

SCRUMBLE

LES REGLES DU JEU

UN JEU AGILE PAR

PYXIS
savoir agile, faire mieux

NOTES ET REMERCIEMENTS

Ce guide a été mis à jour le 15 février 2016. Pensez à vérifier qu'il s'agit bien de la version la plus récente!

A PROPOS DU JEU

Scrumble est né en 2014 dans l'esprit fantaisiste de Romain Trocherie, coach Agile et Scrum Master à [Pyxis Suisse](#). Après tout, si Scrum vient en partie d'un jeu, pourquoi ne pas créer un jeu sur Scrum? Telle était l'idée de départ.

La conception du jeu et les expérimentations ont duré de longues semaines, et Scrumble continuera d'évoluer selon vos expériences! Pour toute remarque ou suggestion, n'hésitez pas à communiquer avec les Pyxisiens par courriel à scrumble@pyxis-tech.com.

LICENCE ET DROITS D'UTILISATION

Le jeu Scrumble est entièrement gratuit, jouable par tous, peu importe le lieu et le moment.

Néanmoins, nous tenons à préciser que celui-ci a fait l'objet d'un travail conséquent de la part de l'équipe de [Pyxis Suisse](#) et que l'objectif final du jeu, c'est-à-dire promouvoir un fonctionnement idéal des équipes selon les principes de Scrum, ne doit pas être altéré. Les véritables principes et règles de Scrum faisant foi sont décrits sur www.scrumguides.org.

A ce titre, le jeu et ses différents supports sont distribués sous licence ([Creative Commons Attribution – Pas d'Utilisation Commerciale – Pas de Modification 4.0 International](#)). Ainsi, vous pouvez les partager en citant les auteurs, sans modification et sans contrepartie financière.



REMERCIEMENTS

Vous n'auriez pu lire ces lignes, ni jouer à Scrumble sans la participation appréciée de gens formidables et très joueurs! Ils ont cogité sans relâche, lancé un millier de fois les dés, pioché jusqu'à épuisement... Bref, un grand merci aux personnes suivantes :

Aux membres de l'équipe de Pyxis Suisse, Christian Lapointe, Francesco Lomonaco, Gaël Luisier, Nedjma Saidani, Tremeur Balbous et Thomas Gibot pour leurs encouragements et avoir servi de premiers cobayes!

À Pyxis Canada et à Pyxis Belgique pour leur soutien lors de la conception et du perfectionnement du jeu.

Aux nombreux joueurs qui nous ont fait parvenir leurs suggestions par e-mail, sur Twitter et lors d'événements, et tout particulièrement à David Preti et Thierry Durandy dont lesquelles sont désormais ajoutées au jeu.

Et naturellement, **à vous** parce que vous portez l'Agilité en jouant à Scrumble!

SOMMAIRE

NOTES ET REMERCIEMENTS	2
A PROPOS DU JEU	2
LICENCE ET DROITS D'UTILISATION	2
REMERCIEMENTS	2
SOMMAIRE	3
INTRODUCTION.....	4
OBJECTIFS DU JEU ET COMPORTEMENTS RECHERCHES	4
PREALABLES ET PREPARATION	5
SUPPORT ET ELEMENTS	5
PREPARATION ET ORGANISATION.....	5
VUE D'ENSEMBLE DU JEU	6
JOUEURS.....	7
SCRUM MASTER	7
PRODUCT OWNER.....	7
MEMBRES DE L'EQUIPE DE DEVELOPPEMENT	7
OBJECTIFS ET CONDITIONS DE VICTOIRE.....	7
OBJECTIFS	7
CONDITIONS DE VICTOIRE	8
TEMPS.....	8
PHASES Du JEU	8
PRE-SPRINT	8
SPRINT	10
PLANIFICATION DE SPRINT - SPRINT PLANNING.....	10
JOURS DE DEVELOPPEMENT	11
REVUE DE SPRINT - SPRINT REVIEW	12
RETROSPECTIVE DE SPRINT - SPRINT RETROSPECTIVE.....	12
VOTRE NEMESIS : LA DETTE TECHNIQUE.....	13
FIN DU JEU	13
ANNEXES.....	14
JEU DE ROLE : PRE-SPRINT	14
VALIDATION DES REPONSES	14
FOIRE AUX QUESTIONS.....	15
ANECDOTES DIVERSES.....	16

INTRODUCTION

Scrumble est un jeu qui est calqué sur les rôles, événements et artefacts de Scrum. Son objectif consiste à présenter le fonctionnement et les préceptes de Scrum à une équipe qui souhaite s'approprier ce cadre de développement ou à révéler les erreurs commises dans le cas d'une adoption défailante.

Le jeu permet, dans une ambiance décontractée, de mettre en lumière bon nombre de problèmes auxquels l'équipe fait face et d'y apporter, par la même occasion, des solutions spontanément.

Les nombreux aléas ont pour vocation de mettre à l'épreuve les joueurs en tant qu'acteurs créateurs de véritables produits. Perturber, créer des déclics, développer les valeurs individuelles et collectives nécessaires au bon fonctionnement d'une équipe, donc de projets, voilà le programme de Scrumble.

Nous vous souhaitons des parties de Scrumble très amusantes et nous espérons que vous tirerez profit des expériences vécues pour apporter les meilleures améliorations possibles dans vos équipes Scrum!

A vous de jouer!

OBJECTIFS DU JEU ET COMPORTEMENTS RECHERCHES

Comme il est indiqué dans l'introduction, Scrumble est un moyen d'apprentissage et de consolidation de Scrum par le plaisir lorsque les participants sont réticents à faire le pas d'eux-mêmes, à apprendre par les guides ou à s'approprier Scrum en tant que tel.

La grande force de Scrumble tient en ses révélations de dysfonctionnements dans l'équipe dans un contexte qui se doit favorable à la transparence et à l'interaction, sans risque pour chacun.

Les règles décrites dans ce guide sont idéalement calquées sur Scrum et peuvent paraître complexes au premier abord. Toutefois, il n'en est rien. Vous vous en rendrez compte très vite après un premier sprint. Aussi, aucune partie ne se ressemble, de par la créativité et les décisions des joueurs. Ils trouvent ensemble des possibilités nouvelles et parfois même des problèmes là où il n'en existe pas! De ces défis découlent de nombreux enseignements et un développement Agile pour tous :

- **Esprit d'équipe** : Chacun prend activement part à l'activité et est entendu par les autres.
- **Créativité** : Sans tricher, les mécanismes du jeu sont exploités par les joueurs.
- **Auto-organisation** : Les initiatives sont collectives, spontanées et rythment le jeu.
- **Transparence** : Chacun s'exprime clairement et les idées sont rendues compréhensibles à tous.
- **Respect** : Les tensions et perturbations inutiles sont évitées, écartées du jeu.
- **Résolution de conflits** : Les problèmes entre joueurs sont résolus intelligemment et sans heurt.
- **Capacité à prioriser** : Le Product Owner cible la livraison de valeur, et l'équipe y contribue.
- **Engagement** : Chacun fait de son mieux en suivant une direction commune vers l'objectif.

PREALABLES ET PREPARATION

SUPPORT ET ELEMENTS

Avant de commencer une partie de Scrumble, il vous faut bien sûr tous ses supports et éléments. Vous les trouverez joints à ce guide. Ils sont prêts à imprimer et à découper depuis le kit fourni. Des substituts peuvent très bien être utilisés à la place des pions et marqueurs. Par exemple, des briques et personnages de LEGO® sont faciles à intégrer.

Voici ce qu'il faut imprimer en couleur, découper, puis coller :

- **L'ensemble plateau, pions et marqueurs (Elements.png) :**
Imprimez le tout sur du papier Bristol blanc (ou sur du papier épais) en grand format (A3 minimum). Le format A2 est parfait pour un grand nombre de joueurs. Découpez ensuite les pièces en suivant les pointillés gris et en conservant une petite bordure blanche. Pliez symétriquement les pions représentant les joueurs et la dette technique et collez leur centre pour qu'ils puissent tenir en position verticale (pour faciliter les déplacements sur le plateau de jeu).
- **Les cartes à piocher (Cartes à piocher.pdf) :**
Imprimez sur du papier blanc ordinaire ou Bristol en format A4. En imprimant des copies recto verso, les cartes auront leur couverture au dos de leur énoncé. Si vous souhaitez économiser de l'encre, vous pouvez imprimer uniquement les pages impaires (ainsi vous n'aurez que les énoncés d'un côté). Découpez les en suivant les lignes pointillées grises à nouveau.
- **Les ensembles de User Stories (Stories.pdf) :**
Si vous souhaitez jouer une partie de Scrumble dans un contexte préétabli pour le produit à développer, imprimez uniquement l'une des trois premières pages.

Si vous souhaitez rédiger vos propres User Stories (scénarios utilisateurs), alors imprimez seulement la dernière page. Écrivez ensuite les scénarios sur les fiches obtenues (un scénario par fiche) avec l'aide du Product Owner.

Ces ensembles peuvent être imprimés sur du papier A4 ou A3. Le format A3 donne une meilleure visibilité néanmoins.

Les articles suivants sont aussi nécessaires : un dé à jouer (ayez en plusieurs à proximité en cas de perte ou bien fabriquez celui du kit), des blocs de notes Post-it® ainsi que des crayons à mine avec gomme à effacer au bout (de préférence).

Des accessoires tels qu'un tableau blanc et des feutres peuvent être utiles, mais ils ne sont pas indispensables.

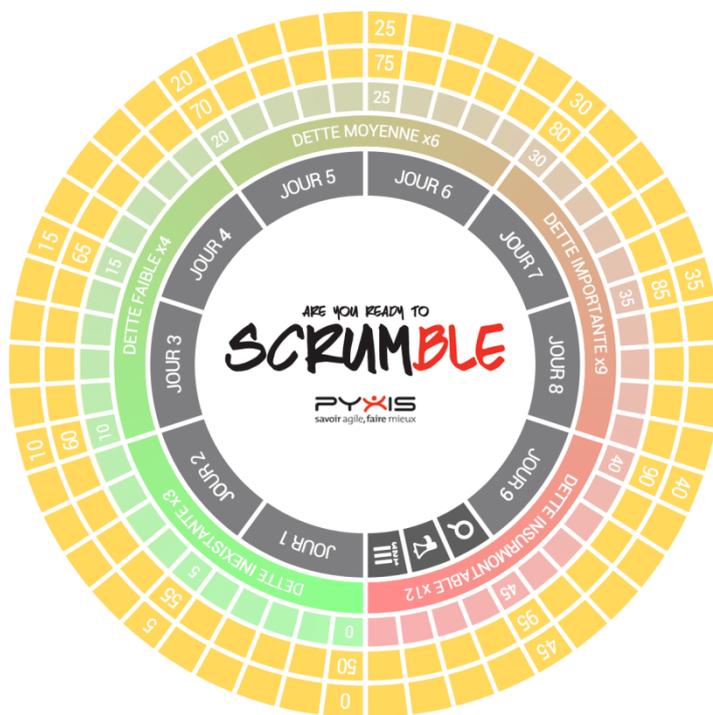
PREPARATION ET ORGANISATION

Vous avez maintenant tous les éléments du jeu sous la main. De plus, vous avez probablement parcouru ce guide, et ce, soit en diagonale ou entièrement. Bravo! Désormais, vous avez à déterminer une ou plusieurs équipes avec lesquelles jouer.

Sur le plan organisationnel, nous vous recommandons de suivre ces conseils pour réussir vos parties de Scrumble :

- Prévoyez deux bonnes heures de jeu que vous aurez planifiées avec l'équipe et un cadre agréable. Une salle est indispensable pour profiter du jeu sans être dérangés, ni déranger les autres.
- Limitez le nombre de joueurs (entre 5 et 11), en comptant le Product Owner et le Scrum Master. Communiquez au minimum l'introduction du jeu pour l'événement. Un rappel de lecture du guide Scrum (www.scrumguides.org) peut aussi être utile.
- Assurez-vous d'avoir un Scrum Master dans le groupe. Son expertise de Scrum doit être suffisante pour intervenir comme animateur du jeu.
- Ayez les éléments du jeu à disposition.
- Si le jeu est expérimenté dans le contexte d'un véritable projet, vous pouvez écrire vos propres User Stories, comme nous l'avons indiqué dans la section précédente. Dans ce cas, vous pouvez vous servir d'un Product Backlog déjà existant pour attaquer la phase de pré-sprint. Si le Product Backlog n'existe pas encore, nous vous invitons à mener un atelier de Big Wall¹ pour l'établir.

VUE D'ENSEMBLE DU JEU



Ci-contre, le plateau de jeu, avec de l'extérieur vers l'intérieur :

- Deux cercles pour les tâches;
- Le cercle de points de dette;
- Le cercle de facteur de dette;
- Le cercle du temps, donc des phases au cours d'un sprint. L'emplacement qui suit le jour 9 concerne la revue, la rétrospective et la planification du sprint à venir.

Le marqueur d'objectifs et le pion des joueurs :



Les différentes cartes à piocher :



Les marqueurs de dette et de temps :



¹ Big Wall : <http://www.mitchlacey.com/blog/the-big-wall-prioritizing-and-estimating-large-backlogs>

JOUEURS

Tout comme dans les règles définies par Scrum, Scrumble comporte trois catégories de joueurs : Scrum Master, Product Owner et membre de l'équipe de développement. Pour faire une partie, il faut un seul Scrum Master, un seul Product Owner et entre trois et neuf personnes pour agir à titre de membres de l'équipe de développement.

SCRUM MASTER

Le Scrum Master est le maître de jeu. Il ne participe pas comme joueur en tant que tel. Néanmoins, son rôle est primordial dans le bon déroulement du jeu en tant qu'**animateur de l'activité et facilitateur de décisions aux moments clés**.

L'animateur connaît suffisamment les règles détaillées dans ce guide et aussi celles de Scrum. Pour une expérience optimale, il lui faut l'attention de tous les joueurs. De plus, un jeu qui est imposé ne conduit qu'à l'ennui et à un abandon certain : l'animateur ne doit donc pas tomber dans le piège d'organiser un simple jeu, mais un véritable atelier ludifié.

Si le Scrum Master (le vrai, celui de l'équipe) ne connaît pas assez les règles du jeu ou s'il n'est pas apte à l'organiser, on l'invitera alors à jouer comme membre de l'équipe de développement et on désignera un autre animateur.

PRODUCT OWNER

Pas de surprise, comme son nom l'indique, le Product Owner est le responsable du produit. Il a un rôle particulier : celui de directeur centré sur la livraison de valeur et celui de communicateur connaissant le domaine pour lequel le produit est destiné.

Au cours du jeu, le Product Owner agit comme **décisionnaire et intermédiaire entre l'équipe de développement et les utilisateurs ou les ayant droits commanditant fictivement le projet**.

MEMBRES DE L'EQUIPE DE DEVELOPPEMENT

Ils sont des testeurs, analystes, développeurs débutants ou d'expérience, administrateurs, designers, stagiaires, etc. Ils sont tous égaux au sein de l'équipe (l'équipe réelle tout comme celle qui joue la partie). Ils s'auto-organisent et contribuent à la création et la livraison incrémentale du produit, cela dans toutes les phases techniques.

Le déroulement du jeu et la fin de la partie reposent donc principalement sur eux selon leurs qualités et interactions.

OBJECTIFS ET CONDITIONS DE VICTOIRE

OBJECTIFS

Bien que l'objectif premier soit une meilleure appropriation de Scrum, il est judicieux que l'animateur s'abstienne d'informer en premier lieu des tenants du jeu, c'est-à-dire des enseignements expressément définis. Les joueurs les découvriront rapidement par eux-mêmes et c'est là tout l'intérêt du jeu!

De plus, exprimer au préalable les leçons attendues peut facilement fausser les comportements des joueurs : les interactions souhaitées ne seront alors pas vécues et les défis seront du coup moins intéressants pour chacun des joueurs. **L'animateur ne mentionnera donc les avantages réels qu'à la fin du jeu, illustrés par les moments forts observés au cours de la partie.**

Enfin, pour diriger le jeu, le succès des joueurs est quantifié sous forme de **★ points de valeur**, que l'équipe devra acquérir en réalisant des **📌 tâches** sprint après sprint. Ce succès doit évidemment être le plus important possible à la fin du jeu, sous forme de **★ points de valeur** accumulés. Il représente la satisfaction du client à recevoir des incréments de produit, et s'obtient selon la dynamique détaillée plus loin, parsemée d'embûches ainsi que de défis individuels et collectifs.

CONDITIONS DE VICTOIRE

Il n'y pas de condition de victoire spécifique dans ce jeu. Vous avez bien lu!

En effet, la réussite des joueurs s'estime selon la complétion du produit en un nombre réduit de sprints. Le succès réel, comme vous l'aurez compris, réside en l'apprentissage réalisé au cours du jeu et dans son application future dans un contexte réel!

La seule possibilité d'échec, c'est la capitulation collective des joueurs sans leçon apprise. Nous vous en rassurons de suite : il s'agit d'une situation très improbable.

TEMPS

L'animateur choisit un nombre de sprints à réaliser selon le temps imparti pour jouer et le nombre de joueurs.

Il faut compter approximativement 15 minutes pour expliquer clairement les règles et 15 autres minutes pour que le Product Owner présente le produit à développer, ainsi que son contexte. Comptez 5 minutes pour la planification, 2 minutes par membre de l'équipe de développement par sprint et 10 minutes additionnelles par sprint pour la revue et la rétrospective. Il est envisageable de jouer par duos (comme la programmation en binôme!) afin de gagner du temps sans compromettre la qualité de l'expérience de jeu.

PHASES DU JEU

PRE-SPRINT

Une fois que l'animateur a expliqué Scrumble et donné les règles de jeu (voir les annexes pour le fil d'Ariane), le tout en quelques minutes, le Product Owner présente le produit que l'équipe devra produire au cours de la partie. Seul le Product Owner donne la vision et le Product Backlog, par un travail réalisé en amont et des User Stories qu'il ou elle peut avoir choisi dans les ensembles fournis.

Le Product Backlog, c'est l'ensemble des User Stories permettant d'alimenter les sprints.

Qu'il s'agisse d'un produit factice ou non, il est important que le thème et l'idée du produit plaisent à tous pour assurer leur participation.

Le Product Owner présente alors à l'équipe 15 User Stories (cas d'utilisations, fonctionnalités), écrites sur des notes Post-it® dans le format suivant :

Identifiant de User Story	Dépend d'un autre User Story
En tant que [QUI?], je veux [QUOI?].	
Valeur : [1 à 5 ★ points de valeur]	Taille : [Taille?]

Le Product Owner affecte alors 45 ★ points de valeur sur les User Stories, de 1 à 5 pour chaque, soit 3 en moyenne. Plus une User Story a de ★ points de valeur plus elle est intéressante et satisfaisante aux yeux des utilisateurs lors de sa livraison par les joueurs.

Vous remarquez aussi l'annotation **Taille**.

Une fois les User Stories présentées et comprises, c'est au tour des membres de l'équipe de développement d'estimer la complexité (taille) relative de chaque User Story à implémenter. À terme, cela donne un nombre total de 📌 tâches (objectifs) à atteindre lors du sprint.

De manière non exhaustive et en utilisant la relativité, les développeurs donnent ensuite leurs estimations en agencant les User Stories selon la table suivante.

Par complexité / Par priorité	1 – XS	2 – S	3 – M	4 – L	5 – XL
⬆ Haute					
⊛ Moyenne					
⬆ Basse					

N'hésitez pas à les afficher clairement sur un espace dédié de la zone de jeu ou sur une fiche d'accompagnement pour un meilleur repérage par la suite.

La complexité ou taille relative, façon t-shirt, permet de garder à l'esprit qu'une estimation exacte des efforts et de la charge de travail n'est jamais possible. Tout être humain est, par nature, mauvais dans le cadre d'une estimation absolue et, en somme, prenant plusieurs User Stories à réaliser par sprint, l'erreur s'atténue dans l'ensemble. Aussi, cela donnera par la suite matière à réfléchir pour ceux qui s'engagent (trop) souvent à poser des estimations en jour-personne et à les prendre pour argent comptant.

Voilà pour la petite histoire! Passons donc au premier sprint ainsi qu'à sa planification. A l'occasion, munissez-vous du marqueur de 📌 dette technique et placez-le de façon à obtenir 15 points de dette et un facteur 4.

SPRINT

Un sprint est composé, par ordre chronologique, d'une planification, de neuf jours de développement (neuf tours) ainsi que d'une revue et d'une rétrospective.

À noter : La non-journée de développement sur les dix jours ouvrables est consacrée à la fois à la planification, à la revue et à la rétrospective. On vous posera certainement la question, vous saurez donc désormais quoi répondre!

PLANIFICATION DE SPRINT - SPRINT PLANNING

Lors de cette phase, l'équipe choisit le nombre de 📌 tâches à réaliser au cours du sprint. Commencez par placer et faire s'orienter le marqueur Temps sur la case foncée du plateau, décorée d'une petite liste.

Pour ce faire, on laisse tout d'abord le Product Owner choisir les User Stories primordiales et à forte valeur pour le produit (focus utilisateur) et ayant d'éventuelles dépendances. Les développeurs peuvent le conseiller, mais c'est le Product Owner qui a le dernier mot.

Ces User Stories composent le Sprint Backlog. Un Objectif de Sprint (Sprint Goal) peut être déterminé par l'achèvement plus importantes. Lors du premier sprint, on recommande au Product Owner et à son équipe de prendre 3 User Stories maximum pour commencer.

Le but du sprint se décompose alors en l'atteinte de micro-objectifs qui récompensent l'équipe en autant de ★ points de valeur sur les User Stories, par la complétion de 📌 tâches.

Ce nombre de 📌 tâches est donné par la formule suivante selon la 🧨 dette technique (son impact est expliqué plus loin) et le 👤 nombre de développeurs (joueurs dans l'équipe de développement, excluant le Product Owner et le Scrum Master).

Le nombre de 📌 tâches à réaliser pour terminer une User Story est le produit de sa complexité, du facteur de 🧨 dette technique et du 👤 nombre de développeurs.

Par exemple, au début d'une partie où vous êtes cinq développeurs et prenez une User Story de taille M, son nombre de 📌 tâches à achever sera de 60.

Complexité d'User Story		Facteur de dette technique				
		1 - XS	2 - S	3 - M	4 - L	5 - XL
Inexistant 🧨 x 3		3 x 👤	6 x 👤	9 x 👤	12 x 👤	15 x 👤
Faible 🧨 x 4		4 x 👤	8 x 👤	12 x 👤	16 x 👤	20 x 👤
Moyenne 🧨 x 6		6 x 👤	12 x 👤	18 x 👤	24 x 👤	30 x 👤
Importante 🧨 x 9		9 x 👤	18 x 👤	27 x 👤	36 x 👤	45 x 👤
Insurmontable 🧨 x 12		12 x 👤	24 x 👤	36 x 👤	48 x 👤	60 x 👤

On écrit le nombre de 📄 **tâches** à réaliser, et donc à atteindre, sur la fiche de la User Story en question. Notez que ce quota peut être affecté par des bonus ou malus par la suite, ou si l'objectif n'est pas atteint au cours du sprint et que l'équipe y reprend son travail.

On place alors un marqueur coloré pour chaque objectif sur le cercle des 📄 **tâches** à l'emplacement du nombre à atteindre, ainsi qu'un pion Joueurs (celui représenté par trois personnages) de la même couleur à 0. Si l'objectif du User Story dépasse 100 📄 **tâches**, répartissez sur le cercle plusieurs marqueurs de même couleur, qu'il vous faudra ensuite tous récupérer. **La User Story sera donc terminée et livrable une fois le pion Joueurs ayant atteint tous les marqueurs.**

Passons au développement et attaquons la première journée!

JOURS DE DEVELOPPEMENT

C'est le moment de réaliser les différentes 📄 **tâches**. Chaque membre de l'équipe de développement (ne comprenant donc pas le Product Owner ni le Scrum Master) lance le dé à tour de rôle après avoir choisi soit de réaliser des 📄 **tâches** ou de réduire la 🧨 **dette technique**.

Dans le cas du choix de réduction de la 🧨 **dette technique**, on recule le marqueur sur le cercle en question (intérieur et bicolore). Pour ce qui est des 📄 **tâches**, on avance un marqueur en fonction du résultat du lancer de dé sur le cercle extérieur (en jaune pâle).

Ce choix est décidé par le joueur selon ce qui semble pouvoir aboutir à plus de valeur livrée à l'issue du sprint, voir au-delà. À vous d'évaluer!

Dans les deux cas, où tous les points obtenus aux lancers sont à utiliser:

- **Si le lancer donne un 1**, le joueur est autorisé à un second lancer. Si on obtient 1 à nouveau, on n'a pas droit à un troisième lancer. Le joueur peut changer de type de progression à l'occasion du second lancer.
- **Si le lancer donne un 6**, le joueur pioche alors une carte 🧨* **Problème**, la lit et présente le scénario à son équipe. Le Product Owner est aussi invité à réagir lors de cet événement.

Une carte 🧨* **Problème** est la plupart du temps un aléa au jeu. L'équipe doit prendre l'initiative d'agir ensemble face à celui-ci et non pas uniquement le lecteur de la carte.

Chaque nouveau jour, excepté le jour 1 ou lors de l'arrivée à la revue, l'un des joueurs pioche une carte 🗨️ **Daily.** Il s'agit d'un défi à présenter et à relever en équipe, d'une question à récompense ou d'une anecdote sur l'Agilité.

Lorsqu'un tour est terminé, on poursuit en passant au jour suivant, et cela jusqu'à ce que l'on arrive à la revue.

Dans le cas où toutes les 📄 **tâches** à réaliser au cours du sprint sont terminées avant la fin du jour 9, l'équipe choisit une option parmi les suivantes :

- **Réalimenter le Sprint Backlog courant** en ajoutant une ou plusieurs User Stories afin de livrer plus à la fin du sprint. Dans ce cas, on ajoute pions Joueurs et marqueurs et on interrompt le tour (jour) en cours dû à l'effort fourni et on passe au jour suivant.

- **Payer de la 📌 dette technique** à défaut d'avancer sur des 📌 tâches pendant le reste du sprint. Et oui, après tout, pourquoi pas?
- **Terminer le sprint**, cette option est possible uniquement s'il n'y a plus aucune User Story dans le Product Backlog afin de dresser le bilan de la partie en passant directement à la toute dernière revue.

REVUE DE SPRINT - SPRINT REVIEW

Lors de l'arrivée à la case Revue (la première des trois cases foncées), le Product Owner à la tâche d'inspecter ce qui est réalisé en termes de points et d'événements. Il peut ainsi dresser un état des lieux du produit actuel, féliciter, encourager et donner des conseils constructifs à l'équipe. Cette phase ne doit pas dépasser 5 minutes.

Il pioche alors autant de cartes 🗉 Revue que le nombre de User Stories complètement terminés pendant ce sprint. Les cartes sont alors lues une à une alors que leur consigne affecte le jeu maintenant ou plus tard.

Aussi, pour toute User Story non terminée, l'équipe observe un malus pour avoir réalisé des activités qui sont à revoir et à refaire plus tard, tant sur le plan technique que fonctionnel. Ce malus donne autant de points de 📌 dette technique que de tranches de 5 📌 tâches non achevées au cours du sprint. Les User Stories dont les marqueurs sont déjà sur le plateau ne sont pas affectées, leur nombre de 📌 tâches ne changeant pas. Le handicap sera appliqué uniquement aux nouvelles User Stories en cours de développement et à l'effort nécessaire pour réduire cette 📌 dette technique plus tard.

Enfin, pour la dernière partie de la revue : l'animateur fait le décompte et le cumul des points atteints depuis le début du jeu. Il est celui qui accrédite l'équipe des ★ points de valeur des User Stories terminées. Bravo!

Après la revue, c'est le moment de passer à la rétrospective.

RETROSPECTIVE DE SPRINT - SPRINT RETROSPECTIVE

Ce moment particulier est entièrement dédié aux participants. Au moment où le marqueur Temps atteint la seconde case gris foncée, l'animateur donne à chacun des joueurs entre une et deux minutes pour exprimer ce qu'il a aimé, moins aimé et détesté au cours du sprint et ce qu'ils souhaitent mettre en place au cours du prochain sprint. Les initiatives sont fortement encouragées.

Lors de cette phase, la dynamique émergente par les joueurs via leurs valeurs et pratiques doit corrélérer de plus en plus avec les principes Agiles au fil des sprints. Ce moment est censé être le plus formateur et révélateur dans le jeu. La dernière rétrospective peut bien sûr s'avérer plus longue comme fin de jeu.

N'hésitez pas à vous référer au manifeste Agile pour cette phase du jeu: www.agilemanifesto.org/iso/fr.

Le sprint est désormais terminé, félicitations! Vous pouvez désormais démarrer le suivant par un nouveau planning.

VOTRE NEMESIS : LA DETTE TECHNIQUE

Il s'agit de votre pire ennemie. Elle prend plusieurs formes... Elle est partout...

Le cauchemar de nombreuses équipes Scrum, c'est la **👛 dette technique**. Elle représente, sous une forme explicite ou non, un dû à l'égard des utilisateurs et ayant droits lorsque des User Stories ou **📄 tâches** ne respectent pas la définition de "Terminé".

La **👛 dette technique** perturbe donc la partie à différents moments, notamment lors du non-achèvement du travail à la fin d'un sprint. Elle génère plus de difficultés lorsque ses points s'accumulent et atteignent les différents paliers, soit à chaque dizaine de points.

Au début de la partie, le facteur par défaut est de x4 (handicap nul), avec le marqueur Dette placé à 15 points. Il évolue ensuite selon les aléas rencontrés et les décisions prises par les joueurs.

Voici un récapitulatif de la **👛 dette technique** :

Points de dette	0 à 9 👛	10 à 19 👛	20 à 29 👛	30 à 39 👛	40 à 49 👛
Facteur de dette	x 3	x 4	x 6	x 9	x 12
Handicap	- 25 %	0 %	+ 50 %	+ 125 %	+ 200 %

La limite maximale de la **👛 dette technique** est 49 points de dette. À ce stade, la situation sera déjà invivable. Nous espérons que vous saurez l'éviter!

FIN DU JEU

Le jeu se termine après la dernière rétrospective, dès que l'équipe est réellement bloquée ou qu'elle en a assez (une situation très rare, mais si cela devait arriver, faites une réflexion collective sur les obstacles rencontrés, facilitée par le Scrum Master).

L'animateur résume les grandes lignes de la partie et révèle les véritables objectifs du jeu. Le décompte des **★ points de valeur** est fait surtout pour le plaisir. Les participants doivent repartir avec le sourire et aussi avec des idées motivantes en tête. Le score n'est pas si important, ce qui compte réellement c'est l'apprentissage, après ces quelques heures de jeu avant d'attaquer de nouveaux projets ou ceux en cours.

Avez-vous réalisé un travail exceptionnel? La **👛 dette technique** a-t-elle été maîtrisée et vaincue? Quelles sont les forces et les faiblesses de votre équipe? Ce jeu était-il vraiment un jeu?... Tant de questions auxquelles nous espérons voir vos collègues et vous répondre!

Toutes nos félicitations, vous avez gagné!

ANNEXES

JEU DE ROLE : PRE-SPRINT

Si vous n'avez pas l'âme d'un rôliste ou si vous êtes en manque d'inspiration, voici une introduction à utiliser lors du pré-sprint. Le reste viendra ensuite naturellement, soyez-en sûr!

Bonjour à tous,

Comme vous le savez certainement, **[Entreprise cliente]** nous a contacté hier pour développer un super produit qui fera **[un truc vraiment chouette]** et qui est totalement nouveau dans son domaine. Les clients et ayant-droits de **[Entreprise cliente]** se sont rapprochés de **[Product Owner]**, qui les a convaincu de vos compétences associées à une démarche Agile, permettant de leur livrer progressivement et fréquemment des fonctionnalités formidables.

Je suis **[Animateur]**. En tant que Scrum Master, je vous assisterai dans cette aventure. Car non, **[Développeurs]**, livrer des fonctionnalités pleinement opérationnelles en deux semaines, ce n'est pas du pipeau! Et avoir du plaisir à développer, oui, c'est possible!

Nous avons donc quelques heures pour voir cela tous ensemble. J'espère que vous vous prêterez au jeu et aimerez y jouer. Je laisse désormais **[Product Owner]** présenter plus en détail le produit que nous développerons, puis nous commencerons à relever ce défi.

Bonne partie à tous!

VALIDATION DES REponses

Il se peut que vous ayez à vérifier les réponses à certaines questions des cartes  **Daily**. En cas de doute, les voici, triées par ordre alphabétique.

	Combien de rôles y a-t-il dans une équipe Scrum?	Trois: il s'agit du Product Owner, des Développeurs et du Scrum Master.
	Combien d'heures au maximum dure la Rétrospective d'un Sprint mensuel?	Trois.
	Combien d'heures au maximum dure la Revue d'un Sprint de deux semaines?	Deux.
	Combien d'heures au maximum dure le Planning d'un Sprint mensuel?	Huit.
	Combien de semaines au maximum dure un Sprint selon Scrum?	Quatre.
	Quelles sont les trois questions usuelles servant l'équipe de développement au Daily Scrum?	Quel progrès depuis le dernier Daily Scrum ou la veille? Que va faire chacun aujourd'hui? Quels sont les obstacles à adresser?
	Qui doit être obligatoirement présent lors des Daily Scrums?	L'équipe de développement, soit les développeurs. C'est le strict minimum!

FOIRE AUX QUESTIONS

Voici quelques renseignements utiles qui n'ont pas été abordés jusqu'ici. Ils sont l'objet d'extensions ou particularités qui peuvent survenir en cours de partie.

- **Un ou plusieurs joueurs doivent ou veulent partir. Cela signifie-t-il la fin de la partie?**

Tout joueur peut quitter le jeu, bien que cela soit regrettable. Dans le contexte d'un projet réel, cela peut tout aussi bien arriver et mettre en difficulté une équipe Scrum en plein sprint. Si cette situation se produit, le sprint en cours continue en suivant les règles prescrites, et le joueur sortant peut faire don de ses bonus et malus permanents. Au sprint suivant, veillez à bien reconsidérer le nombre de joueurs encore en lice!

- **Pour un vrai produit, le Product Backlog évolue... Dommage, on ne vit pas cela dans Scrumble.**

Il est vrai que cette situation est particulièrement intéressante, mais la transposer dans le jeu créerait de la confusion et nécessiterait plus de temps de la part des joueurs. Si vous souhaitez expérimenter, en collaboration avec le Product Owner, ne dévoilez pas tout le Product Backlog aux développeurs dès le premier sprint! Remplacez y ensuite une ou deux User Stories juste avant le troisième sprint. Restez raisonnables et partagez-nous les réactions des joueurs!

- **Scrumble manque de compétitivité. Comment y remédier?**

Bien que l'objectif premier de Scrumble soit de développer l'esprit, les valeurs et les forces de l'équipe au cours du jeu, il est possible d'y ajouter un soupçon de compétition. Si vous avez beaucoup de compétiteurs dans votre équipe ou tout simplement beaucoup de joueurs, vous pouvez envisager de jouer plusieurs parties en simultané avec des équipes de taille égale et de déclarer vainqueur celle qui arrive au plus tôt, en termes de sprints et jours, à terminer le Product Backlog (choisissez alors un même ensemble de User Stories, bien évidemment). Le Product Owner et l'animateur peuvent alors assister les équipes en différentes phases, en restant organisés.

- **Deux équipes de tailles différentes ont-elles les mêmes chances de réussir et de progresser au cours de la partie?**

Tout à fait, car la difficulté du jeu, malgré les divers aléas, n'est aucunement fonction de la taille de l'équipe Scrum.

Les difficultés rencontrées lors d'une partie ne relèvent pas du nombre de joueurs, mais des décisions qu'ils prennent. Une équipe de cinq joueurs peut donc achever un même carnet de produit en autant de sprints et de jours qu'une équipe de dix joueurs si les deux équipes prennent des décisions semblables.

ANECDOTES DIVERSES

Ce jeu a été développé dans un cadre... pas Scrum! En fait, c'est Personal Kanban qui a été utilisé.

L'idée de Scrumble, bien que certains concepts soient semblables ici et là à d'autres jeux sérieux, a germé lorsque son créateur suivait un cours sur la ludification. [Gamification](#), par Kevin Werbach, est un cours gratuit qu'il donne fréquemment. Nous le recommandons à tous ceux qui sont curieux et joueurs.

Souhaitez-vous savoir quelle est l'avancée moyenne d'un développeur par jour au cours du jeu? Si aucun bonus-malus n'est appliqué, elle est de 4,08. Cela vous sera utile pour vos prévisions!

Quelle est la moyenne de dette cumulée par joueur et par sprint aussi? Eh bien, malheureusement, elle est trop complexe à calculer! Retenez juste que vous en accumulerez le moins en faisant simplement preuve de bon sens.

SCRUMBLE VOUS EST OFFERT PAR



/campus

Découvrez l'Agilité et les pratiques d'ingénierie logicielle



/conseil

L'Agilité en termes de diagnostics, stratégie, prise en charge et coaching



/studio

Demandez des solutions logicielles sur mesure à nos développeurs experts