

Absolut**et.**



Scrum 101

Les bases du cadre de travail Scrum

Agenda

01

Approche traditionnelle vs
l'agilité

02

Qu'est-ce que le Scrum?

03

Empirisme

04

Développement itératif et
incrémental

05

Valeurs

06

Cadre de travail

07

Équipe et rôles

08

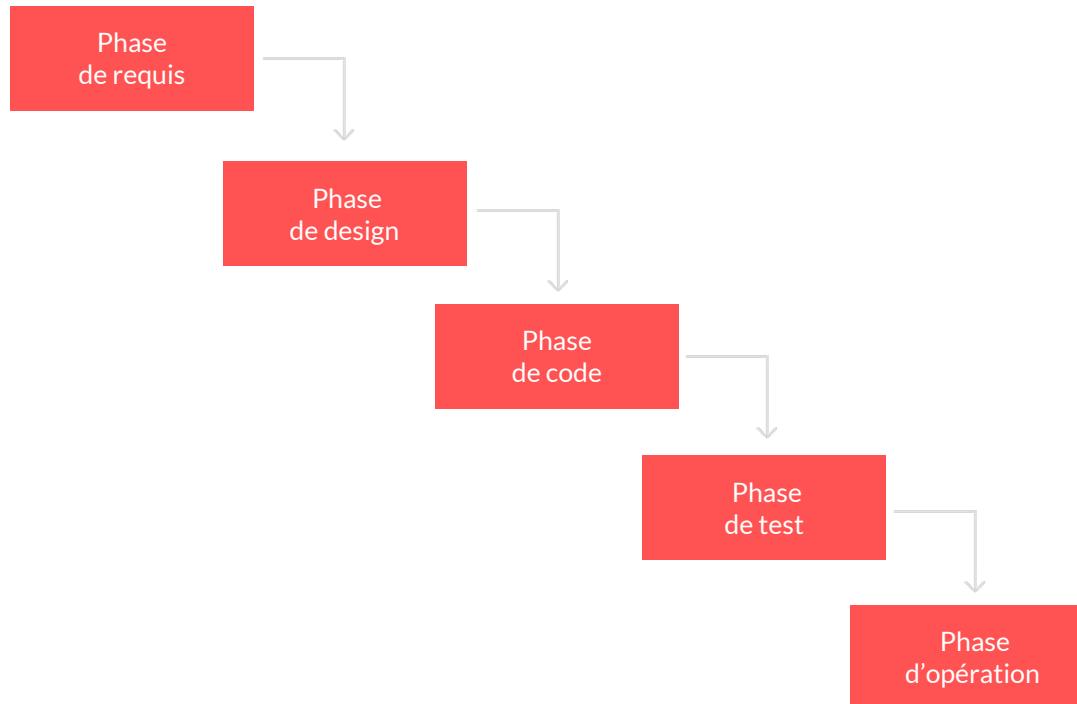
Événements

09

Concepts agiles importants

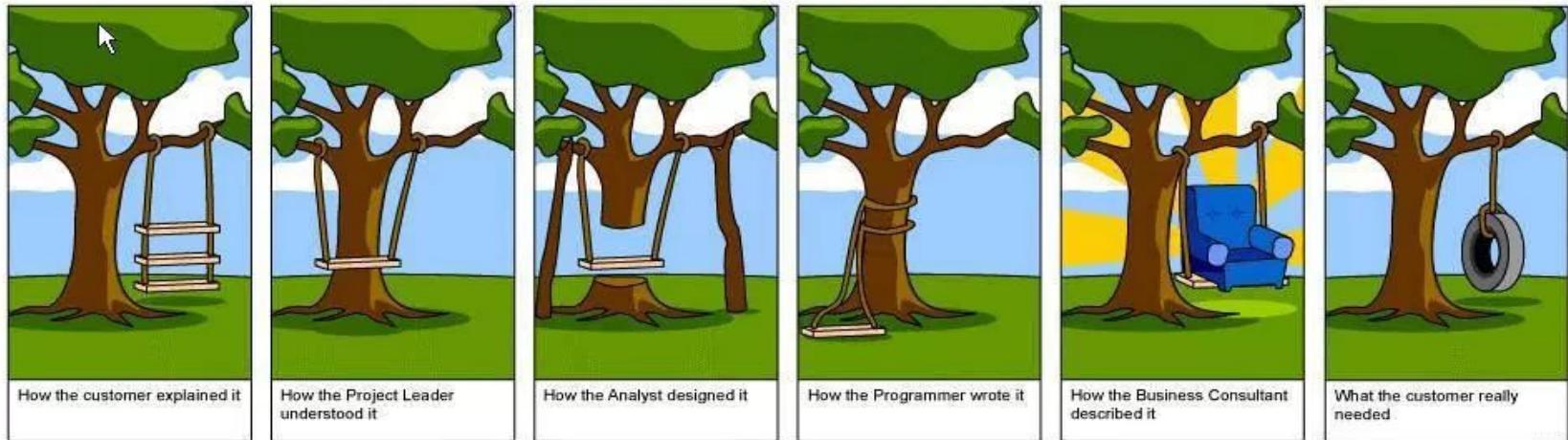
A.

Approche traditionnelle - Waterfall



A.

Pourquoi le “waterfall” est souvent un échec



3 Vérités que nous devons accepter

- 1
- 2
- 3

Il est impossible de rassembler tous les requis en début de projet

Quelles que soient les requis que vous rassemblez, ils sont garantis de changer

Il y aura toujours plus à faire que le temps et le budget le permettront

3 Vérités que nous devons accepter

- 1
- 2
- 3

Il est impossible de rassembler tous les requis en début de projet

Quelles que soient les requis que vous rassemblez, ils sont garantis de changer

Il y aura toujours plus à faire que le temps et le budget le permettront

3 Vérités que nous devons accepter

- 1
- 2
- 3

Il est impossible de rassembler tous les requis en début de projet

Quelles que soient les requis que vous rassemblez, ils sont garantis de changer

Il y aura toujours plus à faire que le temps et le budget le permettront

Les 4 valeurs du Manifeste Agile

En 2001, en se basant sur leur expérience, les auteurs du Manifeste Agile ont stipulé:

“Nous reconnaissons la valeur des éléments de droite, mais privilégions ceux de gauche.”

Les individus et leurs interactions

PLUS QUE

les processus et les outils

Des logiciels opérationnels

PLUS QUE

la documentation exhaustive

La collaboration avec les clients

PLUS QUE

la négociation contractuelle

L'adaptation au changement

PLUS QUE

le suivi d'un plan

A.

Qu'est-ce que le Scrum?

Scrum est un cadre de travail (framework) pour le développement, la livraison et la maintenance de produits complexes. Au début des années 1990, Ken Schwaber et Jeff Sutherland ont développé Scrum. En 2010, ils ont écrit le Guide Scrum.

Le cadre de travail consiste:

- 3 Rôles
- 4 Événements
- 3 Artefacts
- Règles

Scrum est:

- Léger
- Simple à comprendre
- Difficile à maîtriser



Empirisme

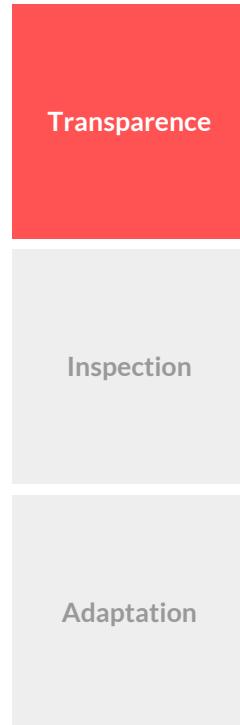
Scrum est fondé sur la théorie du contrôle empirique des processus.

L'empirisme signifie travailler de manière factuelle, fondée sur l'expérience et fondée sur des preuves.

En Scrum, le progrès est basé sur des observations de la réalité et non sur des plans fictifs.



Les 3 piliers de l'empirisme.



Cela signifie :
présenter les faits tels quels.

La transparence exige que toutes les personnes impliquées aient le même langage et la même compréhension de la réalité.

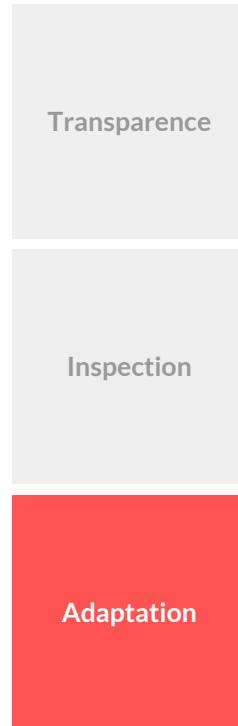
Les 3 piliers de l'empirisme.



L'inspection dans ce contexte n'est pas une inspection faite par un inspecteur / auditeur mais une inspection par l'équipe Scrum et par les parties prenantes.

Dans Scrum, nous devons fréquemment inspecter nos plans, notre façon de travailler et nos livrables.

Les 3 piliers de l'empirisme.



L'adaptation répond à un besoin d'amélioration continue.

L'adaptation est faite en fonction des résultats de l'inspection.

Tout le monde doit se poser cette question régulièrement : Sommes-nous meilleur qu'hier?

Scrum a quatre événements supportant l'inspection et l'adaptation.

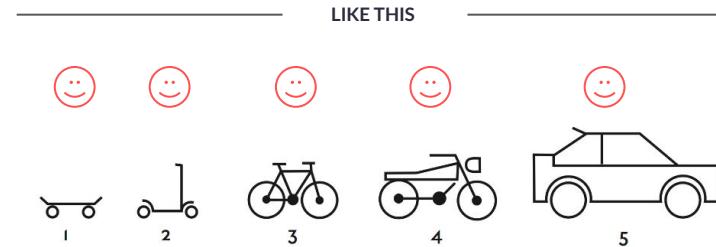
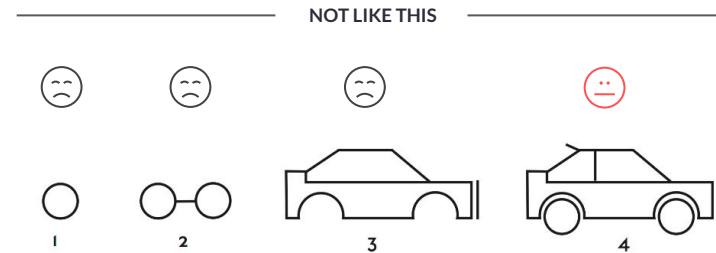
(Planification, Mêlée quotidienne, Revue et Rétrospective)

**Scrum est
itératif + incrémental**

Développe- ment Itératif

L'idée de base derrière cette méthode est de développer un système à travers des cycles répétés (itérations).

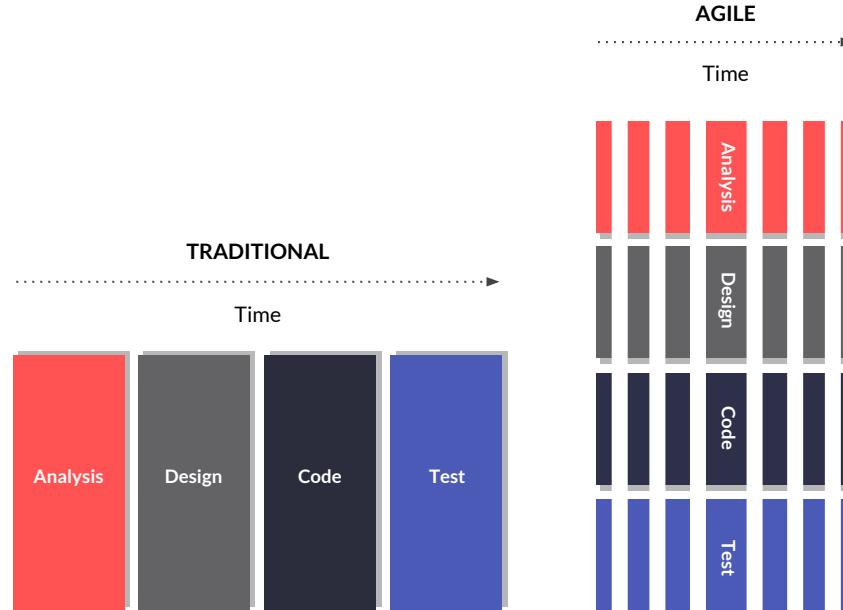
Permettre de tirer parti de ce qui a été appris pendant le développement des itérations précédentes.



Développe- ment Incrémental

Cette méthode découpe la fonctionnalité du système en incréments.

Dans chaque incrément, une tranche de fonctionnalité est livrée des requis jusqu'au déploiement.



**Le succès du Scrum dépend de
l'adoption de ses 5 valeurs.**

Les 5 Valeurs Scrum

Engagement

S'engager à atteindre les objectifs de l'équipe Scrum.

Courage

Avoir le courage de faire la bonne chose et de travailler sur des problèmes difficiles.

Focus

Focaliser sur le travail du Sprint + les objectifs de l'équipe.

Ouverture

Être ouvert par rapport au travail et ses défis.

Respect

Se respecter les uns les autres en considérant les autres comme des gens compétents et indépendants.

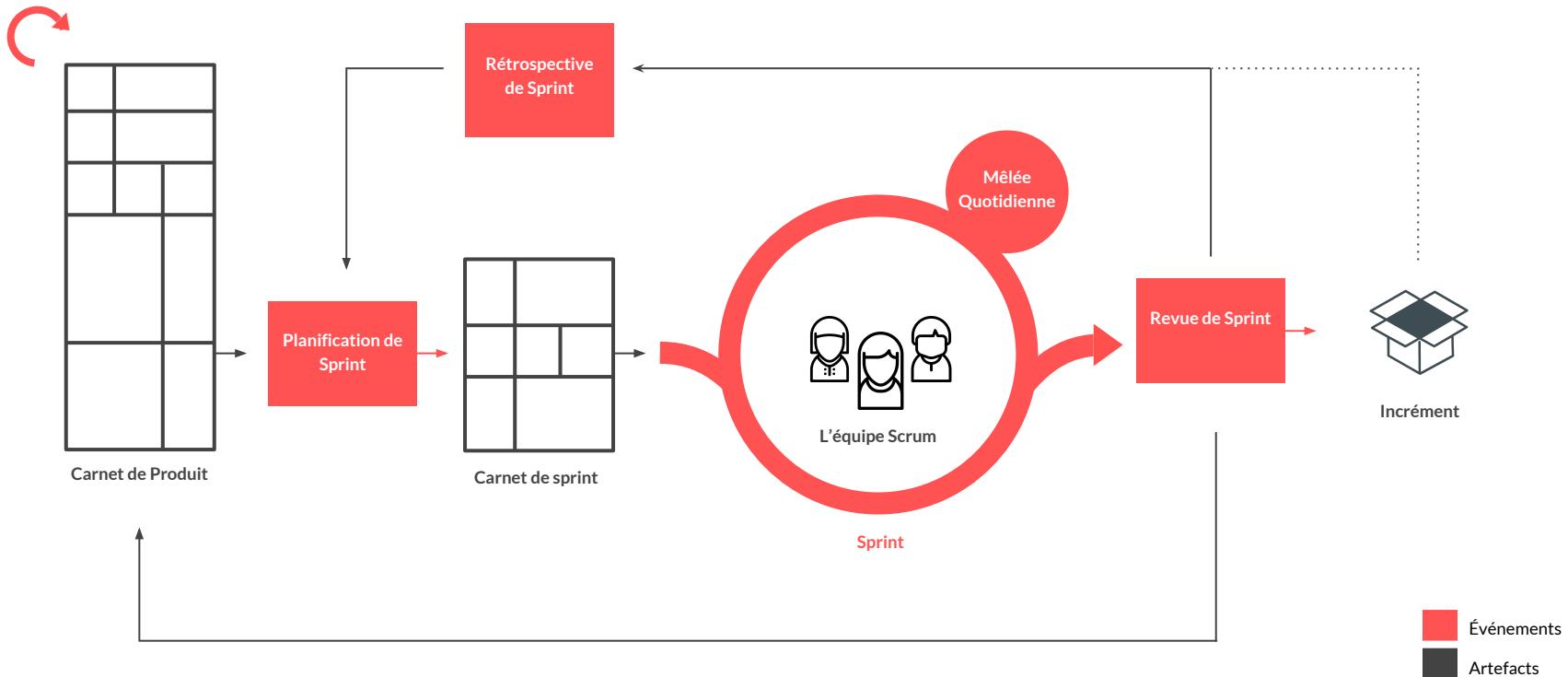
A.

Le Cadre de Travail

Rôles + Événements + Artefacts + Règles

A.

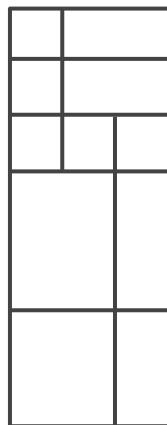
Le Cadre de Travail Scrum



Les Artefacts

CARNET DE PRODUIT

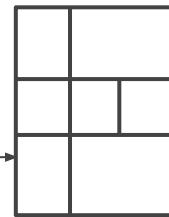
Liste ordonnée de tout ce qui est connu pour être nécessaire dans le produit. Cette liste dynamique est constamment adaptée par le **Chef de Produit**.



Carnet de Produit

CARNET DE SPRINT

Comprend les éléments du **Carnet de Produit** sélectionnés par l'équipe de développement pour le Sprint et le plan pour les livrer.



Carnet de sprint

INCRÉMENT

L'incrément de produit est l'ensemble des éléments terminés pendant ce Sprint, et aussi ceux terminés pendant les Sprints précédents.

Les éléments sont terminés lorsqu'ils remplissent la **Définition de Terminés**.

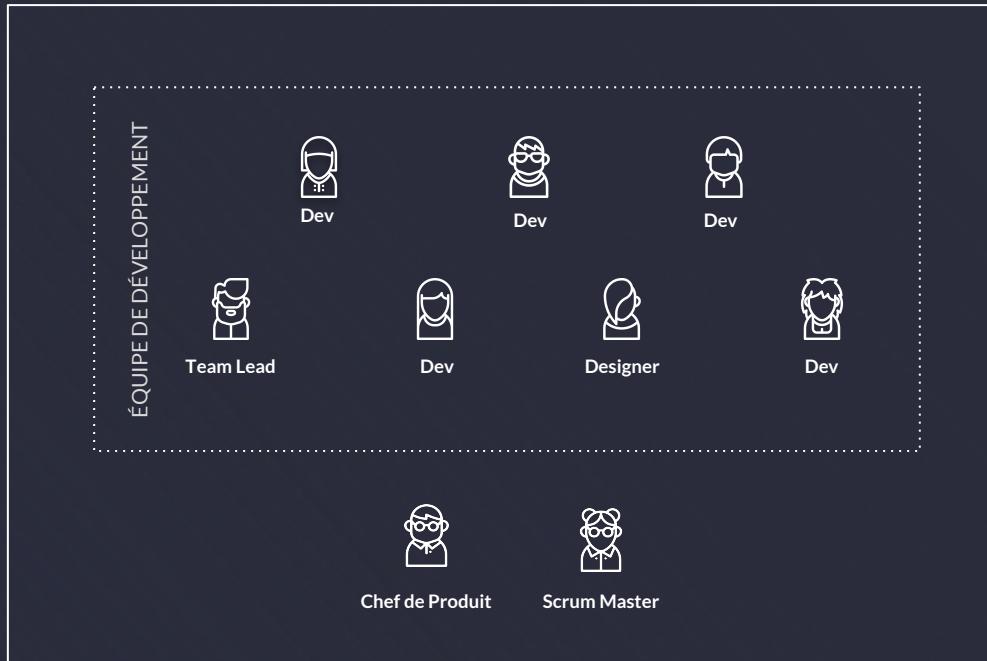
Cet incrément doit être utilisable, qu'il soit déployé en production ou non.



Incrément

A.

L'équipe Scrum

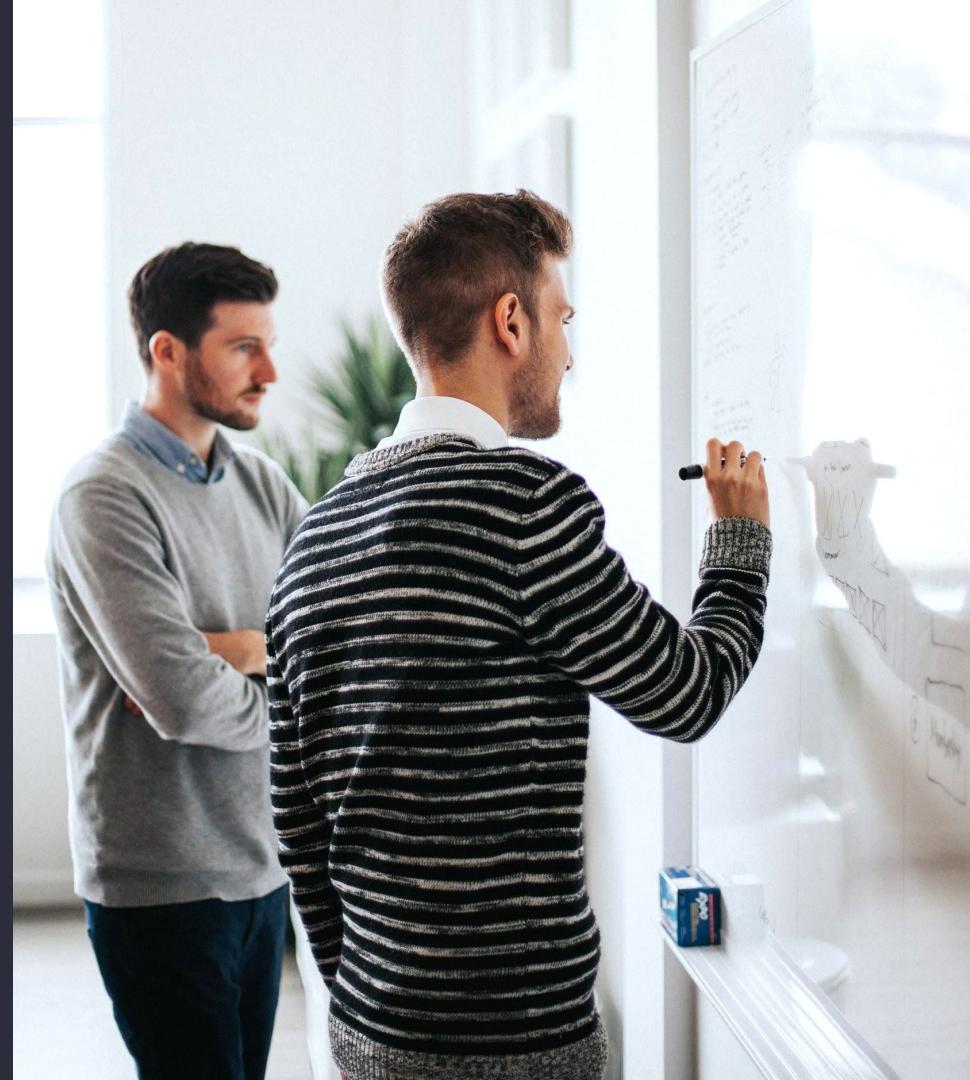


A.

RÔLE SCRUM

Chef de Produit

- “Product Owner”
- Apporte la vision et la stratégie
- Comprend les utilisateurs et les clients
- Conseille le client.
- Crée + priorise le carnet de produit
(valeur + risque).
- Présente les éléments du carnet de produit à l'équipe.



**Le Chef de Produit *est la seule personne*
responsable de la gestion du carnet de produit.**

RÔLE SCRUM

Équipe de Développement

- Équipe de 3 à 9 personnes
- Multidisciplinaire (FE, BE, UX, Designers, etc ...)
- Autonome
- Auto-organisée
 - atténue les risques et trouve ses propres solutions
 - sait ce qu'il faut faire
 - est conscient de sa vitesse/performance et de sa capacité
 - communique ses actions
- La responsabilité est partagée et appartient à toute l'équipe.
- Propriétaire du Carnet de Sprint



Pendant un sprint,
tout changement au Carnet de Sprint
doit être discuté et approuvé
par l'équipe de développement.

RÔLE SCRUM

Scrum Master

- Coach
- Formateur
- “Servant-leader”
- Gardien du cadre de travail Scrum
- S’assure que Scrum est compris et adopté
- Facilite les événements Scrum
- S’assure que les obstacles soient adressés



**Le Scrum Master n'est pas un
gestionnaire de projet
et ne doit pas décider qui travaille
sur quoi et quand.**

L'équipe de développement s'auto-organise.

A.

Les Événements Scrum

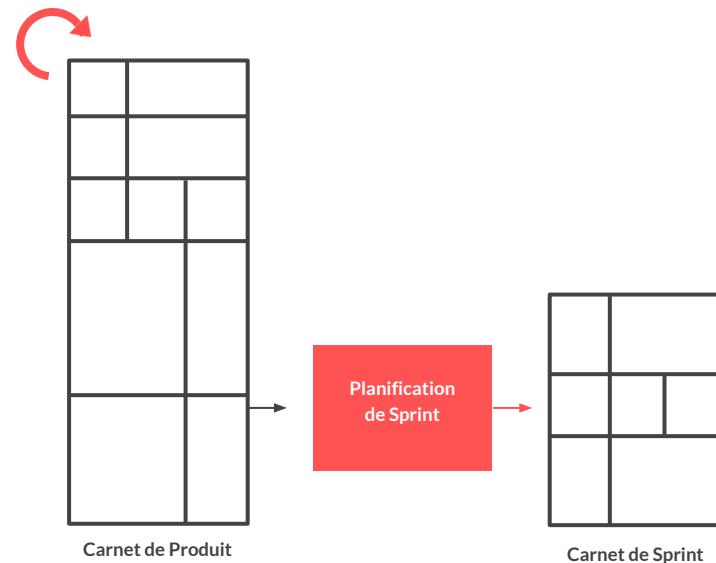
Planification + Mêlées + Revue + Rétrospective

ÉVÉNEMENT SCRUM

Planification

La planification répond aux questions suivantes:

- **Sujet 1:** Que peut-on faire dans le Sprint?
- **Sujet 2:** Comment cela sera-t-il fait?



ÉVÉNEMENT SCRUM

Planification

Sujet 1: Que peut-on faire dans le Sprint?

- **L'équipe Scrum discute:**
 - de l'objectif du sprint
 - des éléments souhaité de **Carnet de Produit** à compléter pour atteindre l'objectif de Sprint.
- **Intrants de la rencontre:**
 - Carnet de Produit
 - Données de l'équipe de développement:
 - Performances passées (vitesse)
 - Capacité projetée
 - Dernier Increment de Produit (Fonctionnalités "Terminées")
- **L'ensemble de l'équipe Scrum définit un objectif de sprint.**

ÉVÉNEMENT SCRUM

Planification

Sujet 2: Comment cela sera-t-il fait?

- L'équipe de développement:
 - décide comment elle va développer les fonctionnalités dans un incrément "Terminé" pendant le Sprint (tâches, etc ...)
 - détermine s'il y a trop ou pas assez de travail et peut renégocier les éléments sélectionnés dans le **Carnet de Produit**.
- Le **Carnet de Sprint** est créé au cours de la Planification de sprint.

**Le nombre d'éléments sélectionnés
pour le Carnet de Sprint
relève uniquement de l'équipe de Dev.**

**Seule l'équipe de Dev peut évaluer ce qu'elle peut
accomplir au cours du prochain sprint.**

ÉVÉNEMENT SCRUM

Mêlée quotidienne

- Événement quotidien de 15 minutes
- L'équipe de dev. planifie les prochaines 24 heures.
- Inspection du progrès par rapport à l'objectif de sprint et l'avancement du travail prévu dans le **Carnet de sprint**
- Certaines équipes de dev utilisent ce format de questions:
 - Qu'ai-je fait hier?
 - Que vais-je faire aujourd'hui?
 - Est-ce que j'ai des bloquants?

La mêlée quotidienne n'est pas une réunion de statut tenu pour le Chef de Produit et le Scrum Master.

C'est une session de planification collaborative quotidienne pour l'équipe de dev.

Si d'autres invités sont présents, ils ne devraient pas perturber cette précieuse session.

ÉVÉNEMENT SCRUM

Revue de Sprint

- Le **Chef de Produit** explique aux parties prenantes (clients) quels éléments ont été «Terminé» et lesquels n'ont pas été «Terminé».
- L'équipe de dev. discute de:
 - ce qui s'est bien passé pendant le Sprint
 - quels problèmes ont été rencontrés
 - comment ces problèmes ont été résolus
- L'équipe de dev. fait la démonstration du travail accompli et répond aux questions sur l'incrément.
- Le **Chef de Produit** discute du **Carnet de Produit** et de l'échéancier.
- Les potentiels éléments du carnet de produits planifiés pour le prochain sprint sont identifiés.

A.

**La revue de sprint
n'est pas seulement une démo.**

Définition de “Terminé”

- Le but de chaque Sprint est de fournir des Incréments de fonctionnalités potentiellement livrables qui adhèrent à la définition actuelle de "Terminé".
- Tout le monde doit comprendre ce que «Terminé» signifie pour assurer la transparence.
- Elle guide l'équipe de développement.
- Au fur et à mesure que les équipes Scrum prennent de la maturité, on s'attend à ce que leurs définitions de «Terminé» s'étendent pour inclure des critères plus stricts pour atteindre un niveau de qualité supérieur.

ÉVÉNEMENT SCRUM

Rétrospective de sprint

But :

- Inspecter le déroulement du dernier Sprint en ce qui concerne les personnes, les relations, les processus et les outils
- Identifier les principaux éléments qui se sont bien passés et les améliorations possibles
- Créer un plan pour mettre en œuvre des améliorations.

L'équipe Scrum prévoit des moyens d'améliorer la qualité des produits en améliorant les processus de travail ou en adaptant la définition de «Terminé»

**La rétrospective doit demeurée
respectueuse, constructive et efficace.**

Limite de temps des événements

(peut bien sûr prendre moins de temps)

Event	4 semaines	3 semaines	2 semaines	1 semaine
Planification de Sprint	8 hr	6 hr	4 hr	2 hr
Mêlée quotidienne	15 min quotidien	15 min quotidien	15 min quotidien	15 min quotidien
Revue de Sprint	4 hr	3 hr	2 hr	1 hr
Rétrospective de Sprint	3 hr	2.25 hr	1.5 hr	.75 hr

Sessions de raffinement

- Les items du **Carnet de Produit** sont examinés et révisés par l'équipe Scrum.
- Ces sessions comportent :
 - Ajout de détails
 - Estimation
 - Ordonnancement des items du carnet de produit
 - Séparation/fusion des items
 - Suppression des items non-pertinents
- L'équipe Scrum décide comment et quand la session de raffinement est tenue. Ces sessions ne devraient pas consommer plus de 10% de la capacité de l'équipe de dev.

L'équipe de dev est responsable de toutes les estimations.

**Le Chef de Produit peut influencer l'équipe de dev
en l'aidant dans sa compréhension
et identifier des compromis,**

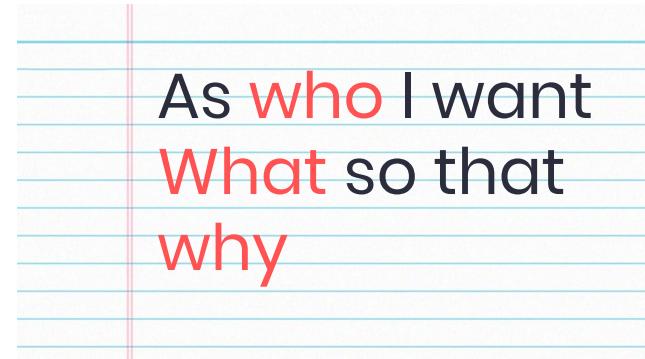
**mais les personnes qui effectueront le travail
ont le mot final sur les estimations.**

Autres concepts importants en agilité

- Récits utilisateurs
- Qualités d'un bon récit utilisateur
- Points d'histoires
- Vélocité
- MVP

Récit utilisateur

- Les récits utilisateurs sont des descriptions courtes et simples d'une fonctionnalité donnée du point de vue de la personne qui désire la nouvelle fonctionnalité, généralement un utilisateur ou un client.
- Ils suivent généralement un modèle simple:
En tant que persona, je veux action pour que je puisse répondre à un besoin.
- Ce format a pour but de changer le focus sur l'écriture de requis vers une discussion autour de ceux-ci.



Qualités d'un bon récit utilisateur

Independant

Autonome, aucune dépendance inhérente à un autre élément du cahier de charge.

Negotiable

Ne doit pas être un contrat explicite, devrait laisser place à la discussion.

Valuable

Doit apporter de la valeur aux parties prenantes.

Estimable

L'équipe doit toujours être capable d'en estimer la taille.

Small

Ne devrait pas être trop grand pour devenir impossible à planifier/découper/prioriser.

Testable

Doit fournir les informations nécessaires pour rendre le développement de test possible.

Story points

Les "Story Points" sont utilisés pour estimer l'effort global qui sera nécessaire pour "Terminer" un histoire.

Cet estimation comprend :

- Quantité de travail
- Complexité
- Risque et incertitude

La méthode du "Planning Poker" est souvent utilisée pour estimer le nombre de story points des récits.

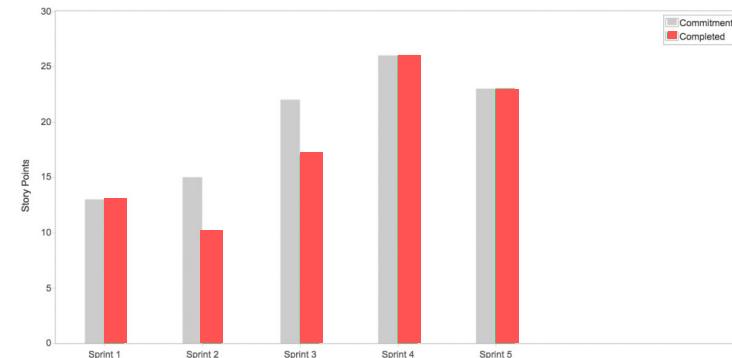


Vélocité

(pts/sprint)

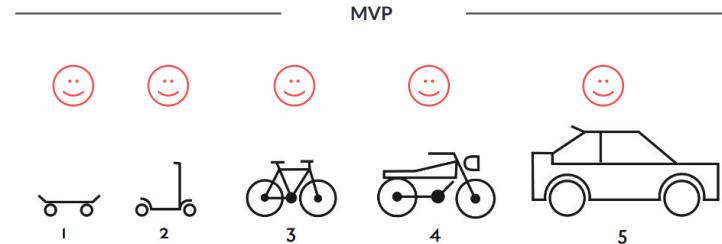
- Calculé à la fin du Sprint en additionnant les points de tous les récits "Terminées".
- Une récit qui n'est pas "Terminée" vaut 0 pt dans le calcul de la vélocité.
- La vélocité moyenne des sprints passés est utilisée pour guider la planification des prochains sprints.
- La planification de sprints basée sur la vélocité est basée sur la prémissse que la quantité de travail qu'une équipe va faire dans le prochain sprint est à peu près égale à ce qu'elle a fait dans les sprints précédents.

Velocity Chart



Minimum Viable Product

- Appellation française: Produit Minimum Viable
- Un produit avec juste assez de fonctionnalités pour satisfaire les premiers utilisateurs, et pour obtenir des commentaires pour aligner les prochains développements.
- Récolter des commentaires par rapport à un MVP est souvent moins cher que d'attendre de développer un produit avec plus de fonctionnalités.



Gâteau Minimum Viable?!

Supposons que vous êtes un boulanger et que vous avez un client:

- Voulant manger un gâteau complet
- Ne pouvant pas attendre le 50 min. nécessaire pour faire un gâteau complet
- Voulant le manger le plus tôt possible.
- Ne pouvant attendre plus que 35 min.

Que feriez-vous?

- Dire que c'est impossible et perdre l'opportunité
- Lui livrer un gâteau complet à moitié cuit.
- Négocier et demander au client d'attendre 15 minutes de plus pour lui offrir exactement ce qu'il a demandé.
- Donner au client un demi-gâteau entièrement cuit pour satisfaire sa faim (Cela prend seulement 30 minutes pour faire un demi-gâteau). Pendant qu'il mange le premier demi-gâteau, cuire un autre demi-gâteau.



Question?

La réponse est oui"

... mais pouvez-vous répéter la question?

Vous voulez en savoir plus sur Scrum?

- [Scrumguide.org](#)
- [Scrum.org](#)
- [Scrumalliance.org](#)
- [Mike Cohn's Blog](#)

Absolunet.

1 877 979 2276
Info@absolunet.com

www.absolunet.com

MONTRÉAL

4398 St-Laurent Blvd
Suite 204
Montréal (QC)
H2W 1Z5

KANSAS CITY

140 Walnut St.
Suite 204
Kansas City, Missouri
64106

STE-THÉRÈSE

4 260 rue Sicard
Bureau 300
Ste-Thérèse (QC)
J7E 3X4

TORONTO

180 John Street
Toronto, (ON)
M5T 1X5

ENGLISH

Absolut**net**.

Scrum 101

The Scrum Framework basics

Agenda

01

Traditional development
vs Agile development

02

What is Scrum?

03

Empiricism

04

Iterative and incremental
development

05

Values

06

Framework

07

Team & roles

08

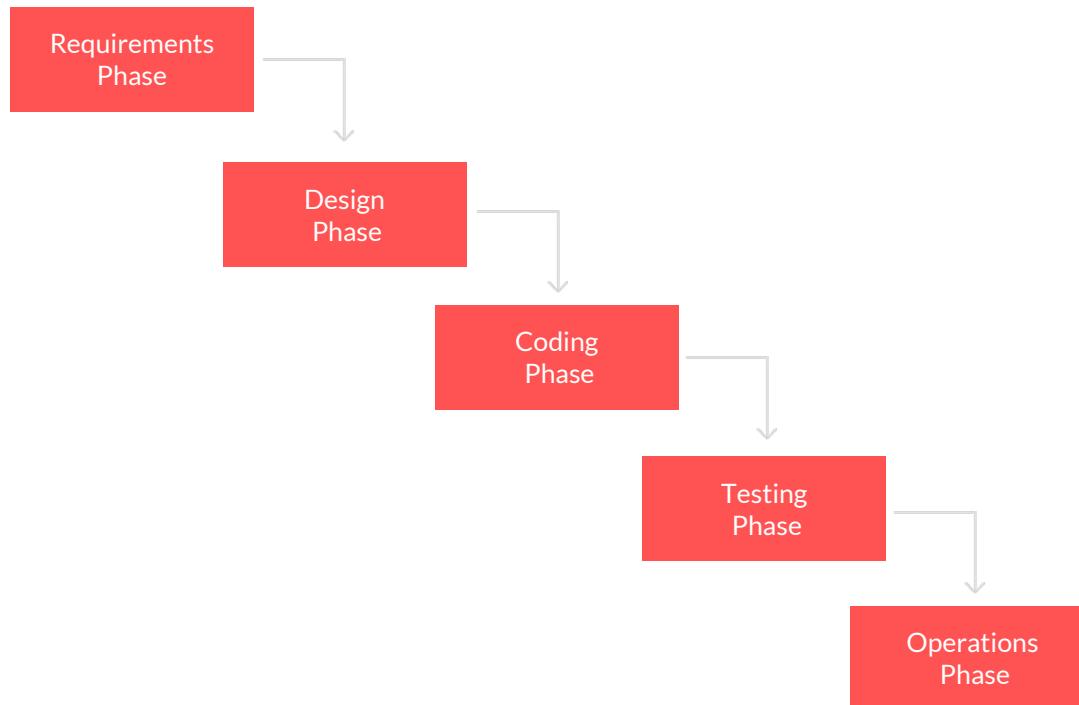
Events

09

Important Agile concepts

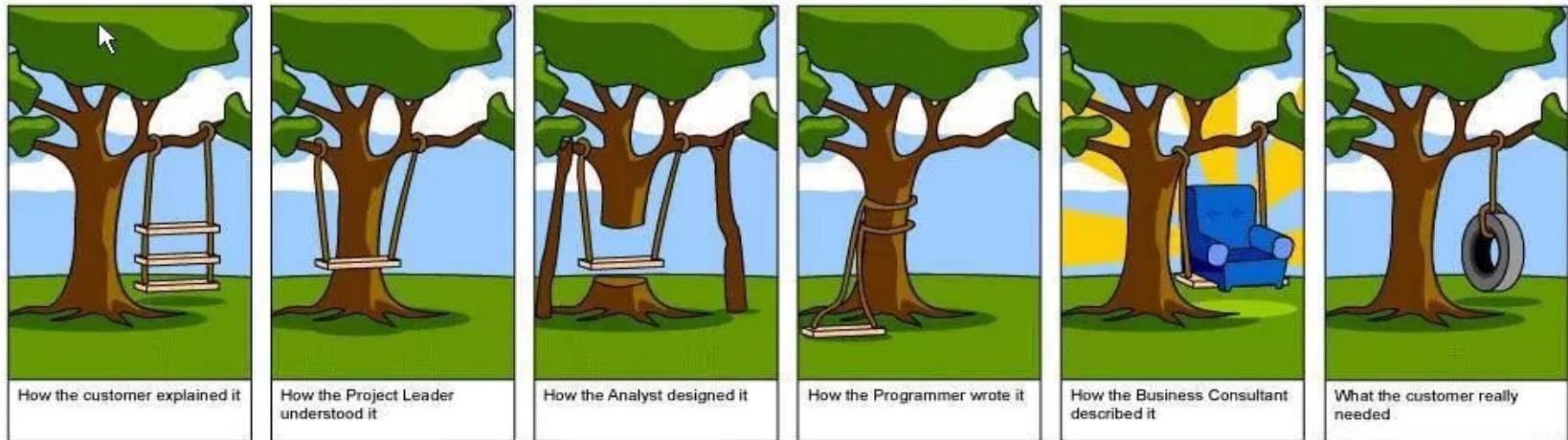
A.

Traditional approach – Waterfall



A.

Why waterfall often fails



3 Truths We must accept.

- 1
- 2
- 3

It is impossible to gather all the requirements at the beginning of a project

Whatever requirements you do gather are guaranteed to change

There will always be more to do than the time and money will allow

3 Truths We must accept.

- 1
- 2
- 3

It is impossible to gather all the requirements at the beginning of a project

Whatever requirements you do gather are guaranteed to change

There will always be more to do than the time and money will allow

3 Truths We must accept.

- 1
- 2
- 3

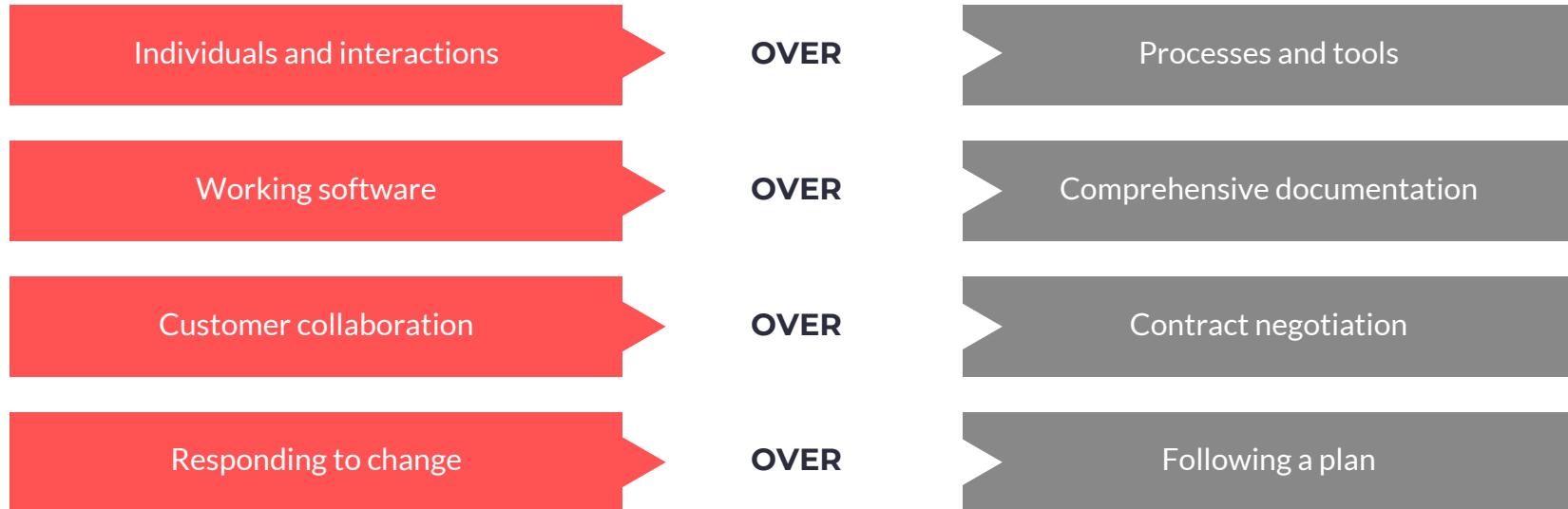
It is impossible to gather all the requirements at the beginning of a project

Whatever requirements you do gather are guaranteed to change

There will always be more to do than the time and money will allow

The 4 Values of the Agile Manifesto

In 2001, based on their experience the authors of the Agile Manifesto stipulated:
“while there is value in the items on the right, we value the items on the left more.”



What is Scrum?

It's a framework for developing, delivering, and sustaining complex products. In the early 1990's, Ken Schwaber and Jeff Sutherland developed Scrum. In 2010, they wrote the Scrum Guide.

The framework consists of:

- 3 Roles
- 4 Events
- 3 Artifacts
- Rules

Scrum is:

- Lightweight
- Simple to understand
- Difficult to master

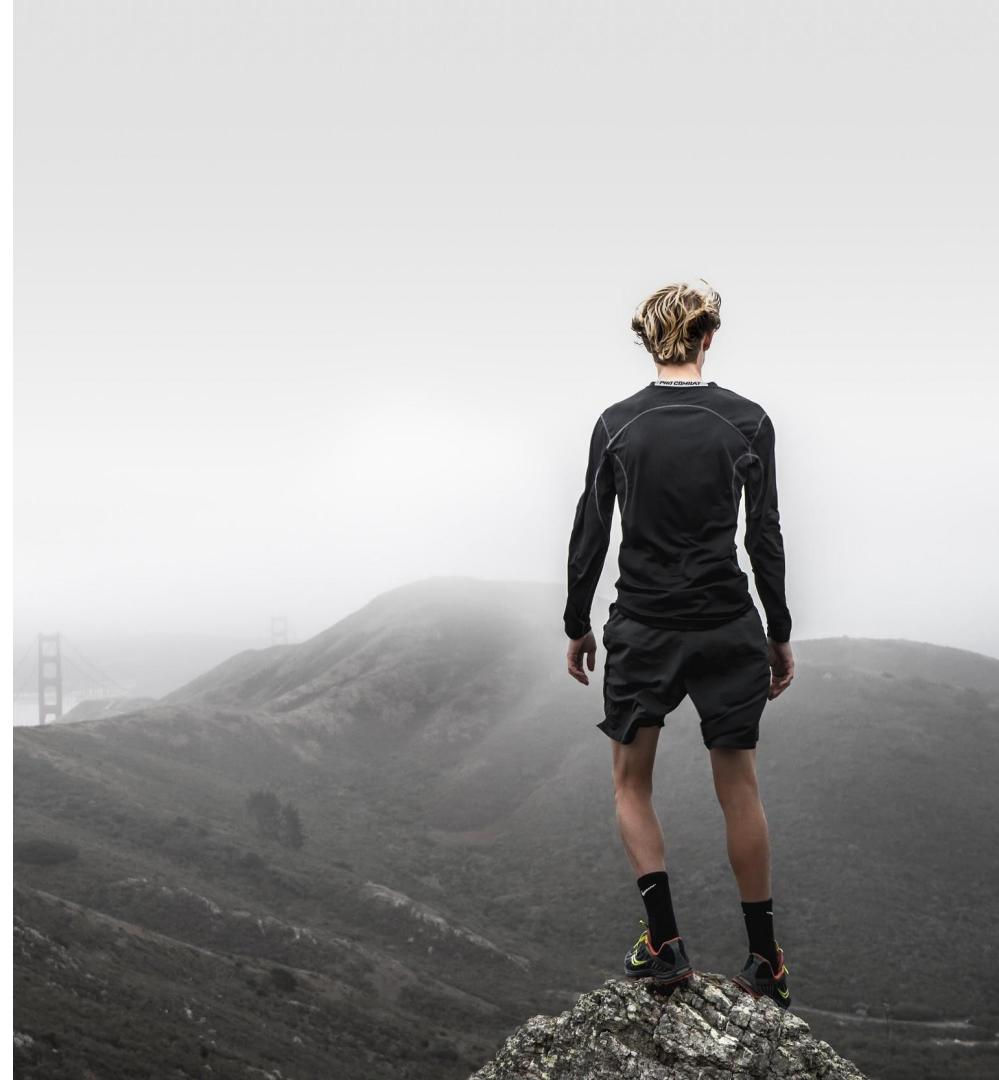


Empiricism

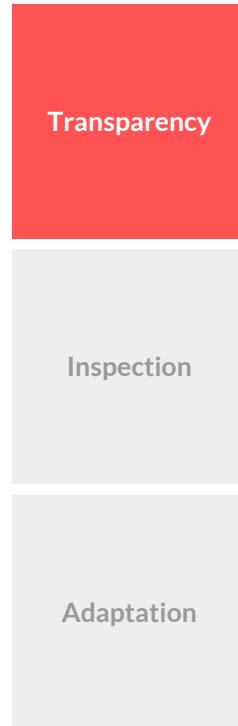
Scrum is founded on empirical process control theory.

Empiricism means working in a fact-based, experience-based, and evidence-based manner.

In Scrum progress is based on observations of reality, not fictitious plans.



The 3 Pillars of Empiricism.



This means presenting the facts as is.

Transparency requires that everyone involved shares the same language and understanding of reality.

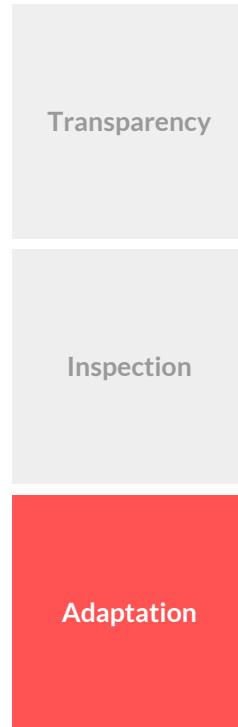
The 3 Pillars of Empiricism.



Inspection in this context is not an inspection by an inspector/auditor but an inspection by everyone on the Scrum Team and by the stakeholders.

In Scrum we must frequently inspect our plans, the way we work and what we deliver.

The 3 Pillars of Empiricism.



Adaptation is about continuous improvement, the ability to adapt based on the results of the inspection.

Everyone must ask this question regularly:
Are we better off than yesterday?

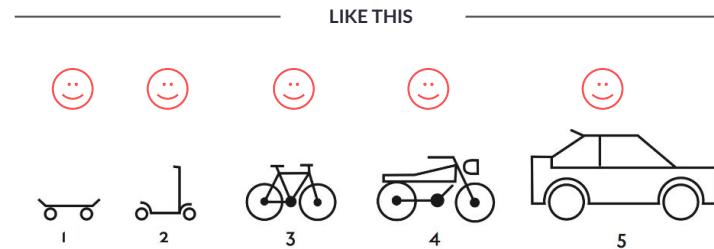
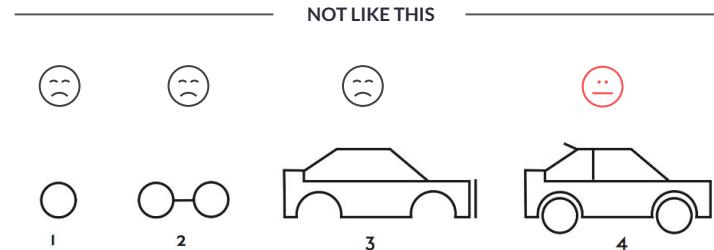
Scrum has four events where we can inspect and adapt.
(Planning, Daily, Review and Retro)

**Scrum is
iterative + incremental**

Iterative Development

The basic idea behind this method is to develop a system through repeated cycles (iterations).

Allowing to take advantage of what was learned during development of earlier iterations.

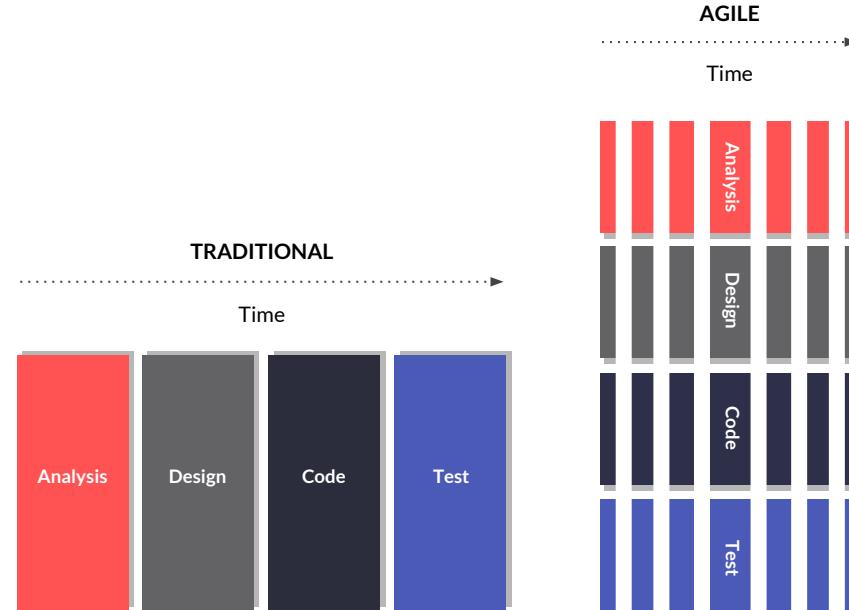


A.

Incremental Development

This method slices the system functionality into increments.

In each increment, a slice of functionality is delivered through cross-functional work, from the requirements to the deployment.



A.

**Success of Scrum depends on people
living by the 5 Scrum values.**

The 5 Scrum Values

Commitment

Committing to achieving the goals of the Scrum Team.

Courage

Having the courage to do the right thing and work on tough problems.

Focus

Focussing on the work of the Sprint + the goals of the team.

Openness

Agreeing to be open about the work and its challenges.

Respect

Respecting each other to be capable, independent people.

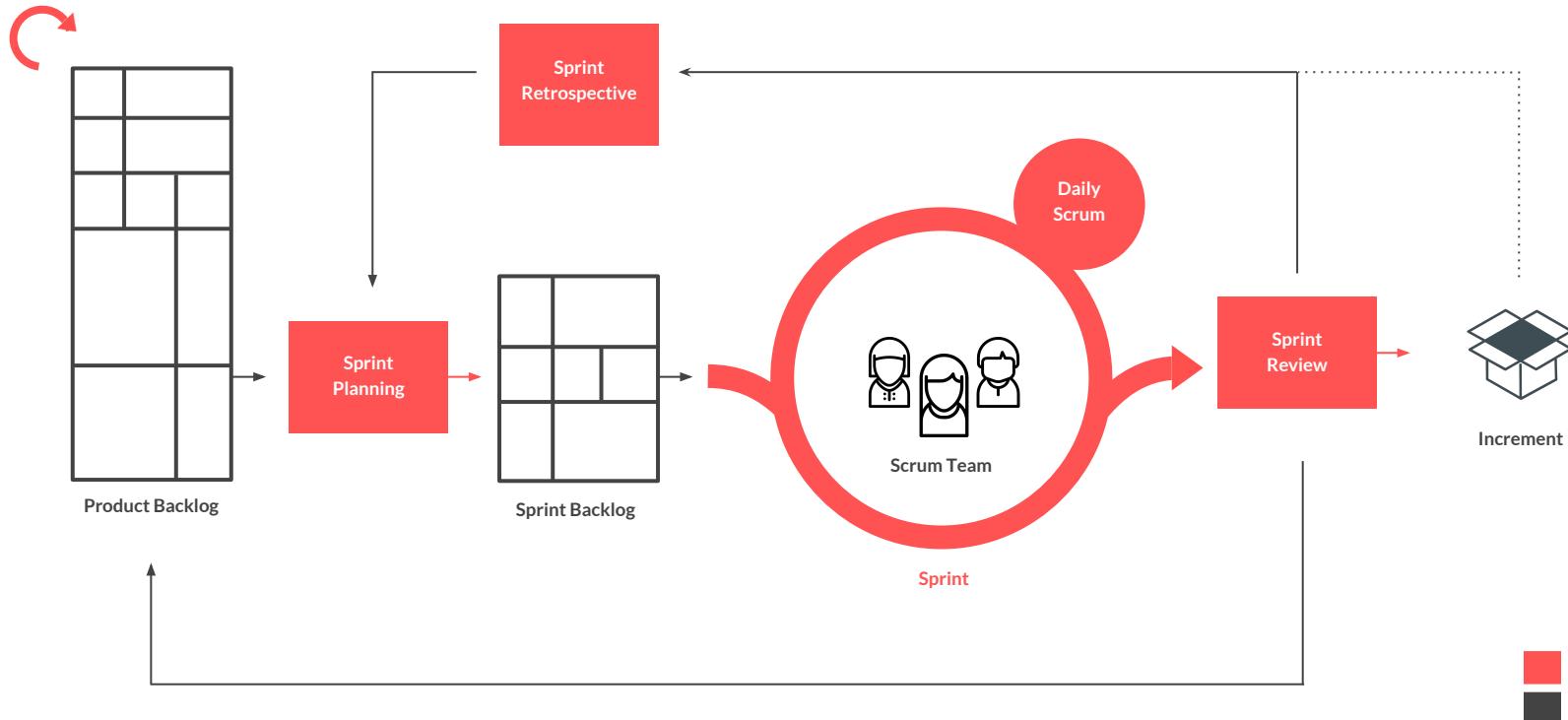
A.

The Framework

Roles + Events + Artifacts + Rules

A.

Scrum Framework

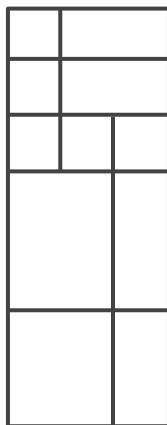


A.

The Artifacts

PRODUCT BACKLOG

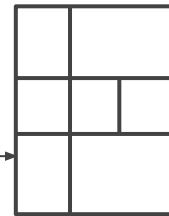
Ordered list of everything that is known to be needed in the product. This dynamic list is constantly adapted by the **Product Owner**.



Product Backlog

SPRINT BACKLOG

The sprint backlog includes the **Product Backlog** items selected for the Sprint by the Dev Team and the plan for delivering them.



Sprint Backlog

INCREMENT

The product increment is the set of items completed during this Sprint, and also those completed during previous Sprints.

Items are done when they meet the **Definition of Done**.

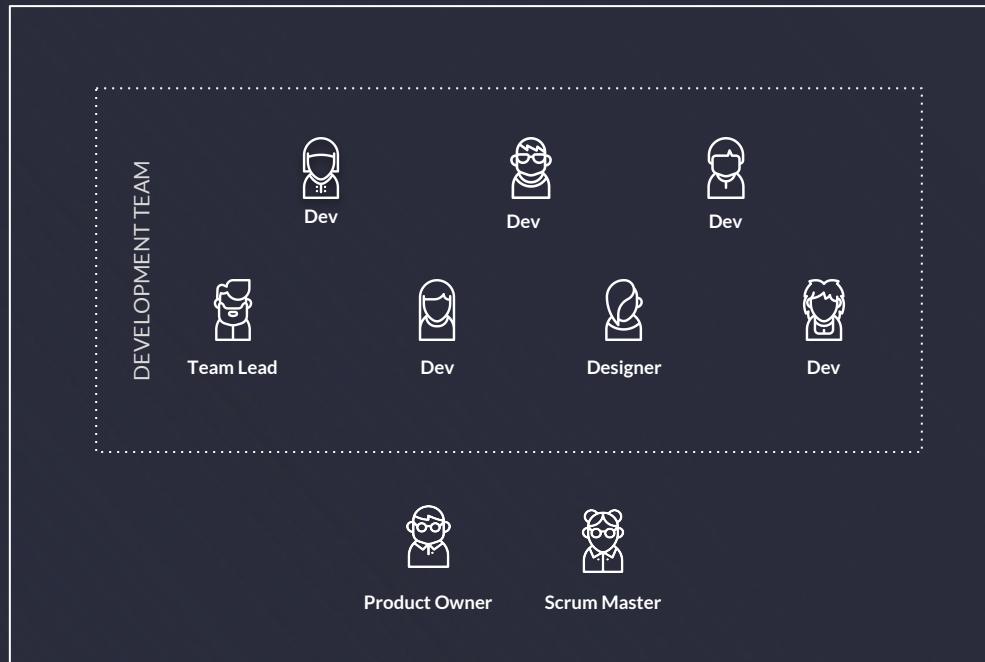
This increment must be usable, whether or not it is deployed in production.



Increment

A.

The Scrum Team



A.

SCRUM ROLE

Product Owner

- Brings the vision & strategy
- Understands users + clients
- Advises the client.
- Creates + orders the backlog (value + risk).
- Explains backlog items to the team.



A.

**The Product Owner is the sole person
responsible for managing the Product Backlog.**

SCRUM ROLE

Dev Team

- Team of 3 to 9 people
- Cross-functional (Front-end Developers, Back-end Developers, UX, Designers, etc...)
- Empowered
- Self-organizing
 - mitigates risks and finds its own solutions
 - knows what needs to be done
 - is aware of its velocity/performance and capacity
 - communicates its actions
- Accountability belongs to the whole team.
- Owner of the Sprint Backlog



A.

During a sprint, any changes to the Sprint Backlog must be discussed with and approved by the Dev Team.

SCRUM ROLE

Scrum Master

- Coach
- Trainer
- Servant-leader
- Guardian of the framework
- Ensures Scrum is understood and enacted
- Facilitates Scrum Events
- Makes sure impediments are addressed



A.

The Scrum Master is not a project manager and must not decide who works on what and when.

The Dev team is self-organizing.

A.

The Scrum Events

Planning + Daily + Review + Retrospective

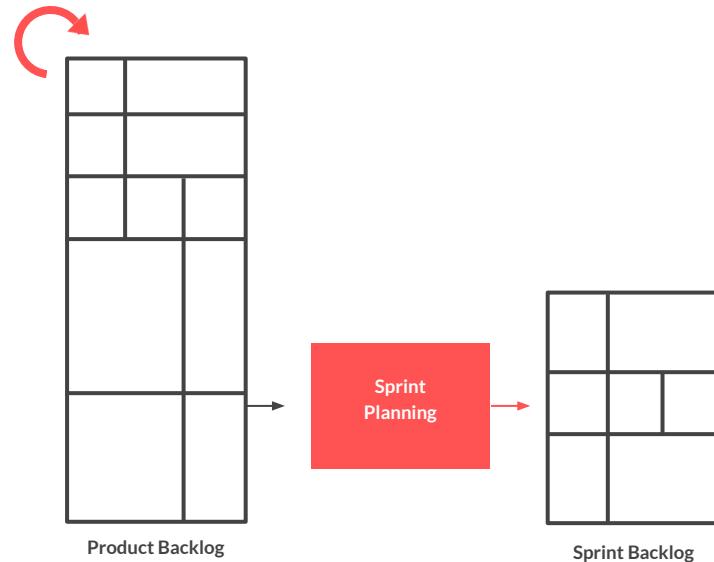
A.

SCRUM EVENT

Planning

Sprint Planning answers the following:

- Topic 1: What can be done this Sprint?
- Topic 2: How will it be done?



SCRUM EVENT

Planning

Topic 1: What can be done this Sprint?

- **The Scrum Team discusses**
 - the desired Sprint Goal
 - the desired Product Backlog items to complete to achieve the Sprint Goal.
- **Meeting input:**
 - Product Backlog
 - Dev Team data:
 - Past performance (velocity)
 - Projected capacity
 - Latest product Increment
(Done functionalities)
- **The whole Scrum Team defines a Sprint Goal.**

SCRUM EVENT

Planning

Topic 2: How will it be done?

- **The Dev Team:**
 - decides how it will build functionalities into a "Done" product Increment during the Sprint (tasks, etc...)
 - determines if it has too much or too little work and may renegotiate the selected Product Backlog items with the PO.
- The **Sprint Backlog** is created during the Sprint Planning.

A.

The number of items selected for the Sprint Backlog is **solely up to the Dev Team.**

Only the Dev Team can assess **what it can accomplish over the upcoming Sprint.**

SCRUM EVENT

Daily Scrum

- Daily 15-minute time-boxed event
- Dev Team plans together the next 24 hours.
- Inspect progress toward the Sprint Goal and how progress is trending toward completing the work in the Sprint Backlog (burndown chart).
- **Some Dev Teams will use questions like:**
 - What did I do yesterday?
 - What will I do today?
 - Do I see any impediment?

**The Daily Scrum is not a status meeting
held for the PO and Scrum Master.**

**It's a daily collaborative planning
session for the Dev Team.**

**If others are present, they shouldn't
disrupt this valuable session.**

SCRUM EVENT

Sprint Review

- The PO explains to key stakeholders what items have been "Done" and what has not been "Done".
- Dev Team discusses:
 - what went well during the Sprint
 - what problems it ran into
 - how those problems were solved
- Dev Team demonstrates the "Done" work and answers questions about the **Increment**.
- PO discusses backlog and timeline.
- Probable Product Backlog items for the next Sprint are identified.

A.

The Sprint Review is not just a Demo.

Definition of Done

- The purpose of each Sprint is to deliver Increments of potentially releasable functionality that adhere to the current definition of "Done".
- Everyone must understand what "Done" means to ensure transparency.
- It guides the Dev Team in knowing how many Product Backlog items it can select during a Sprint Planning.
- As Scrum Teams mature, it is expected that their definitions of "Done" will expand to include more stringent criteria for higher quality.

SCRUM EVENT

Sprint Retrospective

Purpose:

- Inspect how the last Sprint went with regards to people, relationships, process, and tools
- Identify the major items that went well and potential improvements
- Create a plan for implementing improvements.

Scrum Team plans ways to increase product quality by improving work processes or adapting the definition of "Done"

A.

**The retrospective must stay respectful,
constructive and effective.**

A.

Scrum Events time box

(can be less)

Event	4 week	3 week	2 week	1 week
Sprint Planning	8 hr	6 hr	4 hr	2 hr
Daily Scrum	15 min daily	15 min daily	15 min daily	15 min daily
Sprint Review	4 hr	3 hr	2 hr	1 hr
Sprint Retrospective	3 hr	2.25 hr	1.5 hr	.75 hr

Refinement Sessions

- Product Backlog items are reviewed and revised by the Scrum Team.
- In these sessions we:
 - Add details
 - Estimate
 - Order items
 - Split/merge items
 - Remove irrelevant items
- The Scrum Team decides how and when refinement session are held but should consumes no more than 10% of the capacity of the Dev Team.

The Dev Team is responsible for all estimates.

The Product Owner may influence by helping it understand and select trade-offs,

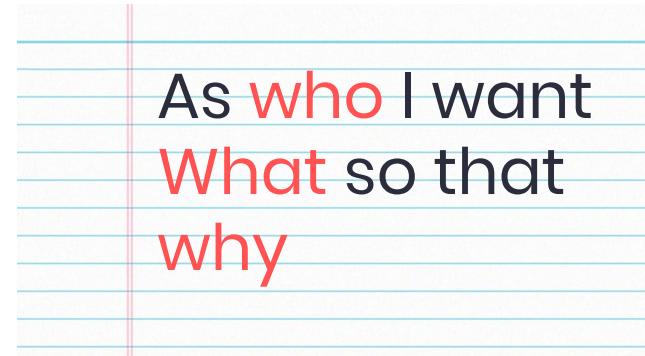
but the **people who will perform the work** make the final estimates.

Important Agile concepts

- User Stories
- Qualities of good user stories
- Story points
- Velocity
- MVP

User Stories

- User stories are short, simple descriptions of a feature told from the perspective of the person who desires the new capability, usually a user or customer.
- They typically follow a simple template:
As a persona, i want to action so that i can fulfill a need.
- They shift the focus from writing about features to discussing them.



Qualities of Good User Stories

Independant

Self-contained, no inherent dependency on another backlog item.

Negotiable

Not an explicit contract, should leave space for discussion.

Valuable

Must deliver value to the stakeholders.

Estimable

Team must always be able to estimate the size.

Small

Should not be so big as to become impossible to plan/task/prioritize.

Testable

Must provide the necessary info to make test development possible.

Story points

Story points are used to estimate the overall effort that will be required to get a story to “Done”.

It includes:

- Amount of work
- Complexity
- Risk and uncertainty

Poker planning is often used to estimate stories in points.

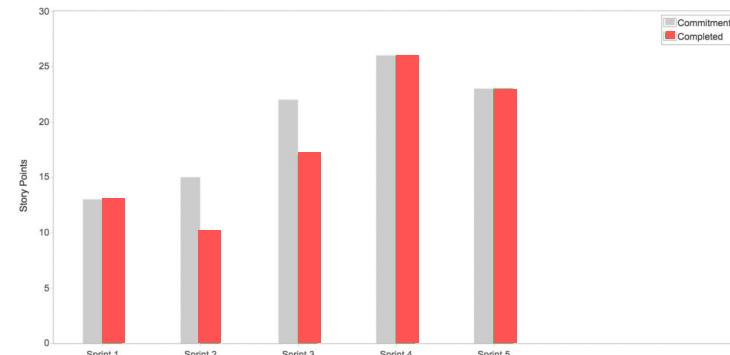


Velocity

(pts/sprint)

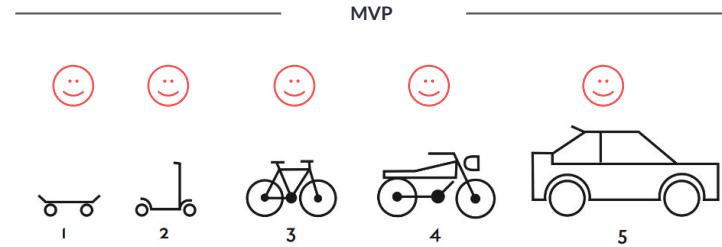
- Calculated at the end of the Sprint by totaling the Points for all the “Done” User Stories.
- A story that isn’t “Done” is worth 0 pts when counting the velocity.
- Average velocity from past sprints is used to guide the planning the upcoming sprints.
- Velocity-driven sprint planning is based on the premise that the amount of work a team will do in the coming sprint is roughly equal to what they’ve done in prior sprints.

Velocity Chart



Minimum Viable Product

- A product with just enough features to satisfy early customers, and to provide feedback for future product development.
- Gathering insights from an MVP is often less expensive than developing a product with more features.



Minimum Viable Cake?!

Suppose you are a baker and you have a customer who:

- Wants to eat a full cake
- Can't wait 50 min. required to bake a full cake
- Wants to eat it as soon as possible.
- The max. time he can wait is 35 min.

What would you do?

- Say it's impossible and lose the opportunity
- Deliver a half-baked full cake to him.
- Negotiate and ask the customer to wait another 15 min. to bake what he asked for.
- Give the customer a fully-baked half cake to satisfy his hunger (It takes only 30 minutes to make a half cake). While he eats the first half-cake, bake another half cake.



Question?

The answer is “yes”

... but can you repeat the question?

Want to learn more about Scrum?

- [Scrumguide.org](#)
- [Scrum.org](#)
- [Scrumalliance.org](#)
- [Mike Cohn's Blog](#)

Absolunet.

1877 979 2276
Info@absolunet.com

www.absolunet.com

MONTRÉAL

4398 St-Laurent Blvd
Suite 204
Montréal (QC)
H2W 1Z5

KANSAS CITY

140 Walnut St.,
Suite 204
Kansas City, Missouri
64106

STE-THÉRÈSE

44 260 rue Sicard
Bureau 300
Ste-Thérèse (QC)
J7E 3X4