécialistes de l'interface, développeurs, testeurs, représentants métier, clients, utilisateurs…).

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

- **Experience map (customer journey)**

Une Experience Map permet de cartographier l'expérience d'un utilisateur, d'identifier les obstacles et les problèmes que le client rencontre avec un produit ou un service ainsi que les opportunités d'amélioration et d'innovation. En d'autres termes, l'objectif est de construire une connaissance organisationnelle des comportements et besoins des utilisateurs à travers les différents canaux. Et ainsi promouvoir l'évolution organisationnelle vers une approche centrée utilisateur.

C'est une vue complète de l'expérience, avec les multiples points de contact et interactions. Elle permet d'identifier les actions, les pensées et les ressentis des utilisateurs.

C'est une méthode complémentaire à celle des personas, à qui elle apporte une dimension dynamique. L'expérience map s'inscrit dans la customer journey et peut être réalisée à travers :

Réaliser une phase d'étude :

- Faire des recherches sur ce qui motive les utilisateurs, ce que font les utilisateurs, ce que pensent les utilisateurs, ce que ressentent les utilisateurs

Définir le parcours :

- Le format sera différent selon l'objectif fixé : par exemple, « améliorer un service » ou « créer un nouveau service ». C'est pourquoi, il existe 3 formats différents :
    - Le format **« Current state »** et **« Day in life »** permettent d'identifier les difficultés rencontrées par les utilisateurs afin d'améliorer l'expérience. (Observation)
    - Le format « Future state » : on imagine comment l'utilisateur pourrait utiliser un service ou un produit inexistant mais toujours avec des vraies données. (Conception)
    - Le format « Service blueprint » que ce soit l'état actuel ou futur, il prend en considération l'expérience client et l'expérience métier. (Conception)

Raconter l'histoire

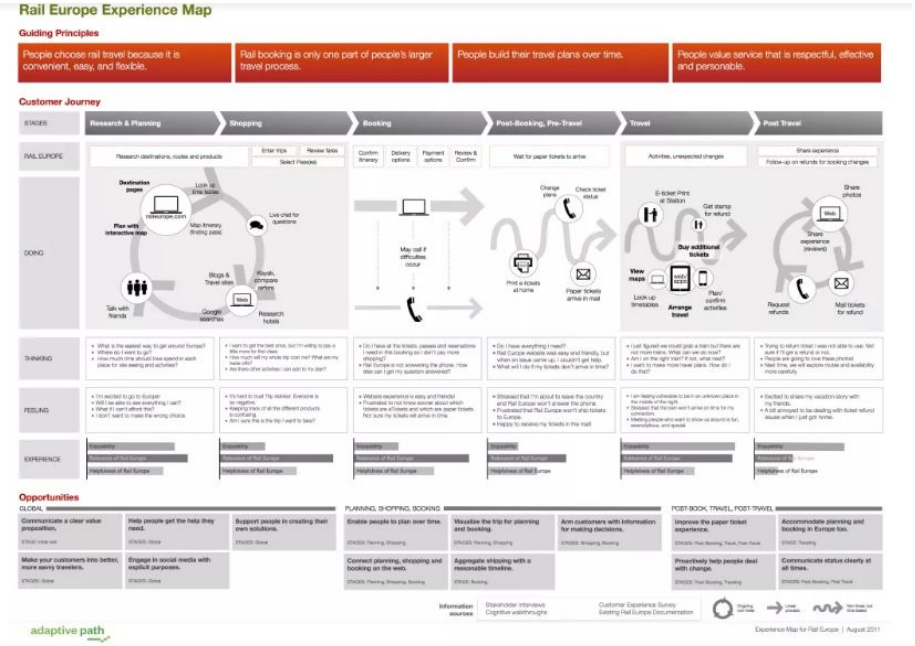
La représentation visuelle est très importante, elle contribue à véhiculer le message. Vous devez savoir quel message vous souhaitez faire passer pour une lecture rapide? En 5mn ou 15mn ? Les personnes qui vont lire ce document auront plus ou moins de temps à accorder à sa lecture et pourtant ce sont souvent les personnes qui sont le plus à même de prendre des décisions pour impacter l'expérience utilisateur.

(cc) BY-NC-ND

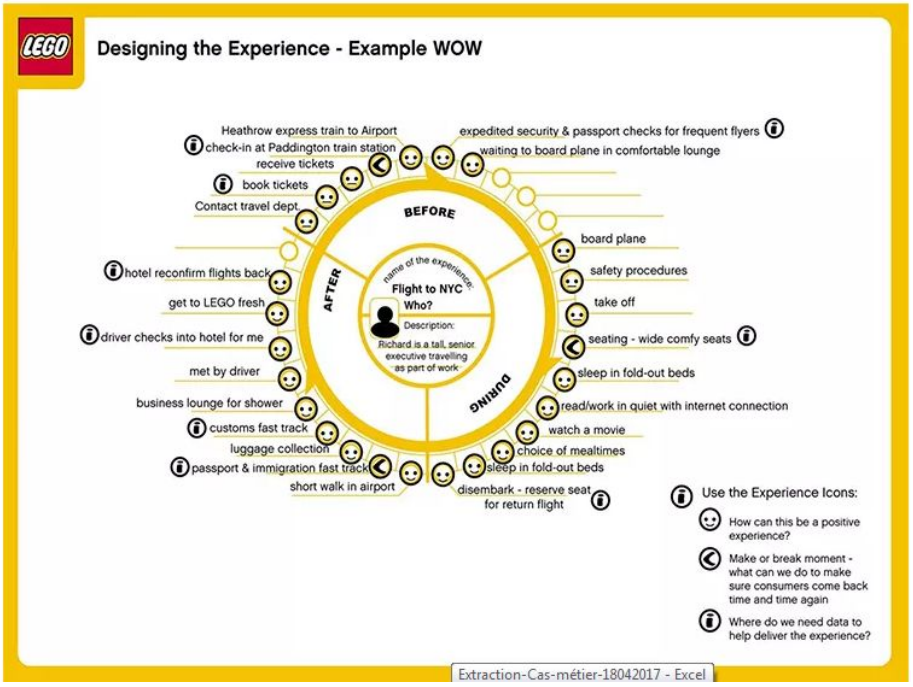
Ségolène Porot - licence CC 4.0

Il existe deux formats de représentation : linéaire ou en roue.

Le format linéaire permet de mettre en avant les points de contact d'un utilisateur avec un service ou un produit. Ce format est très utile pour une expérience cross canal afin de montrer les différents points de contact d'un utilisateur (smartphone, tablette, desktop, boutique, les amis, application, site web, chat, moteur de recherche...)



Le format en roue va mettre en avant les actions réalisées par l'utilisateur.



(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Utiliser votre experience map

Une Experience Map c'est comme un persona ce n'est pas définitif ça peut être ajusté en permanence

Quelques conseils complémentaires pour construire votre experience map est la suivante :

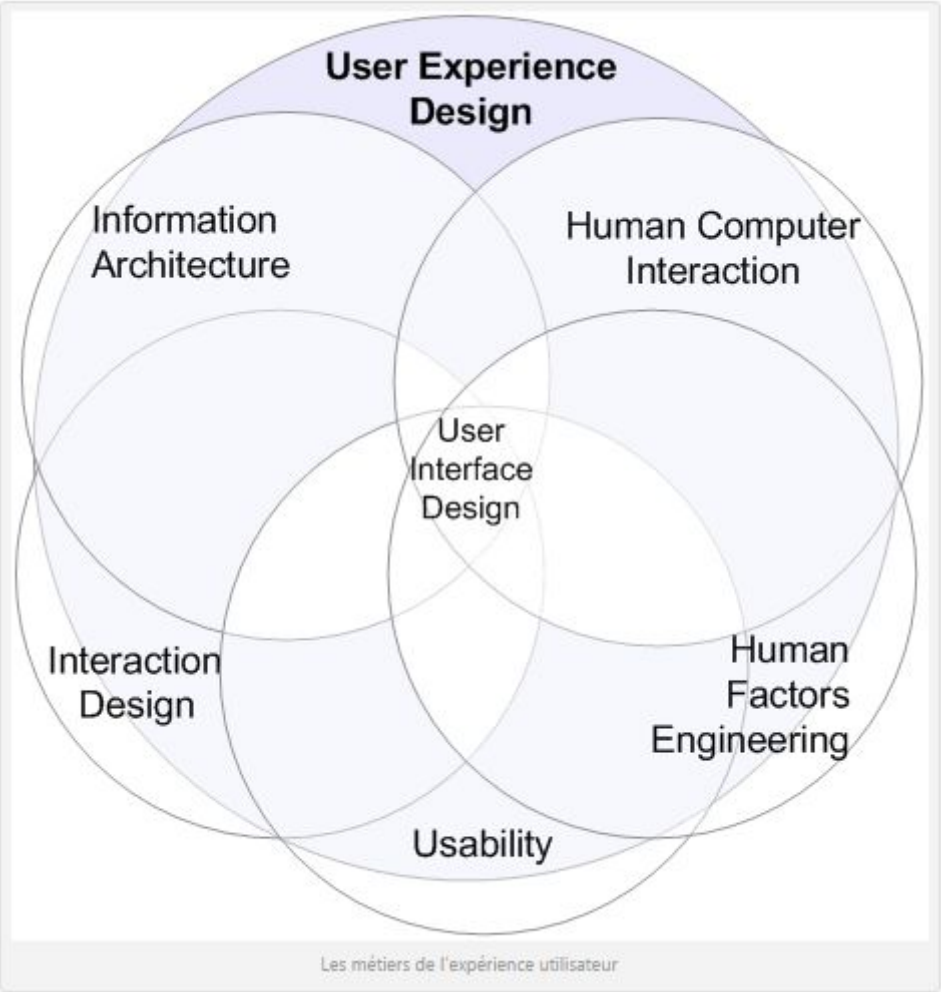
a. Faîtes une expérience map pour une persona pour une durée limitée dans le temps.
b. Votre Experience Map est un déclencheur pas une conclusion
c. Une véritable proposition de valeur qui permet de créer un cadre partagé de référence autour de l'expérience utilisateur

- **Focus sur l'expérience UX**

L'Expérience Utilisateur" se concrétise par les réponses et aux perceptions d'une personne qui résultent de l'usage ou de l'anticipation de l'usage d'un produit, d'un service ou d'un système". "l'anticipation de l'usage" qui montre que l'expérience utilisateur est quelque chose de profondément lié à la psychologie de l'être humain. Les émotions sont d'ailleurs un élément central de l'expérience utilisateur

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0



Les métiers de l'expérience utilisateur

En tant que PO, vous devez cerner ces enjeux et intégrer votre designer dès la phase de framing pour intégrer l'expérience utilisateur en amont de votre produit. Je vous conseille lors de vos ateliers et lorsque vous communiquez vos besoins de toujours les formaliser via une approche UX et cohérente avec la stratégie de votre designer

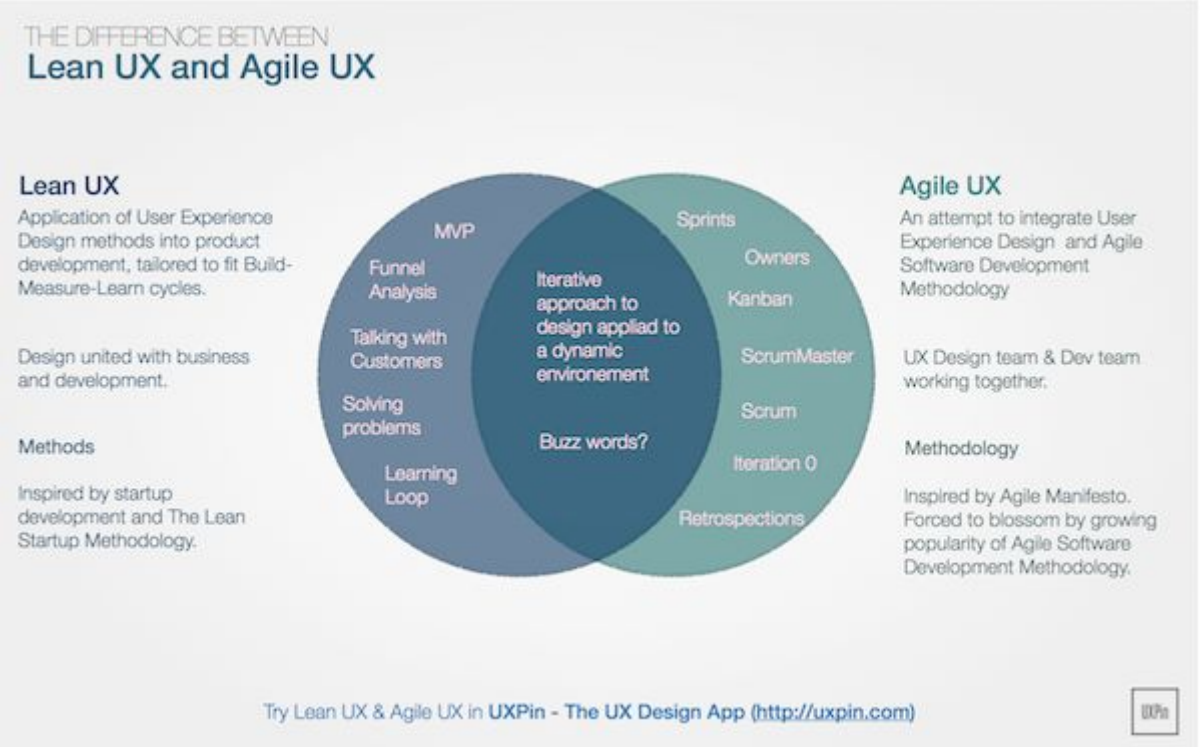
- **UX testing**

Je vous propose de tester votre expérience utilisateur afin de réduire les risques d'échec, réduire les coût à travers du desgin inutile et enfin améliorer votre produit avec une approche centrée utilisateur.

N'hésitez pas à comparer vos tests pour évaluer la meilleure alternative, vérifier à travers le « blur test » que vos utilisateurs captent les images et texte important, mesurer le temps réalisateur par vos utilisateur pour réaliser une tâche (est-ce top important ?), envoyer des sondages pour identifier les points de douleur, les opportunités, les gains

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

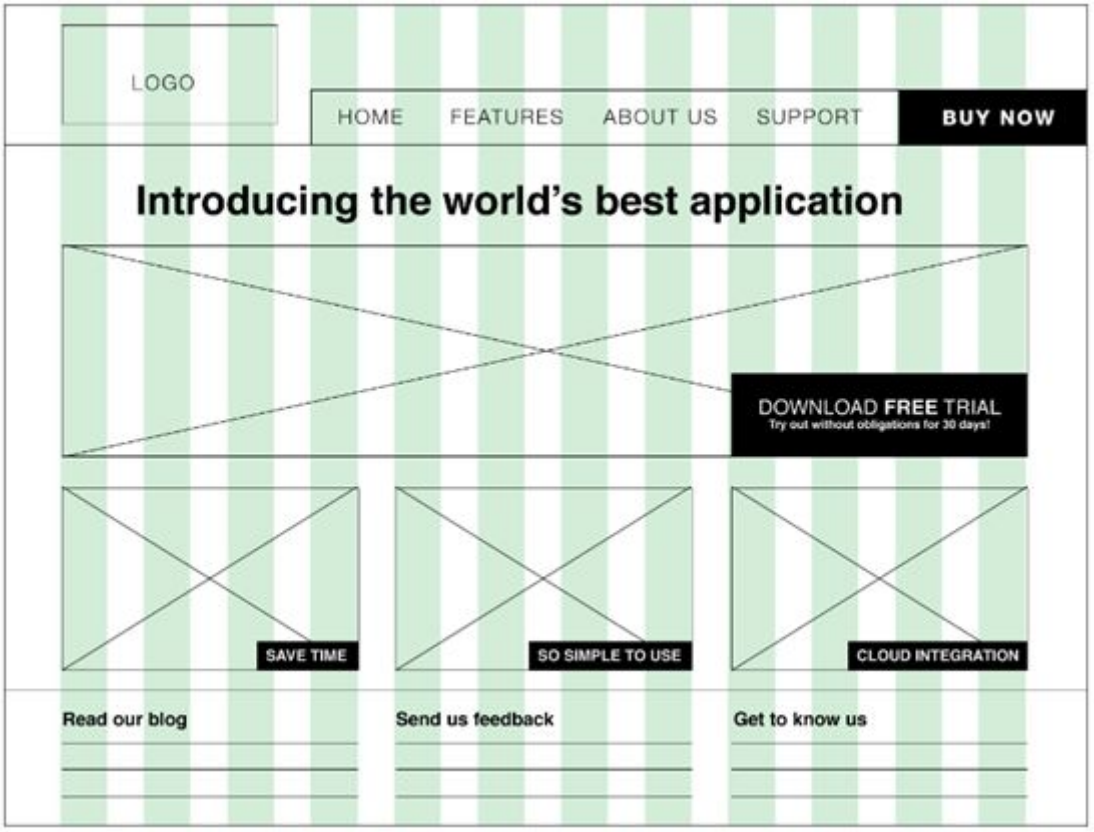


- **Wireframing**

En tant que PO vous participer au wireframing, c'est-à-dire, la maquette fonctionnelle est un schéma utilisé lors de la conception d'une interface pour définir les zones et composants qu'elle doit contenir. À partir d'un wireframe peut être réalisée l'interface proprement dite par un graphiste. La démarche de recourir à des wireframes s'inscrit dans une recherche d'ergonomie. Elle est surtout utilisée dans le cadre du développement logiciel et des sites et applications Web. Le wireframe consiste concrètement en un croquis, un collage papier ou un schéma numérique.

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0



(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

- **Analytics**

Vous pouvez analyser votre produit grâce à l'analytics. Concrètement, il s'agit d'utiliser des outils d'analytics qui vont vous permettre de suivre vos métriques d'usage. De nombreuses solutions s'offrent à vous : Google Analytics, Mixpanel, Kissmetrics, Capptain, Flurry ou encore Kontagent. La pertinence de l'outil dépendra grandement des caractéristiques de votre produit (BtoB, BtoC, monétisation, mobile ou pas, etc.), prenez le temps de benchmarker.

Vous pouvez aussi utiliser le K-factor :

Le K-Factor est un KPI permettant de mesurer la viralité d'un produit. Sa valeur représente le nombre d'utilisateurs moyen dont l'acquisition s'est faite par l'invitation d'un autre utilisateur. Concrètement, si K est égal à X, cela signifie que chaque utilisateur apporte en moyenne X utilisateurs supplémentaires au produit. Pour calculer le K-Factor, vous devez multiplier 3 variables entre elles :

- Le pourcentage d'utilisateurs qui invitent d'autres personnes,
- Le nombre moyen de personnes invitées à chaque fois,
- Le pourcentage de personnes qui acceptent l'invitation (et deviennent donc utilisateurs).

Le rêve de tout Product Manager est, bien entendu, d'avoir un K supé- rieur à 1 pour que l'acquisition organique de nouveaux utilisateurs ait la forme d'une belle courbe exponentielle !

Le framework AARRR offre une bonne base de départ pour cadrer vos objectifs, autour de cinq enjeux mesurés chacun par une série de KPIs :

plus des indispensables objectifs financiers (de type chiffre d'affaires, taux de marge et ROI), nous vous conseillons fortement de déterminer une ambition sur l'ensemble des indicateurs clés du parcours utilisateur. Pour ce faire, vous pouvez, par exemple, utiliser le framework AARRR de Pirate Metrics. Ce framework propose cinq grandes familles de KPIs calquées sur les étapes de la vie d'un utilisateur :

- Acquisition : performance des canaux par lesquels vos utilisateurs accèdent à votre produit (Inbound ou Outbound Marketing, etc.).
- Activation : capacité à activer vos utilisateurs acquis (un utilisateur sera considéré comme activé quand il aura eu un premier usage significatif de votre produit). On parle aussi d'engagement ou d' « onboarding ». Il peut aussi s'agir de la réactivation par des campagnes marketing (push, email, sms, etc.) d'utilisateurs « dormants » c'est-à-dire ayant utilisé votre produit un temps et fini par l'abandonner.
- Rétention : capacité à retenir vos utilisateurs activés dans le temps.
- Revenue : monétisation de votre produit (par exemple, conversion d'utilisateurs gratuits en utilisateurs premium payants).
- Referral : propension de vos utilisateurs à en recruter de nouveaux par recommandation (dynamique virale, partage sur les réseaux sociaux, bouche à oreille de toute sorte, organisée par vos soins ou pas).

Nous vous conseillons également de mesurer régulièrement votre satisfaction client (par exemple, via le Net Promoter Score). Ceci vous amènera peut-être à modifier voire supprimer certaines fonctionnalités rejetées par une majorité d'utilisateurs. L'analytics vous permettre aussi de valider vos nouvelles fonctionnalités : C'est pourquoi, nous vous conseillons de vérifier qualitativement (via

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

des entretiens) puis quantitativement (via l'analytics) la pertinence de vos nouvelles fonctionnalités avant de chercher à en ajouter de nouvelles.

Pour être plus créatif pendant vos ateliers d'expression de besoin ou tout simplement pour revisiter le format de vos réunions vous pouvez mettre en œuvre ces 4 outils : le design thinking, le ROTI, le storming event, le value stream mapping

2) Dynamiser vos ateliers

- **Design thinking:**

Le design thinking est un outil qui peut être utilisé lors de vos ateliers d'expression de besoin ou des ateliers uniquement consacré à la recherche de nouvelles idées. Par conséquent, les contextes d'utilisation du design thinking sont très variés. A mon sens, c'est un outil pertinent pour réaliser des entretiens et « pulser » la créativité pour développer de nouvelles fonctionnalités.

Le processus de réalisation :

- Définir : Identifier le problème à régler, prioriser le projet et déterminer ce qui en assurera le succès. Les outils sont :
  - Interviews
  - Stakeholders mapping :
    i. Définir les stakeholders
    ii. Analyser les stakeholders par impacts et influence
    iii. Réaliser un plan pour gérer la communication, le reporting pour chaque partie prenante
    iv. Engager les parties-prenante s
  - Context mapping (c'est un modèle de données en UML)
  - Generative session (c'est le même concept que les ateliers créatifs
  - Observation
  - Le parcours utilisateur
  - Clustering insight (analyse de base de données à travers 3 aspects : récent, fréquence, valeur)
  - Customer Value Canva : définit par le modèle ci-dessous

Produit-Service	Gain (courbe)	Gain ☺	Customer job : -« on aide le client à quoi, pourquoi ? »
	Pain relivers	Douleurs ☹	

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Le processus de design thinking :

- Rechercher : Revoir l'historique des problèmes rencontrés, collecter des exemples d'échecs, identifier les supporters, investisseurs et critiques du projet, parler au client final.
- Insight (néologisme) : Identifier les besoins et motivations des clients finaux, générer autant d'idées que possible pour répondre à ces besoins sans les juger, brainstormer.
    - Atelier créatif
    - Brainstorming
    - Narratif
    - Storyboard : c'est un template avec les différentes scènes de l'histoire du parcours utilisateurs
    - Le story map
    - Empathie map : se mettre à la place du consommateur et comprendre ses besoins.
    - Idéation : générer plusieurs idées pour solutionner le problème.

- Prototyper : Combiner, croiser et affiner les idées, créer des brouillons/maquettes/prototypes, recevoir un retour de clients potentiels ou non.
    - Choisir un (business) model pour le prototype
    - Construire les points de contact
    - Mettre en place les services

- Sélectionner : Revoir les objectifs, faire perdre la propriété de l'idée sélectionnée à celui qui l'a eu, choisir l'idée la plus surprenante, nouvelle et économique…

- Implémenter (mettre en place) : Rédiger le plan d'action et donner les responsabilités, déterminer les ressources nécessaires, délivrer au client.
    - Former les parties prenantes
    - Tests fonctionnels et techniques

- Apprendre : Recevoir un feed-back du client final, déterminer si la solution validée répond à l'objectif de départ, identifier les sources d'amélioration.

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

- **Feature team**

Afin de limiter les dépendances entre équipes cross-fonctionnelles, certaines organisations les dédient au développement et support d'un périmètre fonctionnel bien défini. Ce sont des équipes orientées « Feature » ou feature team

Il est important de noter que les Feature teams sont autant que possible indépendantes sur le plan organisationnel mais également au niveau de l'architecture du produit : une feature team doit être capable de livrer une nouvelle version de son périmètre sans impacter les autres équipes, ce qui lui permet d'avoir un TTM optimal.

Il s'agit d'une équipe cross-fonctionnelle constituée :

- Facilitateur (coach, scrum master)
- Développeurs (front & back)
- Techlead
- Testeurs
- Ops
- Designer
- Product owner

Cette équipe est en relation avec des experts et des supports internes tels que : architecte, recette globale, marketing, analytics

- **Design sprint**

Le Design Sprint est une méthode qui permet d'accélérer et simplifier le processus de design d'un produit digital.

L'idée est la suivante, il s'agit de réaliser en 5 jours un travail qui peut parfois s'étaler sur des mois. Le Design Sprint permet en 5 jours de répondre par le design et le prototypage à des problématiques complexes d'user expérience (UX) ou d'interface (UI).

A l'issue du Sprint, vous avez un livrable de grande valeur, c'est-à-dire, un prototype cliquable de votre future application.

Je vous propose ci-dessous un agenda type :

<u>1er jour : comprendre</u>

Objectifs : transmettre la connaissance métier à toute l'équipe et définir la problématique à solutionner

Les outils lors des ateliers : présentations par les experts métier, sessions de questions / réponses, rédaction de la user journey typique du persona avec lequel nous allons travailler

<u>2ème jour : diverger</u>

Objectif : ouvrir le plus de portes possibles, designer le plus grand nombre de solutions possibles sans se fixer de limites

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Les outils lors des ateliers : 5 why's, ideas wall, needs / wants / desires, crazy 8, one big idea

3ème jour : décider

Objectif : se mettre d'accord sur la / les solutions les plus pertinentes afin de répondre à la problématique posée le premier jour

Les outils lors des ateliers : zen voting, storyboard, premier tweet

4ème jour : prototyper

Objectif : construire le prototype que nous présenterons aux utilisateurs

Les outils lors des ateliers : prototypage (réalisé essentiellement par les designers, avec l'input régulier des autres membres de l'équipe)

Le format du prototype est à définir par l'équipe, selon la complexité, la plateforme cible et le temps imparti, il peut prendre la forme d'un diaporama animé, d'une vidéo, ou d'un prototype interactif (inVision, Flinto…)

5ème jour : tester

Objectifs : recueillir le feedback des utilisateurs, valider / invalider notre solution, puis prendre une décision sur la suite à y donner (execute, iterate or kill!)

Les outils des ateliers : user tests en personne, débriefing des tests

- **La méthode ROTI**

Cette méthode Return on Time Invested s'inscrit toujours dans les valeurs de l'esprit Lean et Agile : feedback, courage, transparence, amélioration continue.

A la fin d'une réunion, par exemple de la rétrospective ou à la suite d'un nouveau format d'atelier vous pouvez réaliser un ROTI.

Le processus est simple, chacun des participants note la réunion :

- **1 doigt : Inutile**, je n'ai rien gagné, je n'ai rien appris. J'ai vraiment perdu du temps
- **2 doigts : Utile,** mais ça ne valait pas à 100%, le temps que j'y ai passé. J'ai donc perdu mon temps
- **3 doigts : Juste moyenne,** je n'ai pas perdu mon temps mais juste sans plus
- **4 doigts : Bonne,** voilà une réunion au-dessus de la moyenne. J'ai gagné plus que le temps que j'y ai passé
- **5 doigts : Excellente,** voilà une super réunion dont je vais bénéficier. Ça valait bien plus que le temps que j'y ai passé.

Bien entendu, les participants cachent leurs résultats avant de montrer le nombre de doigt concerné… pas de triche et on ne copie pas le voisin.

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

- **Event storming**

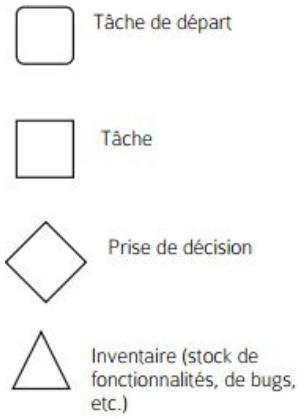
C'est un atelier réunissant l'ensemble des acteurs clés (développeurs, experts métiers) d'un projet pour travailler sur un grand espace de modélisation en utilisant des autocollants pour représenter les Domain Events, les événements du domaine

- **Value stream mapping**

Le value stream mapping ou VSM est un outil regroupant toutes les actions (à valeur ajoutée et à non valeur ajoutée) qui amènent un produit d'un état initial à un état final. Cet outil s'attache à travailler sur un ensemble et non une partie. Le but de cette cartographie est d'obtenir une vision simple et claire d'un processus. Le but ultime du VSM est d'avoir une représentation du temps de défilement (lead time) du produit choisi. L'analyse amènera ensuite des améliorations qui porteront sur la globalité du processus et non sur une étape.

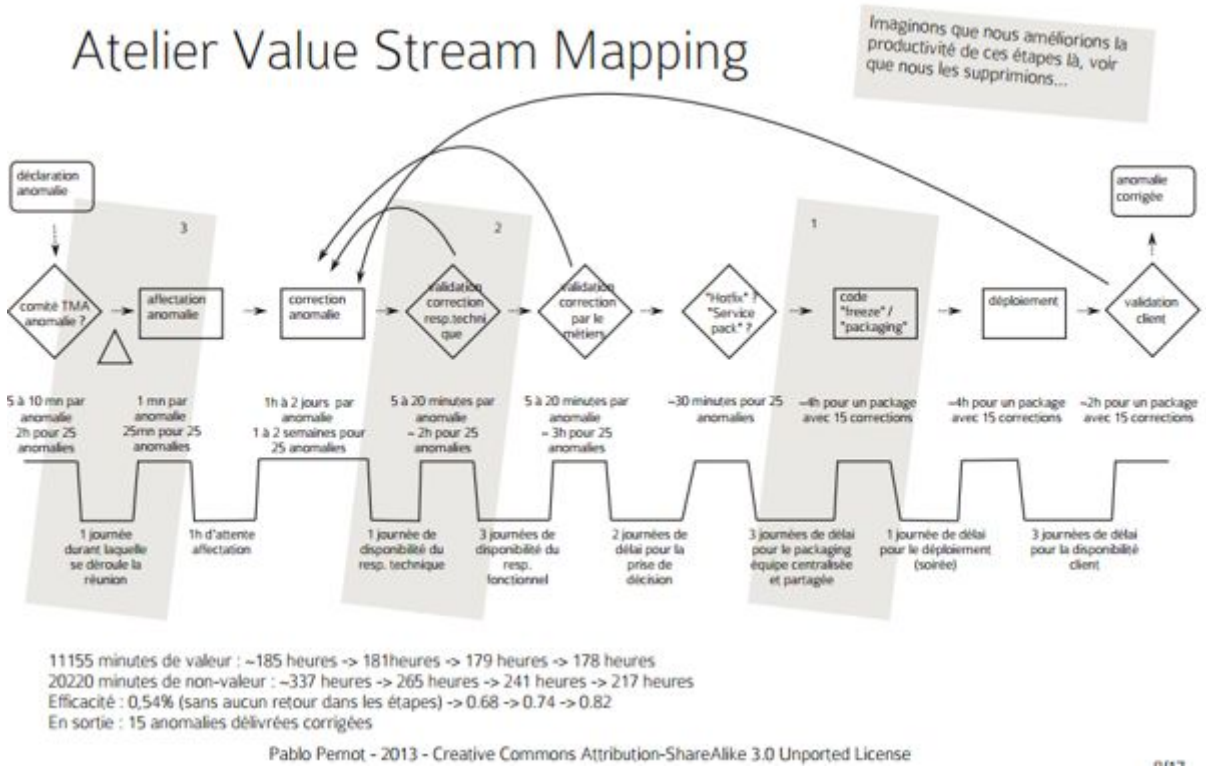
Voici le processus pour réaliser un value stream mapping :

- Choisir un processus
- Définir la première et la dernière étape
- Définir tous les étapes nécessaires
- Collecter toutes les informations concernant ces étapes (durée, nb, etc.)
- Réaliser la cartographie à partir des symboles ci-après



(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0



Value stream mapping (cartographie du flux de valeur)

Pour déterminer si vous supprimer ou non une tâche ou encore l'améliorer vous pouvez utiliser les méthodes :

- RCA : Root Cause Analysis
- Les 5 pourquoi
- Liste Phoenix :
  - Pourquoi est-ce nécessaire de résoudre le problème ?
  - Quels seront les bénéfices lorsque le problème sera résolu ?
  - Quel est l'inconnu ? Qu'est-ce qui n'est pas encore compris ?
  - Quelles sont les informations que vous avez ? Qu'est-ce qui n'est pas le problème ?
  - Est-ce que ces informations sont suffisantes ? Ou sont-elles insuffisantes ? redondantes ou contradictoires ? Devriez-vous dessiner un diagramme du problème ? une figure ?
  - Quelles sont les limites du problème ? Pouvez-vous séparer les différentes parties du problème ? Pouvez-vous les formalizer
  - Quelles sont les relations entre les différentes parties du problème ? Quelles sont les constants du problème ? Avez-vous déjà rencontré ce problème ?
  - Avez-vous déjà rencontré ce problème d'une forme plus ou moins différente ? Connaissez-vous un problème associé à celui-ci ?
  - Essayer de penser a un problème qui vous semble familier le même ou un similaire ? supposer que vous trouvez la solution du problème qui a déjà été trouvé. Pouvez-vous

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

l'utiliser ? Pouvez-vous utilisez cette méthode ? Pouvez-vous re-statuer votre problème ?
- Comment pouvez-vous re-statuer votre problème ? D'une manière plus générale, plus spécifique ? Est-ce que les règles peuvent-être changées ?
- Quelles sont les meilleurs, les pires, les plus probables cas que vous pouvez imaginer ?

- **Extreme quotation pour vos sprint planning**

Lors de vos sprint planning, vous pouvez opter pour l'extreme quotation pour évaluer vos US.

L'extrême quotation est un format d'atelier qui permet de chiffrer en un temps record une liste d'items (stories, EPIC).

Le processus pour définir l'extreme quotation :

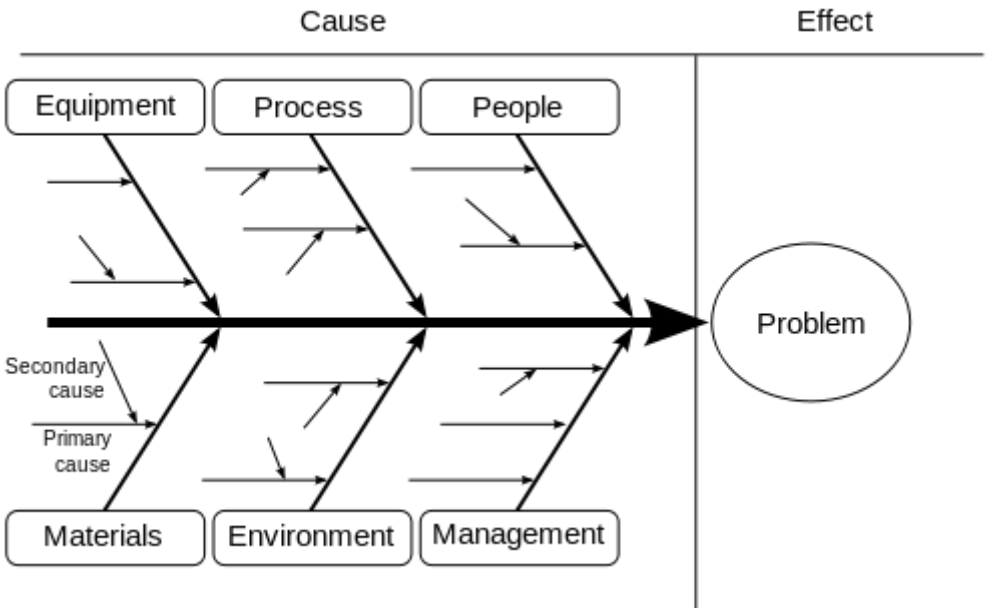
- Au préalable, l'animateur doit noter chaque story (ou EPIC) sur un post-it. Réunissez maximum 5 personnes en essayant de réunir plusieurs compétences clés pour votre produit (par exemple : 1 PO, 1 testeur, 2 devs, 1 UX) afin d'évaluer la complexité de bout en bout.
- Introduction (10min) Pour démarrer, il est important d'expliquer le but de cet atelier et son déroulement. Ensuite il faut choisir l'unité d'estimation : points de complexité ? tailles de t-shirt ? ... Partez sur l'unité avec laquelle l'équipe est la plus à l'aise. Afin d'être sûr que tout le monde soit bien aligné sur la signification de chaque unité, précisez une valeur de référence (fonctionnalité déjà réalisée par exemple) au moins pour chaque extrémité (XS et XL). Les unités sont collées sur un mur (pour les tailles de t-shirt, 5 post-it en colonne : XS, S, M, L et XL). Le Product Owner présente alors brièvement chaque story (lecture du titre, description rapide de l'objectif).
- 1ère partie (5 à 10 min max) : Sans parler, chaque personne prend un post-it et le colle sous l'unité qu'il estime la plus juste. Une fois que toutes les stories sont positionnées sous une unité, l'animateur note la valeur sur chaque post-it
- 2ème partie : (5 à 10 min max) Toujours dans le silence, chacun peut déplacer un post-it sous une autre unité s'il estime que la valeur initiale n'était pas correcte. A la fin du round 2 l'animateur note la nouvelle valeur sur chaque post-it. Toutes les stories qui n'ont pas changé de valeur sont mises de côté.
- 3ème partie (5 à 10 min max) L'heure est maintenant à l'échange. Les participants doivent s'entendre sur une valeur finale à donner aux stories qui ont changé de colonne. Et voilà en 30min vous avez une idée de la complexité des items de votre backlog.

- **De nouveaux formats de retrospectives**

Outil ISHIKAWA :

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0



Il s'agit de décrire pour chaque « arrête » les éléments adaptés à votre projet selon :

- Personne : relation, communication personnelle, bien-être, etc.
- Processus : nos pratiques et façon de travailler, etc.
- Equipement : équipement, matériel, etc.
- Management : relation, communication boite, vision, valeurs, objectifs, cohérence, etc.
- Environnement : (locaux, lieux, etc.

<u>Outil GYSHIDO :</u>

Voici les questions clefs d'une rétrospective Gyshido telles que je les imagine :

- Quelles activités inutiles avons nous malgré tout réalisées ?
- Qu'est ce qui a été embétant à réaliser ?
- Qu'est ce qui s'est révélé pas clair, pas vrai ?
- Quelles réunions ont été inutiles ?
- Qui avez vous aidé ?
- Quand vous êtes vous mal comporté ?

Vous pouvez aussi modéliser les éléments suivants dans un carré :

- Arreter de … par exemple « ouvrir trop de tâche »
- Continuer à … par exemple « apprendre ensemble »
- Plus de … par exemple « personne dans la review »
- Démarrer à … par exemple reformuler « la notion de fini pour mieux la respecter »
- Moins de … par exemple « de longueur dans les reviews)

(cc) BY-NC-ND

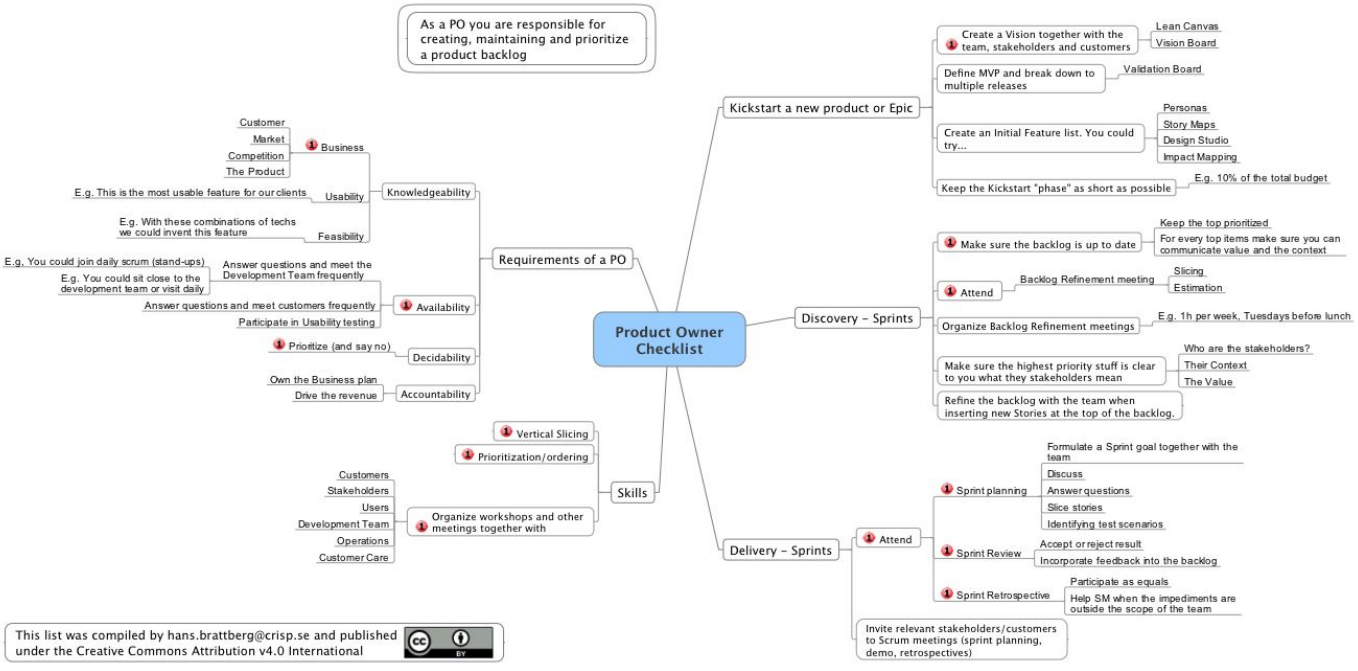
Ségolène Porot - licence CC 4.0

Outil « visual management »

Je vous recommande le « mind mapping », voici à cela ressemble pour le « mind mapping du product owner » :

Le processus pour réaliser votre « mind map » :

- Brainstorming : sur vos idées de thèmes et idées reliées
- Brain Mapping : brainstorming organisé. Les idées sont catégorisées et hiérarchisées. Vous avez mis de l'ordre dans vos réflexions.
- Decision Mapping : vous retenez les idées que vous voulez concrétiser réellement.
- Goal Mapping : Goal pour but/objectif. Vous transformer les idées en objectifs opérationnels, clairs et évaluables.
- Plan Mapping : vous élaborer et planifier le plan d'action
- Progress Mapping : vous suivez et évaluez les actions/projet, les avancements et les difficultés.



(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

PARTIE 3

Savoir, savoir-être, intention et attitude du product owner

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

A) Les compétences générales et détaillées du product owner

Au final, j'ai beaucoup lu d'articles de blog, vidéo. Tout me porte à croire qu'un profil type de product owner n'existe pas en lui-même cependant certaines compétences, dîtes « hard » pour les anglosaxon peuvent être listées :

1) Le référentiel de compétences du product owner

Evaluez vos compétences en tant que product owner !

Je vous propose ci-dessous une liste des compétences pour devenir un product owner expert dans son domaine.

Domaine de compétences	Compétences détaillées	Acquis/ Non acquis
Le marketing stratégique	Analyse de la concurrence Choix de marchés ou de clientèles cibles Analyse fonctionnelle de produits et des services Choix de canaux de distribution Segmentation de marchés Positionnement produits / marchés Ciblage Analyse de couples produits / marchés Élaboration et analyse du Mix - Marketing Participation à la réalisation de plans d'actions marketing Participation à l'élaboration de cahiers des charges fonctionnels	
Le marketing opérationnel	Contribuer à la conception et la mise en œuvre du plan d'actions commerciales dans le respect des objectifs du plan marketing Développer et mettre en œuvre les outils statistiques et de reporting du marketing opérationnel Collaborer à la réalisation des actions de promotion de l'offre produits/services du groupe Mettre en œuvre le développement et le maintien fonctionnel et opérationnel des bases de données	
La stratégie	Stratégie de croissance Marché et opportunités business Optimisation du portefeuille d'affaires Positionnement des prix et des côuts Proposition de développement Planification Stratégie d'entrée sur le marché	
Marketing digital	Analyse de la data et le contenu Search engine marketing Le social media	

BY-NC-ND

Ségolène Porot - licence CC 4.0

	Le marketing mobile Le marketing viral et le marketing visuel	
La gestion de projet informatique	Pilotage, suivi et coordination du projet Définition du périmètre et des objectifs du projet Mettre en place les structures du projet et ses règles de fonctionnement (méthodes, outils de pilotage, indicateurs...). Définir avec les équipes projet, les objectifs et les délais de réalisation des livrables (applications, modules, développement spécifique...). Effectuer les choix et l'affectation des ressources, en fonction des contraintes techniques. Piloter et mesurer l'état d'avancement (création des tableaux de bord, choix des indicateurs, planification des comités de pilotage...). Superviser et coordonner le travail de l'ensemble des acteurs internes et/ou externes Valider les documents (roadmap, product roadmap)	
Les systèmes d'information **Le management des technologies de l'information**	Test et recette fonctionnelle Rédaction des spécifications Progiciels de gestion intégrés Développement Web E-Management Aide à la décision Suivi des données (requête sur des logiciels de base de données) Modélisation et SGBD	

En résumé, le product owner possède les compétences suivantes :

- Bonne connaissance du domaine métier,
- Maîtrise des techniques de définition de produit,
- Capacité à avoir une position respectée et à prendre des décisions,
- Capacité à détailler une fonctionnalité au bon moment,
- Esprit ouvert au changement,
- Bon négociateur

2) Comment acquérir ces compétences ?

Tout d'abord, il est important de noter comme l'illustre le modèle ci-dessous que les compétences s'acquièrent à 70% grâce à l'expérience puis au Feed-back et seulement 10% par la formation.

BY-NC-ND

Ségolène Porot - licence CC 4.0

Modèle 70/20/10, Morgan McCall, Robert W. Eichinger et Michael M. Lombardo, du Center for Creative Leadership, de l'Université de Princeton.

Autrement dit, n'hésitez à aller vers d'autres personnes, collègue de confiance qui pourront vous apporter du feedback, un peu comme une rétrospective afin de distinguer ce qui fonctionne, ce qui ne va pas et les points d'améliorations.

Par ailleurs vous pouvez vous former de diverses manières : formelle, informelle, sous forme de supervision, de parrainage, de partage de bonnes

Surtout restez MOTIVER ! En effet, pour développer vos compétences cela implique de cultiver votre implication qui contribuera à votre épanouissement professionnel et votre dynamisme.

Ainsi, je vous propose une méthode pour développer vos compétences :

a. Définir à quelles questions vous aimeriez avoir une réponse (3 à 5 questions) sur un domaine de compétence et domaine de compétence détaillée
b. Trouvez, dans votre entourage, des personnes qui possèdent déjà cette compétence que vous essayez d'acquérir.
c. Demandez à ces personnes de vous connecter à d'autres spécialistes
d. Appliquer les conseils.
e. Mesurer votre retour sur investissement pour vérifier l'acquisition ou non de cette nouvelle compétence

3) Les conseils des experts pour les nouveaux

- Que se passe-t-il quand il y a un problème ?

Surtout ne paniquez pas, tout problème a une solution et même si ils sont nombreux et qu'évidement ils surviennent tous en même temps. Soyez organiser, utiliser des outils vus précédemment ou encore une approche plus classique de problem-solving pour trouver des solutions. Lorsque vous n'avez pas de retour suite à un problème, ou un échange avec les

(cc) BY-NC-ND

développeurs, continuez vos tâches : le rôle de PO est suffisamment diversifié pour cela.

- Quels sont les échanges avec l'équipe de développement ?

En règle générale lorsqu'un développeur refuse une fonctionnalité cela fait suite à des contraintes de temps ou de délai. C'est à vous, en tant que product owner, de définir une alternative à votre fonctionnalité pour qu'elle puisse être réaliser par les développeurs. En effet, le problème est rarement purement technique : tout est réalisable.

- Une organisation flexible et structurée ?

Je vous conseille de Veiller à ce que la taille du groupe soit comprise entre 5 à 8 personnes.

Si cela est possible, il est fortement indiqué de mêler des profils psychologiques différents (Créatif, analytique…). Pour cela, vous pouvez vous appuyer sur des outils tels que le MBTI, nous le verrons lors de notre dernière partie de ce guide.

Par ailleurs, éviter (si possible) d'inviter des participants ayant des liens hiérarchiques. En effet, certaines personnes n'oseront pas s'exprimer librement devant leur chef d

Enfin, ne conviez à tout prix des clients / utilisateurs. Ces derniers sont généralement bons pour décrire leurs problèmes mais pas nécessairement pour trouver les meilleures solutions

- Quelles sont les caractéristiques de l'équipe ?

Le contexte et les objectifs doivent être clairs pour tous les membres de l'équipe. Il peut être utile de les afficher sur un des murs de la salle de brainstorm pour s'y référer si besoin

De plus, assuez-vous lorsque vous commencer vos ateliers, réunions que le déroulement de la séance et les règles à respecter sont comprises par les invités.

Enfin, je vous propose de créer une ambiance propice à la créativité (sorterz de vos locaux ; décorer la salle), l'agile c'est aussi ça !

Pour finir, vous devez vous poser les bonnes questions et ce guide peut vous aider :

Quels sont mes rôles durant chaque itération, Que se passe-t-il si je n'ai pas de réponse immédiate pour la MOE ? Dois-je recetter à chaque itération ? Comment pondérer les user-stories ? Suis-je forcément le seul P.O ? …

B) Le savoir-être du product owner

Et l'humain dans tout ça… et oui un product owner doit avoir des qualités humaines pour gérer autant d'interlocuteurs différents : des interlocuteurs techniques, des utilisateurs

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

métiers, des managers, des fonctionnels, des développeurs, des designers.

1) Management, organisation des RH

Un product owner est dans une situation où il doit faire preuve d'habileté en termes de relation humaine et faire face parfois à des enjeux similaires aux enjeux du management.

Ainsi, un product owner doit être capable, tout comme un manager, de faire adhérer son équipe (développeur, product owner, scrum master) a un objectif commun. Il faut mettre en place cet alignement commun pour atteindre la vision du produit.

Un product owner doit être dans une optique d'apprentissage du management selon le mode agile, c'est-à-dire, savoir laisser faire grâce à l'auto-organisation tout en gérant les incertitudes.

L'apprentissage du management est déjà une thématique bien connue. il s'agit des différents changement de posture entre chef, manager, boss, et servant leadership, facilitateur. En tant que product owner, je vous encourage à opter pour la posture facilitateur et leadership. En effet, La facilitation c'est aider, supporter, conseiller, etc.

2) Cultiver l'intention bienveillante

Tout d'abord, je vous propose une brève définition du sens que j'entends par intention dans ce paragraphe. L'intention, selon la définition du Larousse, c'est la disposition d'esprit par laquelle on se propose délibérément un but ; ce but lui-même.

Je vous conseille de toujours opter pour une intention doit être bienveillante, porté sur les faits et rien que les faits quel que soit votre interlocuteur : le développeur, vos utilisateurs, designers. L'intention se ressent lors de vos échanges et surtout dans votre attitude que nous verrons dans un second temps. Cette bienveillance est d'autant plus importante qu'elle permettra à votre produit d'atteindre le niveau d'exigence que vous souhaitez. En effet, vous allez garantir la préservation et l'amélioration du bien-être des personnes dans les interactions quotidiennes. En effet, cela est bien connu, les individus sont plus efficaces et le résultat optimal lorsque les individus se sentent bien. Pour cela rester toujours dans vos propos, par exemple lorsque vous présentez vos user stories : honnête, indulgent envers vos développeurs et designer et bien sûr bienveillant selon les retours qu'ils vous feront. Par ailleurs, tenez-vous engagement, soyer loyal, un product owner sur lequel on peut compter pour avoir une bonne réponse à leurs questions, efficace, un travail soigné. Vous êtes responsable de votre product backlog, de vos US, bref de toutes les tâches d'un product owner, c'est aussi une forme d'intention que de prendre en compte ces responsabilités à différents niveaux.

Enfin, l'intention se concrétise également par une appétence pour le produit, avec le sens,

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

avec la valeur, l'intention, l'attitude. Le product owner doit garder en tête qu'il endosse les résultats et doit protéger son équipe tout en assumant ses choix.

2) Attitude : le product owner caméléon

L'attitude que je souhaite mettre en avant, c'est la posture du « non », cela n'est pas toujours facile mais nécessaire. Il faut prendre sur soi et démontrer ces arguments à vos équipes. Par exemple, le contexte est suivant, en tant que product owner vous devez valider ou rejeter des US, accepter ou refuser de les intégrer dans votre product backlog.

Autrement dit, vous devez changer d'attitude et parfois même d'altitude. Je m'explique, vous devez changer d'attitude pour savoir dire non et d'altitude pour bien vivre la situation, ce n'est pas de votre fait personnel après tout :

- « Non, on ne modifie pas le product backlog en cours de sprint »
- « Non, je n'intègre pas cette US avec une valeur aussi faible »
- « Non, les critères de DOD ne sont pas respectés, je ne valide pas l'US »

Ces quelques exemples pour vous montrer la difficulté que l'on peut rencontrer en tant que product owner et c'est bien l'attitude qui vous permet de surmonter ces difficultés.

Tout d'abord, qu'est-ce que l'attitude ? il s'agit d'une manière d'être qui manifeste certains sentiments ; comportement (source Larousse)

Dans une autre mesure, on a vu précédemment les différentes postures que le product owner pouvait avoir, c'est-à-dire, stratège, tacticien et facilitateur.

Quelle que soit sa posture, le product owner doit être fier de développer son produit. En ce sens, c'est un véritable caméléon qui ne perd jamais de vue de son objectif final : son produit.

Par ailleurs, le Product Owners embrasse les valeurs de l'agilité, dont la collaboration et l'entraide sont élémentaires comme nous avons pu le voir au début de ce guide. Ainsi, dans son quotidien, le product owner aime : communiquer, communiquer, communiquer… pour porter la vision du produit. Vous devez savoir vous adaptez à votre interlocuteur pour communiquer cette vision, d'où ce côté parfois caméléon qu'un product owner peut adopter.

De plus, je vous conseille d'adopter la posture de l'étudiant ou de l'élève qui veut toujours apprendre. En effet, rester curieux de tout : de votre environnement interne et externe, de votre produit, de votre équipe. Cela vous permettra d'être en mesure d'expliquer aux métiers les compromis et les décisions techniques prises par l'équipe de développement mais aussi d'être autonome sur les outils et les petites tâches techniques… un vrai luxe pour vos développeurs !

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Pour finir, vos deux plus grandes forces seront :

- votre adaptabilité. En effet, l'équipe accordera plus facilement sa confiance à un Product Owner si elle sent qu'il peut s'adapter rapidement et qu'il est capable de proposer des alternatives.
- votre capacité à être humble et fun : n'oubliez jamais de donner à l'équipe de développement le crédit qu'elle mérite !

C)Proposition de perspective sur les rôles du PO

Dans cette dernière partie, je souhaite aborder les interactions humaines du product owner. Le product est un peu comme un chef d'orchestre, il doit savoir interagir avec plusieurs groupes d'individu avec des caractéristiques, enjeux, objectifs parfois bien différents voir en opposition.

1)Gérer le change management

L'un des axes que je souhaite aborder dans cette partie est l'enjeu et le besoin de conduite de changement ou « change management » en tant que product owner selon un contexte spécifique.

Je suis actuellement dans une mission où cette problématique est structurante pour mettre en place la méthode agile. En effet, pour simplifier, j'ai d'un côté un groupe d'individu de l'équipe enthousiaste et volontaire pour mettre en place la méthodologie agile pour développer notre produit et d'un autre un groupe d'individu de la même équipe qui souhaite conserver les façons de faire actuelles, c'est-à-dire, le cycle en V traditionnel. Les comportements qui en découlent, comme vous pouvez l'imaginer, sont radicalement différent mais quelque part très enrichissant d'un point de vue externe.

Cette expérience m'a amené en plus de d'une autre au sein d'un groupe de sidérurgie Européen où la problématique de conduite de changement était également présente à me demander si la conduite de changement faisait partie du rôle de Product Owner, si oui, comment intégrer la conduite du changement à notre métier de product owner ? Quelles sont les limites et enfin les indicateurs clés du succès et échec (on apprend aussi grâce à l'échec)

Il existe plusieurs leviers en conduite de changement, ce que je retiens essentiellement pour conduire le changement est d'insister sur les émotions des groupes d'individu.

Selon Kotter, on peut mettre en place un plan d'action basé sur 8 étapes :

- Créer un sentiment d'urgence
- Former une coalition

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

- Développer une vision : on constate que la vision produit est un levier pour piloter du changement, pour garder l'objectif final en tête.
- Communiquer la vision
- Lever les obstacles au changement
- Démontrer des résultats à court terme
- Bâtir sur les premiers résultats pour accélérer le changement
- Ancrer les nouvelles pratiques dans la culture d'entreprise

Je motive mon équipe de développement grâce à un description occasionnelle de la vision du produit avec les bénéfices du planning et les impacts de la prochaine release du produit ou incrément

Le PO a des leviers pour conduire le changement à son niveau grâce à des actions telles que :

1. Je motive et forme mes développeurs en les impliquant dans la rédaction et l'analyse des US lorsque cela est possible
2. Je fais confiance dans les capacités de mon équipe de développement. Si non, j'essaye de construire de la confiance en leur offrant : de la formation, recrutement, une meilleure communication, des changements personnel

2) Un rôle de coach, être « coach product owner »

Je me suis demandée si le product owner n'avait pas une autre posture à adopter : celle de « coach ». C'est au cours de ma mission actuelle, où je me suis également posée cette question. En effet, je suis entourée de 4 product owner qui ont une application du rôle de product owner bien à eux et qui ne s'assimile pas à l'approche décrite depuis le début de cet ouvrage.

Ainsi, cela revient à coacher mes 4 product owner pour être plus à l'aise dans ce métier et leur apprendre de nouveau élément ?

Je souhaite tout d'abord définir ce que j'entends par coach, à mon sens c'est quelqu'un qui cherche à développer les performances d'un individu (salarié, product owner,...). A travers le processus de coaching, le product owner coaché approfondit ses connaissances et améliore ses performances.

Le rôle du coach pour être un « coach de product owner » :

Son rôle de partenaire requiert qu'il soit dans la position de comprendre les enjeux professionnels de son client sans toutefois être l'expert du métier de celui-ci

- Découvrir et clarifier ce que le product owner souhaite accomplir pour ensuite mesurer l'objectif avec des indicateurs de résultat suite au coaching.
- Aider le product owner à identifier ses motivations, ses stratégies et ses compétences.
- Montrer au product owner l'ensemble des options qui s'offrent à lui.

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

- Confronter le product owner à ses engagements ou à sa problématique, si cela peut le faire progresser.
- Aider le product owner à dévoiler l'ensemble des options qui s'offrent à lui.
- Aider le product owner à produire ses propres solutions et ses stratégies.

En tant que « coach product owner » vous engagez votre responsabilité. C'est pourquoi il est important d'alerter ses products owners lorsqu'ils constatent des dysfonctionnements. On peut alors une liste d'action associé au rôle de « coach product owner » :

- Donner au product owner client une orientation (vision produit) pertinente, choisir ses outils et définir les structures ad hoc ;
- Mettre en place une communication réussie qui permette de se concentrer sur les missions fondamentales de product owner. ;
- Participer aux rituels, cérémonies agiles avec le maximum d'efficacité et d'enthousiasme ;
- Concilier ses obligations liées aux enjeux du marketing stratégique, ses utilisateurs/clients et technique ;
- Exercer sa prise de décision (validation ou refus « savoir dire non avec plus d'efficacité) ;
- Faire face à des évolutions plus ou moins importantes de son environnement.
- Rétablir un contexte plus agile ;
- Réorienter ses priorisations ;
- Identifier ses enjeux et être performant plus rapidement dans sa prise en main de product owner.

Par conséquent et suite à la première partie concernant les compétences du product owner, je souhaite ajuster ce référentiel, à travers un certains aptitude et d'attitude à adopter :

Domaine de compétences	**Aptitude**	**Attitude**
Communication	En Co-créer la relation avec le client Communiquer avec efficacité Poser des questions pertinentes qui permettent de révéler les informations nécessaires. Pratiquer une communication directe et indirecte.	Construire un climat fondé sur la confiance et le respect. Créer une relation spontanée avec le client par une communication ouverte, souple et rassurante. Écouter avec attention les dits comme les non-dits et favoriser l'expression personnelle du client.
Analyse	Être capable d'intégrer et d'évaluer avec précision les sources multiples d'informations et présenter des	Co-créer des opportunités d'apprentissage continu, pendant les séances de

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

	suggestions qui aident le client à atteindre les objectifs fixés.	coaching et les situations quotidiennes, afin d'entreprendre de nouvelles actions qui généreront le plus efficacement possible les résultats souhaités.
Gestion de projet	Planifier et établir des objectifs communs sur les attendus du rôle de product owner	Être capable de centrer son attention sur ce qui est important pour le client et le laisser assumer la responsabilité de ses actions.

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Le coach « product owner » peut apporter des réponses à des questions de type :

- Quel doit être mon implication ?
- Comment collaborer efficacement avec l'équipe ?
- Comment communiquer avec l'extérieur ?
- Quels outils utilisés ?
- Comment fixer des priorités ?
- Quel niveau de granularité pour les items de Backlog de produit ?

3) La pédagogie : l'outil inattendu du PO

Nous avons un métier qui demande de la diplomatie, de la fermeté et surtout une grande capacité d'adaptation relationnelle à travers une certaine forme de pédagogie envers ses interlocuteurs. Je pense que c'est un facteur clé du succès, sauf que voilà, parfois nous savons faire preuve d'une grande empathie mais aussi comprendre les enjeux de nos interlocuteurs et cela ne suffit pas pour atteindre nos objectifs liés aux valeurs et à la mise en place de la méthode agile.

C'est l'une des raisons pour laquelle, je me suis intéressée en quelque sorte à la psychologie. En effet, l'agilité est une véritable transformation d'un point de vue de gestion de projet mais aussi humain.

A mon sens, lorsque l'on travaille pendant plus de 20 ans dans une entreprise où la hiérarchie règne avec la mise en place de « fief » comme j'aime les appeler ou des territoires si vous voulez, la rétention d'information, l'opacité dans les échanges, l'hypocrisie face à la remontée des vrais problèmes de fond, le manque d'enthousiasme et d'honnêteté par rapport au fait et rien que les faits. Ce n'est pas étonnant que l'agilité et dans une autre mesure que le rôle de PO soit parfois complexe ! J'ai eu la chance récemment d'avoir une expérience dans un contexte où l'on était en mode start-up, toute la liste précédente se voit transformer.

Pourquoi est-ce que je souhaite parler de psychologie ? Tout simplement parce que cela est nécessaire pour comprendre notre utilisateur. Quels sont ses attentes, ses besoins, ses peurs, son style de vie. Par ailleurs, nous travaillons en équipe, nos développeurs n'ont pas forcément le même schéma de pensée. Puis, il y a l'aspect personnel, quels sont les valeurs de la personne en face de moi quelle qu'elle soit : nos utilisateurs, notre client, notre développeur, notre scrum master. C'est pourquoi, je vous propose cet outil qui permet de cerner les différentes valeurs possibles, ainsi par le comportement, les actions, les échanges vous pouvez identifier celles de votre interlocuteur et par conséquent mieux le comprendre et permettre des échanges plus constructifs. C'est la théorie des 10 valeurs universelle de Schwartz. Vous pouvez ensuite déterminer, bien entendu, de manière relative les valeurs prioritaires et secondaires… et attention à ne pas vous méprendre sur votre propre interprétation, perception et/ou préjugé que vous pouvez avoir en premier lieu, ils peuvent être trompeurs !

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Valeur	Objectif	Caractéristiques	Comportement
Autonomie	Indépendance de la pensée et de l'action : choisir, créer, explorer.	Découle du besoin de contrôle et de maîtrise et les exigences d'interactions nécessaires à l'autonomie et à l'indépendance.	créativité, liberté, choisissant ses propres buts, curieux, indépendant ainsi que [amour propre, intelligent, droit à une vie privée]).
Stimulation	enthousiasme, nouveauté et défis à relever dans la vie	Découle du besoin de variété et de stimulation ; elles permettent de maintenir un niveau d'activité optimal et positif tout en écartant la menace qui amènerait un niveau trop élevé de stimulation.	une vie variée, une vie passionnante, intrépide. Probablement en relation avec ceux qui sous-tendent les valeurs d'autonomie.
Hédonisme.	plaisir ou gratification sensuelle personnelle	Les valeurs d'hédonisme proviennent des besoins vitaux de l'être humain et du plaisir associé à leur satisfaction.	plaisir, aimant la vie, se faire plaisir
Réussite	le succès personnel obtenu grâce aux compétences socialement reconnues	Être performant dans la création ou l'accès à des ressources est une nécessité pour la survie des individus ;. le fait d'être performant au regard des normes culturelles dominantes, et d'obtenir ainsi l'approbation sociale.	ambitieux, ayant du succès, capable, ayant de l'influence ainsi que [intelligent, amour-propre, reconnaissance sociale
Pouvoir	statut social prestigieux, contrôle des ressources et domination des personnes.	Les valeurs de pouvoir peuvent aussi découler des aspirations individuelles au contrôle et la domination	Items associés : autorité, richesse, pouvoir social ainsi que [préservant mon image publique, reconnaissance sociale]).
Sécurité	sûreté, harmonie et stabilité de la société, des relations entre groupes et entre individus, et de	Les valeurs de sécurité découlent des nécessités fondamentales du groupe et de l'individu. Il y a deux sortes de valeurs de sécurité. Certaines concernent avant tout des intérêts individuels (par exemple, propre), d'autres concernent surtout des intérêts	ordre social, sécurité familiale, sécurité nationale, propre, réciprocité des services rendus ainsi que [en bonne santé, modéré, sentiment d'appartenance]).

(cc) BY-NC-ND

	soi-même.	collectifs (par exemple, sécurité nationale). Mais même ces derniers sont liés, de manière non négligeable, à un objectif de sécurité pour soi-même (ou pour ceux auxquels on s'identifie). Les deux sortes de valeurs de sécurité peuvent donc être réunies dans une valeur qui les englobe.	
Conformité	modération des actions, des goûts, des préférences et des impulsions susceptibles de déstabiliser ou de blesser les autres, ou encore de transgresser les attentes ou les normes sociales.	Les valeurs de conformité proviennent de la nécessité pour les individus d'inhiber ceux de leurs désirs qui pourraient contrarier ou entraver le bon fonctionnement des interactions et du groupe. Telles que je les conçois, les valeurs de conformité concernent l'autolimitation dans les interactions quotidiennes, généralement avec des personnes proches.	(Items associés : obéissant, auto-discipliné, politesse, honorant ses parents et les anciens ainsi que [loyal, responsable]).
Tradition.	respect, engagement et acceptation des coutumes et des idées soutenues par la culture ou la religion auxquelles on se rattache.	La tradition est liée aux pratiques, des symboles, des idées et des croyances qui représentent leur expérience et leur destin commun et deviennent ainsi les coutumes et les traditions du groupe, qui leur accorde beaucoup de valeur. Ces coutumes et traditions deviennent l'expression de la solidarité du groupe, expriment sa valeur singulière et contribuent à sa survie. Elles prennent souvent la forme de rites religieux, de croyances, et de normes de comportement.	(Items associés : respect de la tradition, humble, religieux, acceptant ma part dans la vie ainsi que [modéré, vie spirituelle]).

Remarque : Le pouvoir et la réussite sont deux valeurs qui visent la reconnaissance sociale. Cependant, les valeurs de réussite (par exemple, ambitieux) mettent l'accent sur la démonstration d'une compétence effective lors d'une interaction concrète, tandis que les valeurs de pouvoir (par exemple autorité, richesse) concernent plutôt le fait d'atteindre ou de conserver une position dominante à l'intérieur d'un système social plus global.

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Ces différents aspects de valeur sont plus facilement observables pour vos collaborateurs et faciliter une cohésion d'équipe. Cependant, avec une analyse des insights de vos clients, vous pouvez également déceler selon leur comportement, les valeurs associées.

Pour terminer, je vous propose d'analyser les différents profils MBTI, un test de personnalité qui vous permettra également de mieux communiquer, de mieux comprendre les émotions de vos clients et leurs trouver des solutions personnalisées.

Les dimensions du test sont les suivantes :

- Orientation de l'énergie : E Extraversion / I Introversion
- Recueil d'information : S Sensation / N Intuition
- Prise de décision : T Pensée / F Sentiment
- Mode d'action : J Jugement / P Perception

Voici les 16 types de personnalités et leurs qualités :

Rôles Stratégiques				Qualité
NT	Construit	**ENTP**	Invente	Inventif
Rationnels		**INTP**	Conçoit	Logique
Travaille avec	Organise	**ENTJ**	Mobilise	Commandement
les systèmes		**INTJ**	Implique	Indépendant
Rôles Diplomatiques				
NF	Arbitre	**ENFP**	Motive	Optimiste
Idéalistes		**INFP**	Concilie	Non directif
Travaille avec	Développe	**ENFJ**	Éduque	Persuasif
les personnes		**INFJ**	Guide	Empathique
Rôles Logistiques				
SJ	Soutient	**ESFJ**	Pourvoie	Conciliateur
Gardiens		**ISFJ**	Protége	Fidèle
Travaille avec	Réglemente	**ESTJ**	Applique	Demandeur
le matériel		**ISTJ**	Certifie	Puissant / Silencieux
Rôles Tactiques				
SP	Improvise	**ESFP**	Démontre	Généreux
Artisans		**ISFP**	Synthétise	Artistique
Travaille avec	Facilite	**ESTP**	Persuade	Imprévisible
l'outillage		**ISTP**	Instrumente	Manuel

Il s'agit d'un sujet très intéressant et qui devrait être développé dans un chapitre à part entière.

Si vous souhaitez plus d'information, je souhaite créer prochaine un rapport pour traiter ces différents aspects.

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Conclusion

Le métier de product owner est un nouveau métier, difficile à cerner pour certaines entreprises voir même pour certains product owner. J'espère que cet ouvrage vous aura plu et surtout qu'il vous aura apporter des éléments de réponses à vos questions.

Nous avons vu le rôle du Product Owner, au sein d'un projet Scrum : c'est le responsable du succès du produit et le représentant des utilisateurs. Être Product Owner c'est choisir un rôle « pivot » et exigeant, il nécessite une bonne connaissance des pratiques agiles de recueil des besoins, de priorisation et de pilotage par la valeur.

Nous avons vu aussi, les outils principaux pour réaliser toutes les étapes du cycle de vie du produit numérique. Comme vous pouvez vous en doutez, c'est une liste exhaustive, j'ai déjà des idées pour vous proposer des guides de poches sur des thèmes plus spécifiques.

Vous pouvez aussi me faire part de vos retours concernant cet ouvrage, nous sommes en agile ! n'hésitez pas à me communiquer ce que vous avez aimé, ce que vous n'avez pas aimé et ce que vous souhaiteriez comme amélioration.

L'agile est une méthodologie et une pratique qui intègre des thèmes variées, vous pouvez vous aussi, continuer de vous améliorer ou même créer votre propre Framework comme cela a été le cas pour le framework SaFe. Le Scaled Agile Framework (SAFe) est un cadre agile tentant de répondre aux nombreux défis auxquels doit faire face une grande entreprise lors d'une transition agile. C'est un point de départ solide pour quiconque voudrait démarrer une initiative d'agilité à grande échelle.

De mon point de vue personnel, je compare ce métier à de la haute voltige les bons jours et gardien de zoo les mauvais jours. Vous me demandez pourquoi la haute voltige ? Tout simplement pour l'enthousiasme, pour le dynamisme, pour la vitesse et les allers-retours à toute allure pour prendre une nouvelle direction et enfin pour ce petit sourire en coin comme les pilotes lorsque vous redescendez de vos nuages quand le produit a finalement été livré dans les temps avec la qualité et le budget souhaité.

J'aime ce métier qui m'épanouis, les difficultés doivent être surmontées avec équipe, n'oubliez jamais cela : vous n'êtes jamais seul face à un problème vis-à-vis du produit : c'est le problème et l'enjeu de toutes les parties prenantes.

Je vous recommande enfin de toujours faire de la veille sur les nouveaux outils, les nouvelles pratiques misent en œuvre, les retours d'expériences des autres product owner ou encore les conseils des coachs agile.

A bientôt pour un nouveau guide,

Ségolène Porot

beNext Company

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Annexes

Nous avons abordé plusieurs outils dans cet ouvrage, voici les mots clés :

Impact mapping, Story map, KANBAN, Rituels agiles, Analyse du marché, Analyse des utilisateurs, Lean startup, Picth canva, Business model canva, Frugal innovation, Vision produit, Stakeholder map, Lean canva, Product box, Remember the future, Speed boat, Produit canva, Elevator speech, Tableau de bord produit, Product roadmap, Go product roadmap, Les 3C, les US Persona, Empathy map, Conduit d'entretien, Innovation game, Gherkin , Critères d'acceptation , BDD : bahaviour driven development, Definition of done , US check list, Product backlog, MoSoCoW, Fibonacci, Taille de T-shirt , Planning poker Burndown chart, Burnup chart, product backlog refinement memo, User grooming Session, découper les US, design studio, experience map/customer journey, design thinking, customer value canva, feature team, design sprint, ROTI, Lean UX, UX testing, Wireframing, analytics, event storming, value stream mapping, extreme quotation,Ishikawa, Gyshido, visual management, referentiel métier de product owner, 10 valeurs de Schwartz.

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Annexe 1 : la journée type d'un product owner

(source : http://www.exam-pm.com/scrum-la-journee-type-dun-product-owner/)

Voici Pat. Elle est Product Owner. Elle est vraiment passionnée par la vision qu'elle a du produit. Elle ne connait pas le détail de ce que son produit va faire, mais elle sait pourquoi on fait ce produit, quels problèmes il va résoudre et pour qui. Elle en parle tout le temps.
Voici les autres Parties Prenantes (PP). Ceux qui vont utiliser, maintenir, ou être de près ou de loin concernés par le produit en cours de développement. D'après la vision de Pat, tous ces gens vont adorer notre système, l'utiliser et en parler tout le temps. Les besoins et les idées des PP sont exprimés comme des « histoires utilisateur » (US). Par exemple, pour un système de réservation de vol, les gens doivent pouvoir chercher un vol. Ça serait une des US. Pat et les PP ont plein d'idées, Pat les aide donc à les convertir en US concrètes.
Maintenant, quelqu'un doit FABRIQUER le système. Les voila : petite, co-localisée, multi-compétente, auto-organisée : l'équipe de développement. Vu que c'est une équipe agile, ils ne gardent pas une grosse livraison complète pour la fin, mais ils livrent tôt et souvent. Dans notre cas, ils livrent habituellement 4-6 US par semaine, c'est donc leur capacité. La capacité est facile à mesurer : compter le nombre d'US livrées par semaine. Certaines US sont grosses, elles comptent pour 2, d'autres sont petites, elles comptent pour la moitié. Mais au final c'est environ 4-6 US par semaine. Certains parlent de « story points », mais je dirai simplement des « User Stories », par semaine. Pour maintenir ce rythme, et ne pas être embêté par des tests de régression manuels, l'Equipe a fortement investi dans des outils de tests automatiques et d'intégration continue (CI). Donc chaque US a au moins un test d'acceptation automatique au niveau fonctionnel, et la majorité du code a des tests unitaires automatiques. Le problème, c'est qu'il y a un paquet de PP qui demande plein de choses, et ils ne vont surement pas se limiter à 4-6 idées par semaine. Ils ont PLEIN d'idées et PLEIN de souhaits. Et chaque fois qu'on leur livre quelque chose, ça leur donne encore plus d'idées et ils demandent encore plus de trucs ! Que se passe-t-il si on essaie de leur faire plaisir, de faire tout ce qu'ils demandent ? On est submergé. Supposons que l'Equipe commence à travailler sur 10 nouvelles US par semaine. Si on a 10 en entrée et 4-6 en sortie, l'Equipe sera surchargée. Cela entrainera multitâches, démotivation et finalement une sortie plus faible et de mauvaise qualité. C'est une proposition perdant-perdant. C'est comme essayer de fourrer plus de papier dans une imprimante pour qu'elle imprime plus vite, ou ajouter encore des voitures sur une autoroute déjà saturée. Ça ne marche pas : ça fait qu'empirer les choses. Alors, on fait quoi ?
Le moyen d'éviter ce problème en Scrum et XP est appelé « la météo d'hier ». L'Equipe dit : « ces dernières semaines, on a terminé 4-6 fonctionnalités par semaine. Donc quelles 4-6 fonctionnalités devrions-nous faire cette semaine ? ». Le boulot du PO est de trouver, parmi toutes les US de l'univers, les 4-6 prochaines US à livrer. L'approche Kanban consiste à limiter le TAF (Travail A Faire) ou WIP (Work In Progress). Si l'Equipe décide que le nombre optimal de US à traiter simultanément est 5, ils ont appris que c'est juste assez pour occuper -sans surcharger- tout le monde, Ils ont donc décidé que leur limite de WIP est 5. Dès qu'une US est terminée, une nouvelle sera acceptée, s'assurant ainsi que la limite de 5 US en cours ne sera jamais dépassée.
Ces deux approches fonctionnent bien ; comme l'équipe aura juste assez de travail pour travailler rapidement et efficacement; Avec un effet de bord cependant : il va se former une file d'attente en amont de l'Equipe, c'est ce qui, en Scrum, est appelé le Backlog Produit

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

(PBL). Cette file d'attente doit être gérée. Si les PP persistent à demander 10 nouvelles US par semaine et que l'Equipe en réalise 4-6 par semaine, ça veut dire que la file grandira indéfiniment. Avant d'avoir eu le temps de vous en rendre compte, vous aurez une liste de 6 mois dans le PBL. Ce qui veut dire qu'en moyenne, chaque US réalisée répondra à un besoin exprimé 6 mois avant. C'est agile ça ? Il n'y a vraiment qu'une seule façon de ne pas perdre le contrôle de la file d'attente. C'est le mot NON. C'est le mot le plus important pour un PO, et Pat s'entraine tous les jours devant son miroir. Dire OUI à une nouvelle demande est facile. Surtout si ça veut simplement dire l'ajouter à un backlog qui ne fait que grossir. Le travail le plus important pour un PO est de décider ce qu'il NE faut PAS construire, et penser aux conséquences de cette décision. Et c'est pour ça que c'est difficile, bien sûr ! Le PO décide ce qui est embarqué et ce qui ne l'est pas. Le PO décide aussi de l'ordre : ce qu'on fait maintenant, ce qu'on fait plus tard, et de la longueur que cette liste doit avoir. C'est un travail difficile donc Pat ne le fait pas toute seule, elle le fait en collaboration avec les PP et l'Equipe. Pour pouvoir prioriser, le PO doit avoir une idée de la valeur (métier) des US ainsi que de leur taille. Certaines sont critiques, indispensables, alors que d'autres, des fonctionnalités bonus. Certaines prennent quelques heures à faire, d'autres des mois. Maintenant, devinez le rapport entre la valeur d'une US et sa taille ? Et oui : aucun ! Plus gros ne signifie pas mieux. Pensez à n'importe quel système que vous avez utilisé. Je parie que vous pouvez penser à au moins une fonctionnalité vraiment simple, mais très importante, que vous utilisez tous les jours. Je parie que vous pouvez penser à au moins une énorme fonctionnalité bien compliquée et qui n'a aucune importance. La valeur et la taille, c'est ce qui aide Pat à prioriser intelligemment. Comme ici : ces deux US ont à peu près la même taille, mais ont une valeur différente. Donc faire celle-ci en premier. Et là, ces deux US ont presque la même valeur, mais pas la même taille. Donc faire celle-là en premier, etc. OK, ça a l'air assez facile…Mais, une seconde… Comment connait-elle la valeur d'une US ? Comment connait-elle sa taille ? Et bien voici la mauvaise nouvelle : elle ne les connait pas. C'est un jeu de devinettes. Et c'est un jeu qui implique tout le monde !
Pat parle en permanence avec les PP pour voir ce qui a de la valeur pour eux. Et elle parle en permanence avec l'Equipe pour voir ce qui est petit ou gros pour eux en termes d'effort d'implémentation. Ce sont des estimations relatives, pas des nombres absolus. Je ne sais pas combien pèse cette pomme ou cette fraise, mais je sais que la pomme est 5 fois plus lourde, et que la fraise a meilleur goût, du moins pour moi. C'est tout ce que Pat doit savoir pour prioriser le PBL : C'est plutôt cool comme ça. Au début d'un nouveau projet, nos estimations sont forcément mauvaises. Mais c'est OK, la plus grande valeur est en fait dans les conversations plutôt que dans les chiffres. Et chaque fois que l'Equipe livre quelque chose aux vrais utilisateurs, on apprend quelque chose, et on devient meilleur pour estimer la valeur et la taille. C'est pourquoi on estime et priorise en permanence. Essayer de tomber juste dès le début est assez bête : c'est là qu'on en connait le moins. La boucle de feedback est notre amie ! Cependant la priorisation n'est pas suffisante. Pour livrer tôt et souvent, on doit décomposer les US en bouchées comestibles, de préférence juste quelques jours de travail par US. On veut cette forme d'entonnoir sympa, avec des petites US bien claires à l'avant et des US plus vagues à l'arrière. En faisant cette décomposition des US « juste à temps », on peut profiter des dernières connaissances sur le produit et les besoins utilisateur. Tout ce dont on vient de parler – estimer la valeur et la taille des US, prioriser, décomposer – s'appelle habituellement l'« affinage » [ou l'entretien] du Backlog. Pat mène un atelier d'affinage tous les mercredis de 11h à 12h. Une heure par semaine.

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Généralement, toute l'Equipe est présente, et parfois quelques autres PP. Le programme peut varier un peu : on se focalise parfois sur les estimations, parfois sur la décomposition d'US, sur l'écriture de critères d'acceptation pour une US, etc. J'espère que vous remarquez la thématique ici : la communication ! Le rôle de PO c'est avant tout de la communication. Quand je demande à des PO expérimentés ce qu'il faut pour réussir, ils mettent généralement en avant la passion et la communication. Ce n'est donc pas une coïncidence que le premier principe du manifeste agile soit « Individus et interactions avant processus et outils ». Le rôle du PO n'est donc pas de nourrir l'Equipe à la petite cuiller avec des US. C'est ennuyeux et inefficace. Pat s'assure plutôt que tout le monde comprend la vision, que l'Equipe a un bon contact avec les autres participants, que les livraisons sont fréquentes et permettent un feedback rapide des vrais utilisateurs. Ainsi l'Equipe apprend et peut prendre elle-même quotidiennement des décisions, permettant à Pat de se focaliser sur la vue d'ensemble. Regardons quelques-uns des compromis qui doivent être trouvés par Pat et l'Equipe. Il y a avant tout les compromis entre les différents types de valeur. En début de projet, l'incertitude et le risque sont nos ennemis. Il y a le risque métier : construit-on la bonne chose ? Il y a le risque social : ces personnes peuvent-elles la construire ? Il y a le risque technique : Ça marchera sur la plateforme cible ? Ça passera bien à l'échelle ? Et il y a les risques sur les couts et les délais : Peut-on finir dans des délais raisonnables ? Et avec des coûts raisonnables ? La connaissance peut être vue comme le contraire du risque. Donc quand l'incertitude est grande, on se concentre sur l'acquisition de connaissance, comme des prototypes d'interface utilisateur, ou des expérimentations techniques (Spikes). Peut-être pas trop excitant pour les clients, mais cela a de la valeur car on réduit les risques. Du point de vue du client, la courbe ressemble à ça au début. Et comme l'incertitude diminue, on se concentre progressivement sur la valeur client. On sait ce qu'on va construire et comment, alors allons-y ! Et en commençant par les US de plus grande valeur, on obtient cette belle courbe de valeur escarpée. Puis progressivement, la courbe s'aplatit. On a produit les choses les plus importantes, maintenant, on ajoute simplement les fonctionnalités « bonus », les garnitures sur la crème glacée. C'est un endroit confortable car à tout moment Pat et l'Equipe peuvent décider de « couper la queue » de liste Couper juste ici Et passer à un autre projet plus important, ou commencer un tout nouveau domaine fonctionnel sur le même produit. C'est ça l'agilité métier. Donc quand je parle de Valeur ici, je parle en fait de valeur-Connaissance + de valeur-Client. Et on doit trouver un compromis entre ces valeurs. Un autre compromis est : penser à court terme ou à long terme. Que doit-on faire maintenant ? Traiter ces bugs urgents ou développer cette nouvelle fonction géniale qui va éblouir les utilisateurs, ou cette mise à jour de plateforme difficile qui va sûrement nous permettre de développer plus rapidement par la suite ? On doit en permanence trouver l'équilibre entre le travail réactif et le travail préventif "Combattre le feu" ou "éviter qu'il y ait le feu". Un autre type de compromis : Doit-on se focaliser sur « faire les bonnes choses » ? ou « bien faire les choses » ? ou peut-être « faire les choses rapidement » ? Idéalement, on veut les trois, mais pas facile de trouver l'équilibre. Supposons qu'on soit ici : on veut faire le produit parfait avec son architecture parfaite si on passe trop de temps à essayer d'être parfait, on peut rater une période de mise sur le marché, ou avoir des problèmes de trésorerie. Supposons qu'on est ici, on se dépêche de transformer un prototype en produit utilisable. Super pour le court terme, peut-être, mais sur le long terme, on peut se noyer dans la dette technique et notre vélocité s'approcher de zéro. Supposons qu'on est ici, on construit une superbe cathédrale en un temps record. Sauf

(cc) BY-NC-ND

que les utilisateurs n'avaient pas besoin d'une cathédrale mais d'un camping-car. Il y a une tension saine entre les différents rôles Scrum. Les PO se concentrent sur « construire les bonnes choses », les Equipes sur « bien construire les choses ». Et les Scrummasters ou coaches agiles sur « raccourcir la boucle de feedback ». Ça vaut le coût d'insister sur la vitesse. Car une boucle de feedback courte accélèrera l'apprentissage, on apprendra plus rapidement ce qu'est « la bonne chose », et comment « bien la faire ». Les trois vues sont importantes, essayez de trouver l'équilibre. Il y a enfin le compromis entre développer un nouveau produit et améliorer un ancien produit. « Product Backlog » est un terme assez déroutant car il sous-entend qu'il n'y a qu'un seul produit. Et « projet » est aussi ambigu, il sous-entend que les développements ont une fin.
Mais un produit n'est jamais vraiment « terminé », il y a toujours de la maintenance et des améliorations à faire, tout au long de la vie du produit, jusqu'au moment ou il est définitivement arrêté. Donc quand une équipe commence à développer un nouveau produit, que se passe-t-il pour le précédent ? Transmettre un produit d'une équipe à une autre est couteux et risqué. Un scénario plus habituel est que l'équipe continue à maintenir l'ancien produit tout en développant le nouveau. Donc on n'a plus vraiment un PBL, mais plutôt un Backlog d'Equipe – une liste de choses que le PO veut que l'Equipe produise – et ça peut être un mélange de choses pour différents produits sur lesquel le PO doit trouver des compromis en permanence. De temps en temps, un Participant appellera Pat et dira : « Hé, mon truc sera fini quand ? », ou « j'aurai quoi de ce que j'ai demandé à noël ? ». En tant que PO, Pat est responsable de la gestion des attentes, ou plutôt des attentes réalistes. Ça veut dire : pas de mensonge. Je sais, c'est pas facile, mais qui a dit que l'Agilité, c'était facile ? Ce n'est pas si difficile de faire une prévision, tant qu'elle ne doit pas forcément être exacte. Si vous mesurez la vélocité de votre Equipe, ou de toutes vos Equipes, vous pouvez dessiner un burnup chart. Comme ça. Ce graphe montre le nombre cumulé d'US au cours du temps ou de story points si vous préférez. Notez la différence : cette courbe montre une production, celle-ci, un résultat. C'est la sortie, et c'est le résultat qu'on espère atteindre avec la production. Notre but n'est pas d'avoir le plus de production possible, mais d'atteindre le résultat voulu – des PP contentes – avec le moins de production possible. Moins, c'est plus. Regardez le burn up chart et dessinez les courbes de tendance optimiste et pessimiste. On peut faire ça avec statistiques vaudoues fantaisistes ou juste à la main. L'écart entre ces lignes dépend bien sûr de la prédictibilité de votre vélocité. Heureusement, elle tend à se stabiliser dans le temps, donc notre cône d'incertitude devrait s'affiner de plus en plus. OK, retour sur la gestion des attentes. Supposons que les PP demandent à Pat « Quand tout ÇA va être fait ? » « Quand serons-nous ici ? » C'est une question périmètre fixe/temps variable. Pat utilise les 2 lignes de tendance pour répondre « le plus probablement entre avril et mi-mai » Supposons que les PP demandent à Pat : « On en aura combien à noël ? » C'est une question temps fixe, périmètre variable. Les lignes de tendance nous disent « on aura probablement fini tout ça, à noël, un peu de ça et rien de ça ». Enfin, supposons que les PP disent : « Peut-on avoir CES fonctionnalités la pour noël ? » C'est une question à périmètre fixe et temps fixe. En regardant les courbes de tendance, Pat dit « Non désolée, ça n'arrivera pas » suivi de : « voila ce qui peut être fait pour noël » ou « voila de combien de temps on a besoin pour tout faire » C'est généralement mieux de réduire le périmètre plutôt que d'allonger les délais car si on réduit en premier le périmètre, on aura toujours le temps d'allonger les délais par la suite pour ajouter le reste des US. L'inverse ne fonctionne pas parce que –flûte !!– on ne peut pas remonter le temps ! C'est connu : Le temps est assez

(cc) BY-NC-ND

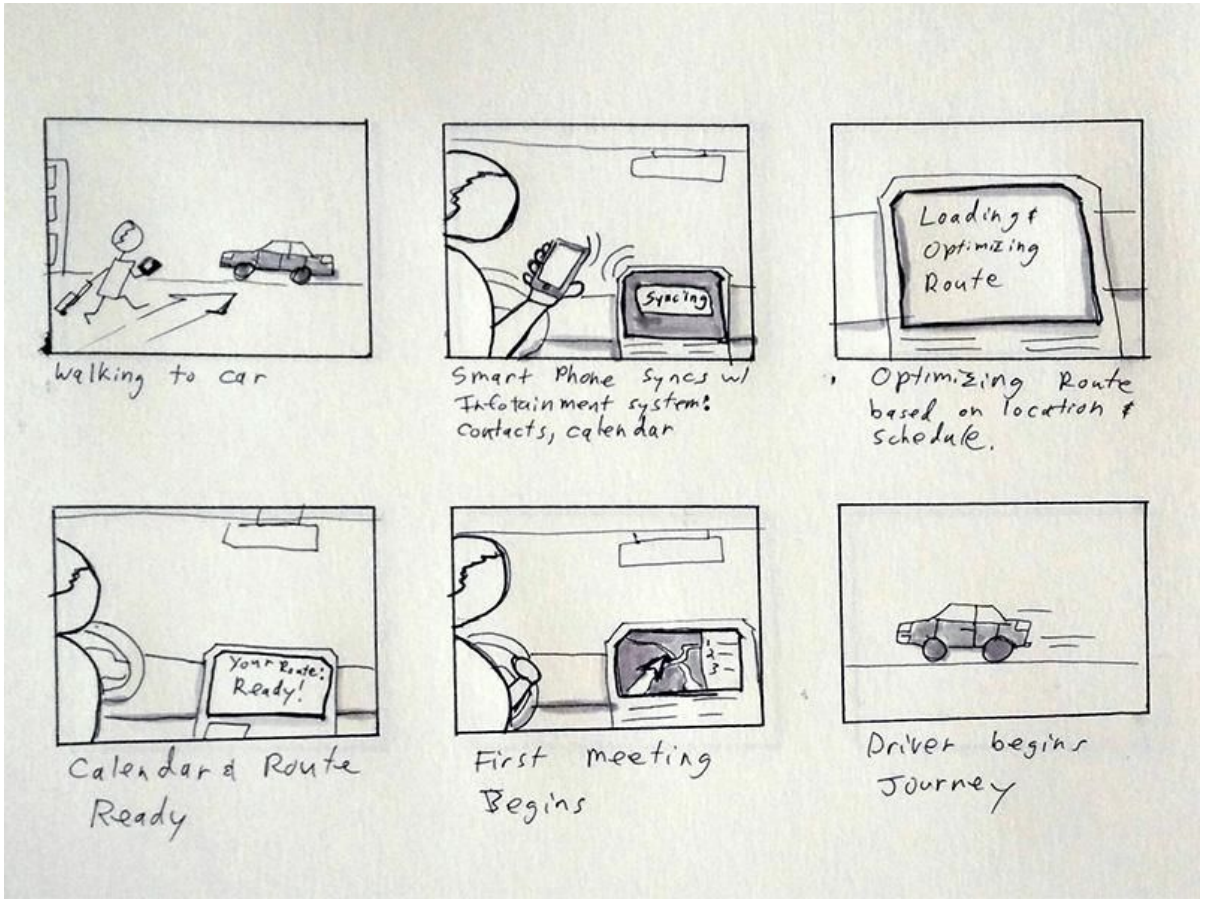
Ségolène Porot - licence CC 4.0

pénible pour ça ! Pat le dit comme ça : « on peut livrer quelque chose ici, et le reste après, ou alors rien ici, et le reste après. Vous préférez quoi ? » Les calculs sont simples à faire, donc Pat met à jour les prévision toutes les semaines. Le plus important ici, c'est qu'on utilise des vraies données pour faire les prévisions, et on est honnête sur l'incertitude. J'avais pas dit "pas de mensonges", pas vrai ? Et c'est un moyen très honnête de communiquer avec les PP, et en général ils apprécient beaucoup. Si votre organisation n'aime pas la vérité et l'honnêteté, elle n'aimera probablement pas l'agilité. Petit avertissement cependant : Si l'équipe accumule de la dette technique –s'ils n'écrivent pas de tests, et n'améliorent pas l'architecture en permanence– ils seront de plus en plus lents au fil du temps et la courbe de burnup s'aplatira progressivement. Ce qui rendra quasiment impossible pour Pat de faire des prévisions. Donc l'équipe est responsable du maintien d'un rythme soutenable, et Pat évite de les mettre sous pression en prenant des raccourcis. Et si on avait un projet plus gros avec des équipes multiples ? Et plusieurs PO, chacun avec son backlog pour une partie différente du produit. Globalement, le modèle est le même. On a toujours besoin de gérer la capacité, on a toujours besoin de communiquer avec les PP, on a toujours besoin de PO qui peuvent dire NON, on a toujours besoin d'affiner le backlog on a toujours besoin de feedbacks rapides, etc. La vélocité est la somme de toutes les sorties et peut être utilisée pour faire les prévisions. Ou faites des prévisions distinctes pour chaque équipe si cela a plus de sens. Dans un scénario multi-équipes cependant, les PO ont une responsabilité supplémentaire très importante : se parler entre eux ! On doit organiser les équipes et les backlogs pour réduire les dépendances. Mais il y aura toujours des dépendances. On a donc besoin d'une synchronisation entre les PO pour construire les choses dans le bon ordre et pour éviter la sous-optimisation. Dans les gros projets, on fait généralement appel à une sorte de Chef des PO pour synchroniser les PO. Voila, c'est fini !

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Annexes Visuel n°1 : le Story map

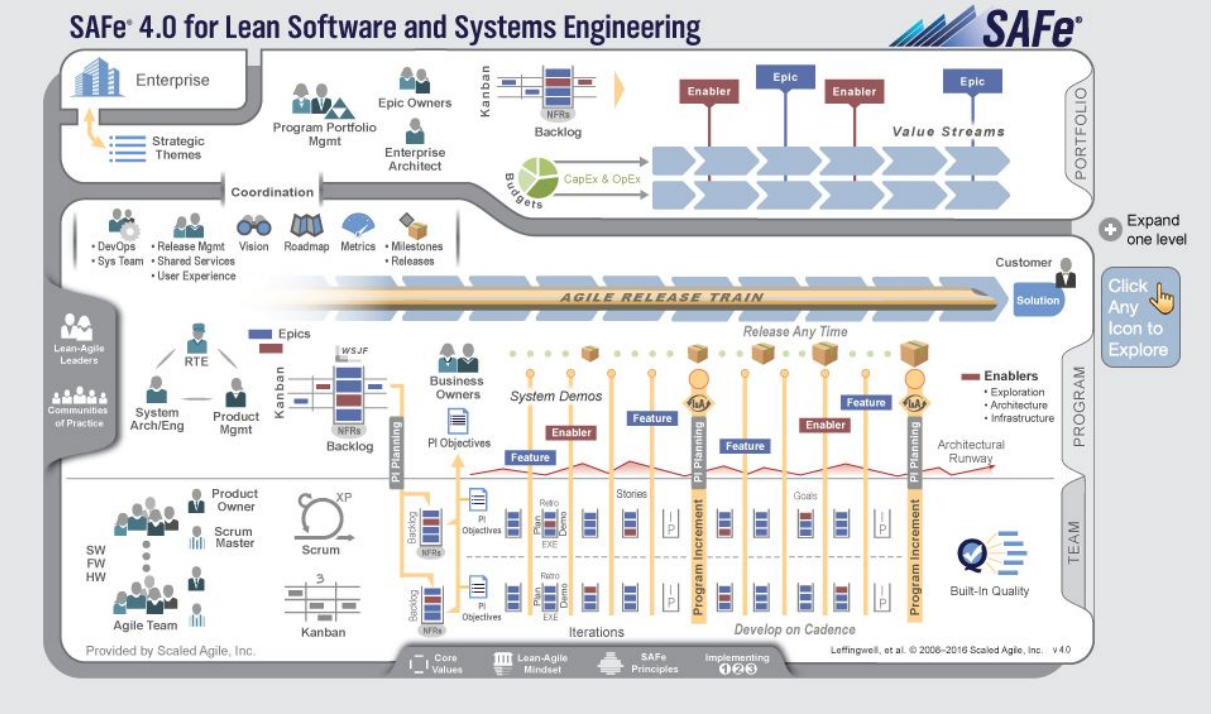


(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Annexe Visuel n°2 : le framework SAFE

(source : http://developpementagile.com/posts/2013/march/safe_fondements)



(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Sources :

Mes sources d'inspiration pour cet ouvrage sont multiples :

- http://www.agiliste.fr/guide-de-demarrage-scrum/ (l'image de l'US sous le format card)
- (https://lebuzzdubuzz.wordpress.com/2014/03/31/atelier-vision-et-enjeux/)
- http://www.areyouagile.com/2015/09/trois-apprentissages-clefs-dune-approche-agile/
- https://www.competentia.be/comment-acquerir-des-competences
- http://www.van-rooden.nl/blog/2016/03/10/product-backlog-refinement-explained-33/
- http://www.romanpichler.com/
- http://developpementagile.com/posts/2013/march/safe_fondements
- http://www.16-types.fr/16types.html
-

(cc) BY-NC-ND

Ségolène Porot - licence CC 4.0

Mes recommandations :

- http://blog.crisp.se/
- http://www.areyouagile.com/ (blog de Pablo Pernot)
- http://www.andwhatif.fr/ (blog Dragos Dreptate)
- http://blog.soat.fr/tag/pablo-pernot/
- http://lucyinthescrum.com/ (blog de Lucie Lesage)
- http://www.thiga.fr/
- http://www.romanpichler.com/
-

Ce(tte) œuvre est mise à disposition selon les termes de la Licence Creative Commons Attribution - Pas d'Utilisation Commerciale - Pas de Modification 4.0 International.