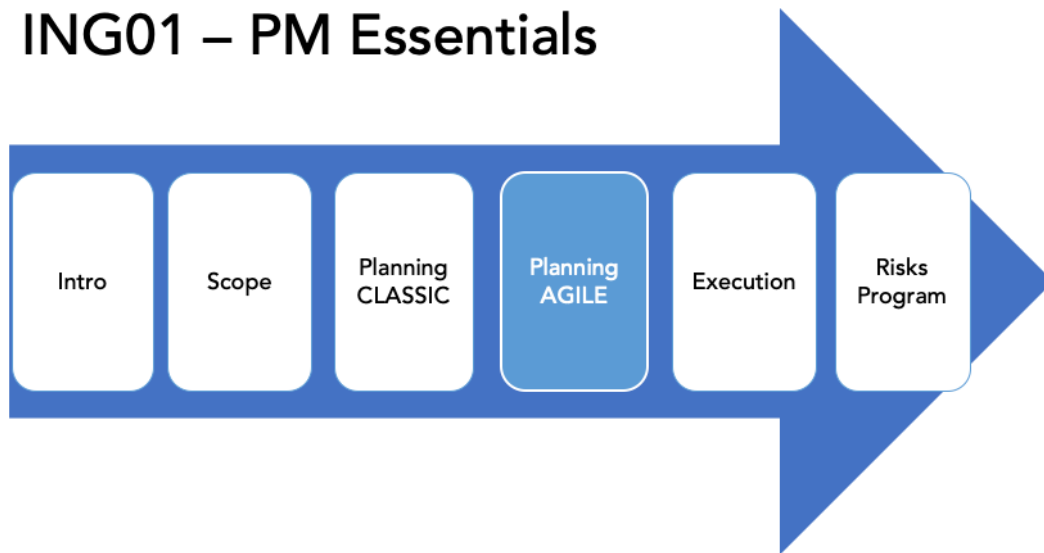


ING01 – PM Essentials





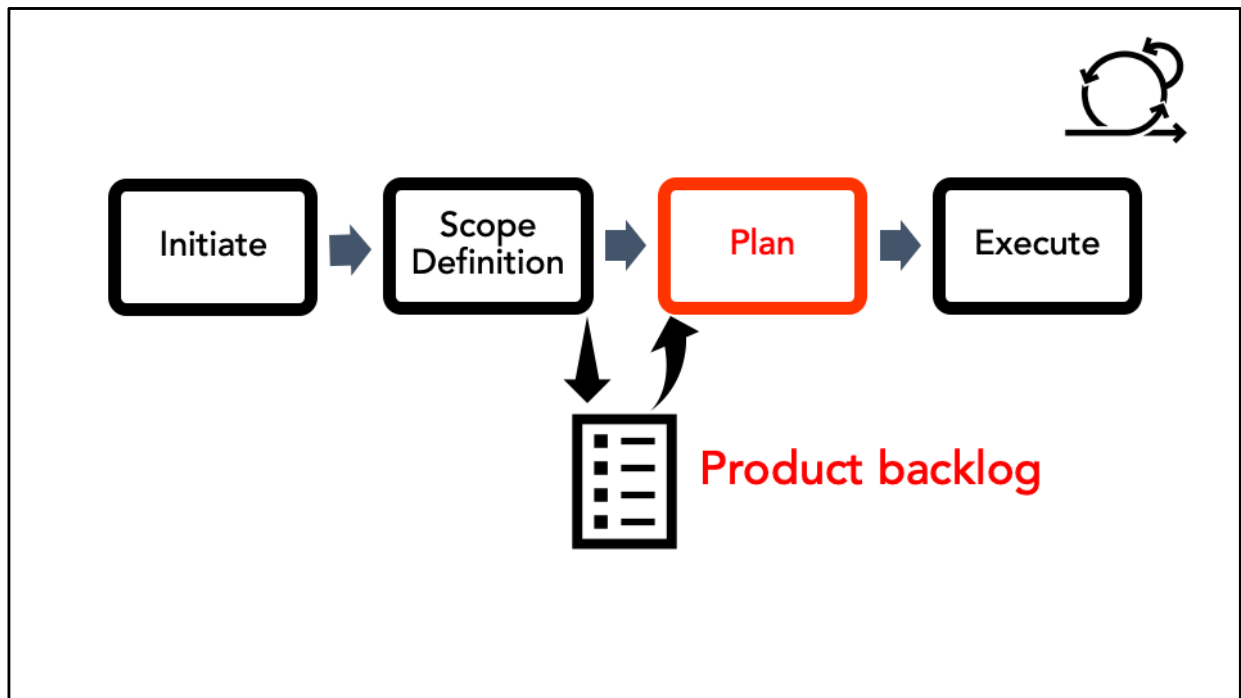
ING1-GPRO-Module 4

Planning AGILE

Planification Projet, Méthode AGILE

The initial planning of the project in the Agile methodology may seem less crucial than in the conventional method. Nevertheless, moving from a Product Backlog to developing a first sprint requires a set of planning activities that we are reviewing in this module

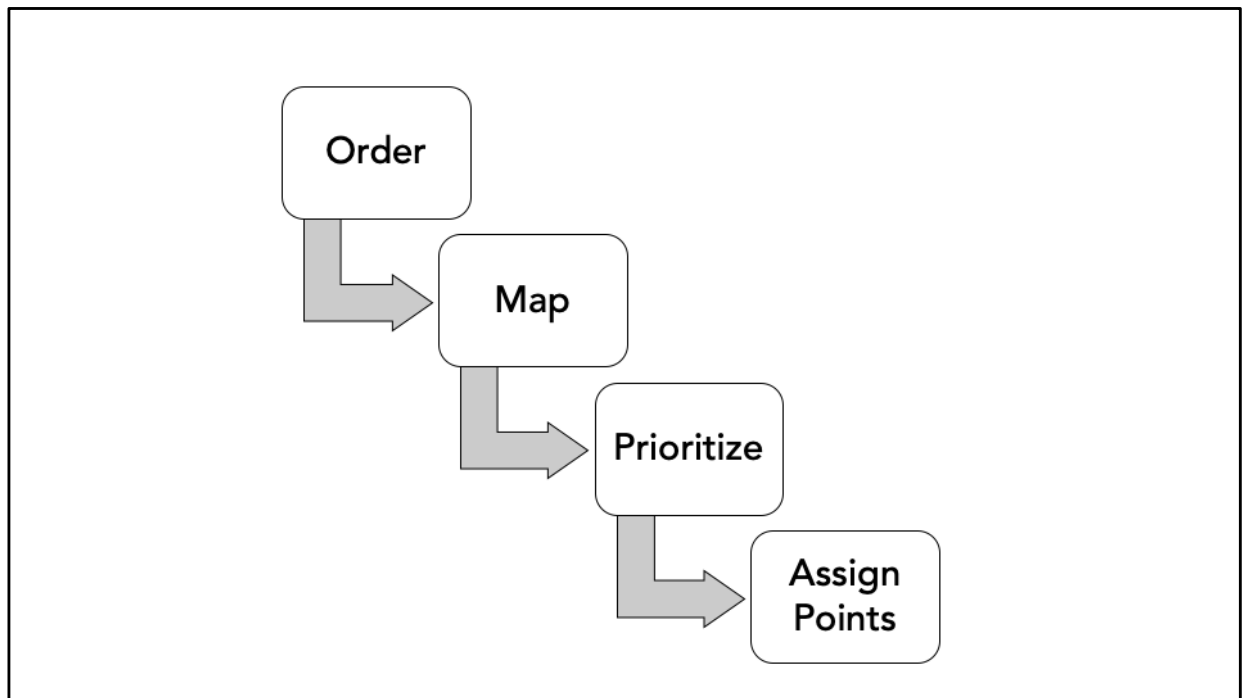
La planification initiale du projet dans la méthode agile peut paraître moins cruciale que dans la méthode classique. Néanmoins, passer d'un Product Backlog au développement d'un premier sprint nécessite un ensemble d'activités de planning que nous allons parcourir dans ce module



At the end of the Scope definition Phase we have build our first version of the product backlog with mainly user stories describing the functional requirements accompagnied with technical, quality and project constraints. This is the pre-réquisite to the planning phase.

As we will be in position to manipulate and add informations to the Product backlog , the planning phase is performed with Product Owner and Dev. Team under the teaming of the Scrum Master.

À la fin de la phase de SCOPE definition , nous avons construit notre première version du Product backlog avec principalement des user stories décrivant les exigences fonctionnelles accompagnées de contraintes techniques, de qualité et de projet. Il s'agit d'un pré-réquis à la phase de planification. Comme nous serons en situation de manipuler et d'ajouter des informations au Product Backlog, la phase de planification est effectuée avec le Product Owner et l'équipe de développement sous l'égide du Scrum Master.



Processus de Classification des User Story dans le Product Backlog



Set an order to User stories Ordonner les User Stories

The first task is to order the Product Backlog. The principle is to detect the part of the product that brings the most value and order then based on this criteria

This P.B. scheduling can use different methods

- Simple methods such as MOSCOW :
 - Mo — Must have: must be achieved.
 - S — Should have: should be carried out if possible.
 - Co — Could have: could be achieved if there is no impact on other tasks in progress.
 - W — Won't have: will not be realized right away but would be desirable
- Most complex: For instance “Story mapping” that combines the analysis of the relationships between stories, the analysis of the impact on the business, This may lead to a product roadmap

Priorities will be deduced and these priorities will contribute to the ordering of the Product Backlog, but other features (e.g. the expected availability date) may also be included in the ordering of the Product Backlog.

Ordering the PB is part of the PB refinement. We put in place an initial order that will be reviewed regularly, especially in the PB Grooming process.

La première tâche consiste à ordonner le Product Backlog. L'idée générale consiste à essayer de déterminer ce qui a le plus de valeur pour dans le produit pour les utilisateurs et d'ordonner suivant ces critères .

Cet ordonnancement du P.B. peut utiliser différentes techniques

- *les plus simples type MOSCOW :*

Mo — Must have : doit être réalisée.

S — Should have : devrait être réalisée si possible.

Co — Could have : pourrait être réalisée s'il n'y a pas d'impact sur les autres tâches en cours.

W — Won't have : ne sera pas réalisée tout de suite mais serait souhaitable

- *Des plus complexes : Story mapping qui mêlent l'analyse des relations entre les story , l'analyse de l'impact sur le business pour aboutir à une roadmap du produit*

On va en déduire des priorités et ces priorités vont contribuer à la mise en ordre du Product Backlog, mais d'autres caractéristiques (par exemple la date de disponibilité attendue) peuvent aussi rentrer en compte dans la mise en ordre du Product Backlog.

Le fait d'ordonner le PB participe au PB refinement. Nous mettons en place un ordre initial qui sera revu régulièrement notamment dans le process de PB Grooming.

Simple methods such as **MOSCOW**:

- **Mo** — Must have: must be achieved.
- **S** — Should have: should be carried out if possible.
- **Co** — Could have: could be achieved if there is no impact on other tasks in progress.
- **W** — Won't have: will not be realized right away but would be desirable

Story Mapping

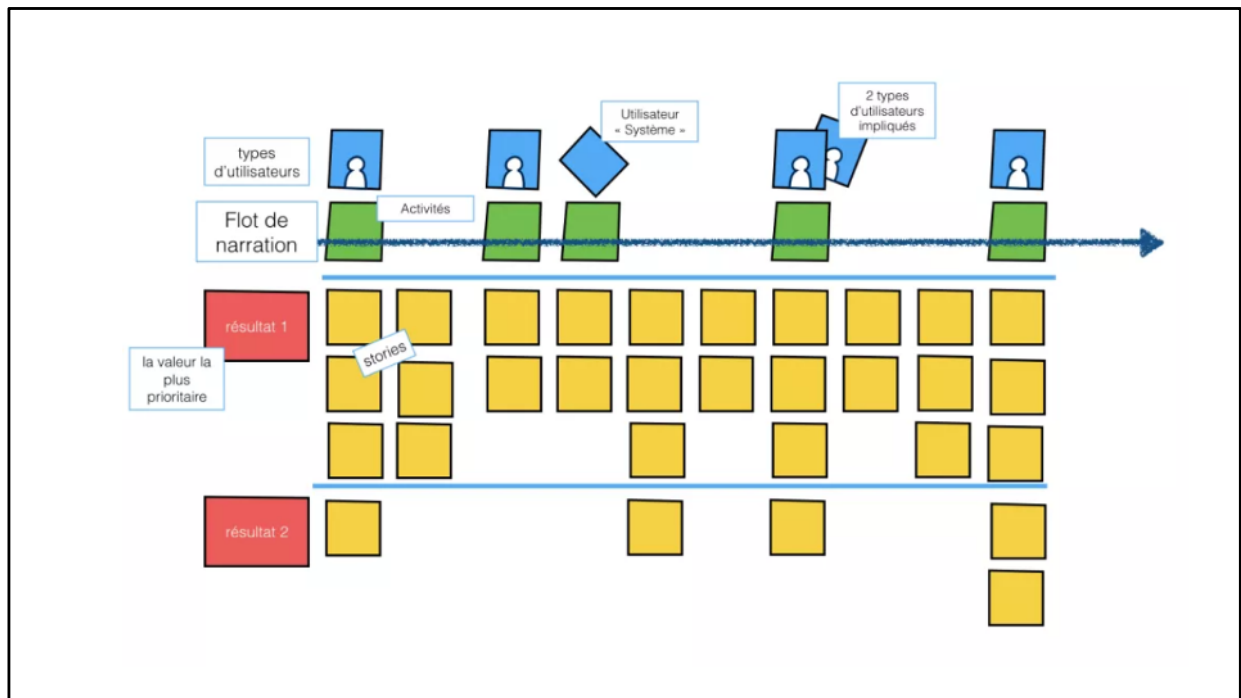
This exercise consists in having a different representation of the product backlog to help in :

- Determining the interaction between US (interdependances)
- Determining the value and the priority based on a rational method

This technique brings rational in the way we select the stories that need to be done first , and second and third

I may also be helpful in determining the release carving

This technique is just about helping , and is not absolutely compulsory .



The basic idea is to organize the User story in a table with two axis :

Horizontal is flow of user Journey in the application , there is in addition , the representation also of the concerned users
Vertical corresponds to the priority of user stories.

This exercise can be done in a workshop



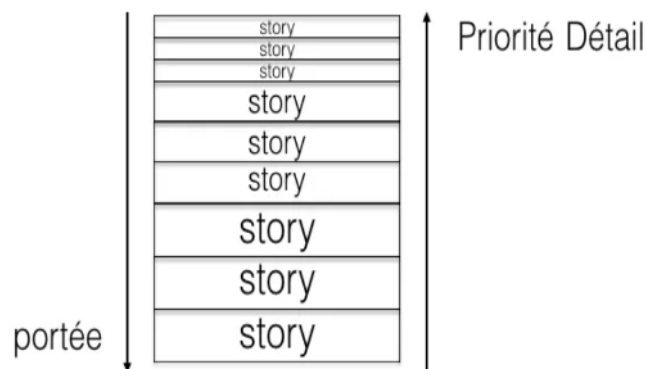
Set Priorities

Affecter des priorités

Thanks to the order settings of the Product back log, we have grouped the user stories which are have high dependancies , we have determined the ones that bring the highest value. Then we have just to give a priority ranking based on this information. We will have then on the top of the product backlog the stories or group of stories that have the highest priority up to the bottom of the P.B with low priority stories. Most of the time the highest priority user stories are precisely stated and give a detailed description of what is expected. On the other end , at the bottom of P.B. we will just find EPICS or even ideas of new features .. Product Owner is the only person that update the P.B.

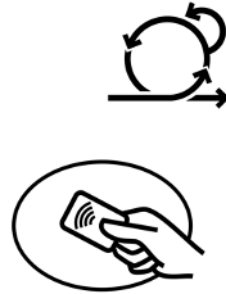
Grâce à l'ordonnement du P.B., nous avons regroupé les user stories qui sont ont des dépendances élevées, nous avons déterminé celles qui apportent le plus de valeur. Alors nous avons juste à procéder à un classement en affectant des priorités basées sur ces informations. Nous aurons alors en tête du P.B. les stories ou le groupe de Stories qui ont la plus haute priorité et en bas du P.B les stories à faible priorité. La plupart du temps, les U.S. les plus prioritaires sont énoncées avec précision et donnent une description détaillée de ce qui est attendu. Par contre , au bas du P.B. nous aurons juste des EPICS ou même des idées de nouvelles fonctionnalités .. Le Propriétaire est la seule personne qui met à jour le P.B.

Product Backlog prioritized

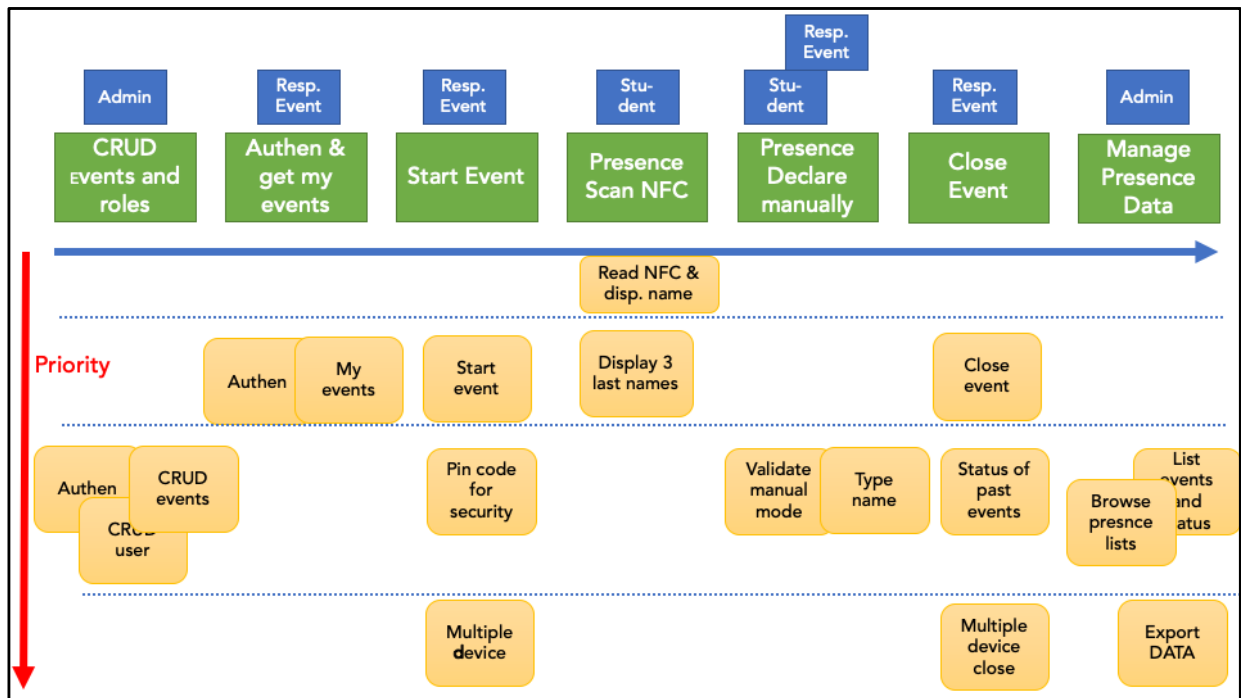


This picture represents the product backlog once prioritised. At the top the user stories that bring the most value and the user story are then detailed . At the bottom the US that are the less important and are described roughly.

Presence Project : What are the priorities?



Recall Presence Project context and Ask to students to define priorities to user stories





First horizon of planning: **The Release**

Premier Horizon de planification : La Release

Release: a coherent set of features that will be made available to users thru a deployment in production.

Once we have prioritized our product backlog, we are at a point where we can define releases, and at least the first release.

Very simply we can draw a line where all the stories from top to this line are part of the first release



Assign « Story Points » Affecter des « Story points »

For each User Story, the development team will then assign a number of points: This is an estimate of the difficulty in implementing this story. The difficulty includes **complexity, risk** and **effort**. It is above all a **relative** estimate between the different user stories.

Scales are used to differentiate between these difficulty levels:

Example: Fibonacci Suite: 1,2,3,5,8,13,21

The idea is that it is not the Product owner, nor the Scrum Master who defines these points but directly the development team with the search for consensus among the members.

This initial assignment will be reviewed very regularly during project execution

Ce processus vise à qualifier dans le product backlog la difficulté de développer chacune des user story.

*Pour chaque User Story, L'équipe de développement va lui affecter un nombre de points: C'est une estimation de **la difficulté** à implémenter cette story. La difficulté inclut **la complexité, le risque et l'effort**.*

C'est surtout une estimation relative entre les différentes user stories.

This process aims to qualify in the product backlog the difficulty of developing each user story.

On utilise des échelles qui permettent de bien différencier ces niveaux de difficulté:

Exemple: Suite de Fibonacci: 1,2,3,5,8,13,21

L'idée est que ce n'est pas les Product owner , ni le Scrum Master qui définisse ces points mais directement l'équipe de développement avec la recherche d'un consensus entre les membres

Cette affectation initiale sera revue très régulièrement lors de l'exécution du projet.

Suite de Fibonacci
1, 2, 3, 5, 8, 13, 21, ~40

Presence Project : Story points ?



Recall Presence Project context and Ask to students to define priorities to user stories

En tant que	Je veux	Afin de	Points
Resp. Événement	m'identifier en badgeant ma carte et voir les événements qui me sont associés sur le device d'enregistrement	de pouvoir retrouver l'événement en cours	8
Resp. Événement	lancer la session de présence sur un événement	commencer le badgeage de présence des participants	2
Administrateur	Gérer la liste des événements	créer , supprimer ,modifier les événements	13
Professeur	Controler la présence sur la base d'une liste d'élèves devant suivre le cours	pointer les absents aux cours apprentis	40 ?
Participant	la lecture du badge RFC soit rapide en ne necessite pas des essais infructueux	d'assurer la fluidité des operation	5



Sprint = Timebox < 1 mois

Définition contenue dans [scrumguide.org](https://www.scrumguide.org) :

The Sprint: The heart of Scrum is a Sprint, a **time-box of one month or less** during which a "Done", useable, and potentially releasable product Increment is created. Sprints have consistent durations throughout a development effort. A new Sprint starts immediately after the conclusion of the previous Sprint.

During the Sprint:

- No changes are made that would endanger the Sprint Goal;
- Quality goals do not decrease; and,
- Scope may be clarified and re-negotiated between the Product Owner and Development Team as more is learned.

Each Sprint may be considered a project with no more than a one-month horizon. Like projects, Sprints are used to accomplish something. Each Sprint has a goal of what is to be built, a design and flexible plan that will guide building it, the work, and the resultant product increment.

Sprints are limited to one calendar month. When a Sprint's horizon is too long the definition of what is being built may change, complexity may rise, and risk may increase. Sprints enable predictability by ensuring inspection and adaptation of progress toward a Sprint Goal at least every calendar month. Sprints also limit risk to one calendar month of cost.

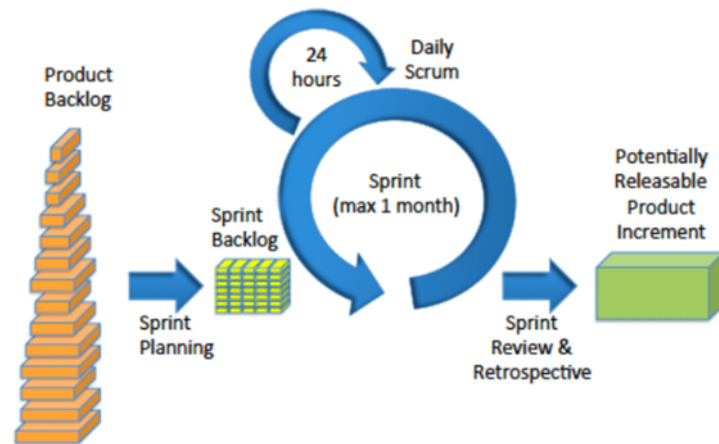
Le coeur de Scrum est le Sprint, qui a une boîte de temps (time-box), une durée, d'un mois ou moins au cours de laquelle un Incrément Produit « Fini » fonctionnel et potentiellement publiable est créé. Les sprints ont une durée cohérente durant la phase de développement. Un nouveau Sprint commence immédiatement après la conclusion du Sprint précédent.

Pendant le Sprint:

- L'objectif du sprint est fixe ; les changements qui le remettent en cause ne sont donc pas permis ;*
- Les objectifs de qualité sont maintenus ; ils ne sont jamais revus à la baisse ; et,*
- Le périmètre peut être clarifié et renégocié entre le Product Owner et l'équipe de développement selon ce que l'équipe Scrum apprend.*

Chaque sprint peut être considéré comme un projet n'ayant qu'un horizon d'un mois. À l'instar d'un projet, un Sprint est utilisé pour accomplir quelque chose. Chaque Sprint a un objectif de ce qui doit être construit, une conception (design) et un plan flexible qui guidera la construction, le travail lui-même et l'incrément produit résultant.

Sprint



AGILE : Approach based on SCRUM: <https://www.scrumguides.org/index.html>



Sprint Carving Découpage en Sprints

- First step: define the duration of a sprint - according to the capacity of the development team.

Example: 1 month - 4-month project so **4 Sprints**.

Be careful, each Sprint must produce an increment (PI) and therefore the team must have time to design, realize, and TEST within a Sprint time and the result shall be validated by the customer.

So do not make sprint very short Especially for students who do not work full time on the project ...

- Step Two: Define the content of each sprintEach sprint must contain about the same number of points.
 - A Sprint must produce an Increment (PI), which is a working product.
 - Sprints must also take into account the product backlog and the dependence between the U.S.

Première étape: définir la durée d'un sprint – en fonction des capacités de l'équipe de développement.

Exemple: 1 mois – Projet de 4 mois donc 4 Sprints.

Attention , il faut que Chaque Sprint produise un increment (PI) et donc il faut que l'équipe ait le temps de concevoir, réaliser , et TESTER au sein d'un Sprint et présenter/ faire valider le resultat par le client . Donc pas de sprint très court non plus surtout pour des Etudiants qui ne travaillent pas a temps plein sur le projet...

Deuxième étape: définir le contenu de chaque sprint

Chaque sprint doit contenir à peu près le même nombre de points.

Un Sprint doit produire un Increment (PI), qui est un produit fonctionnant.

Les Sprint doivent aussi prendre en compte l'ordonnement du Product Backlog et les dépendance entre les U.S.



The Release

Release has been set in terms of features that will be made available to users.

Sprint carving shall take this into account

Eg: The 4 first sprints of the project comes into a coherent and usefull software application, the result of this release will be placed in production for the first utilization of the product .

Sprint 0 ?

Sprint 0: Get some technical & quality items on the Product Backlog, Provide a minimal environment that enables the writing of quality code, and. Write a piece of real code, no matter how small.

This shall be a quite small time unit where the dev team defines the technical architecture of the application and sets up this development environment .

Il s'agit de définir un socle minimum avec l'architecture produits, les composants logiciels, l'environnement de développement qui va permettre de dérouler des sprints.

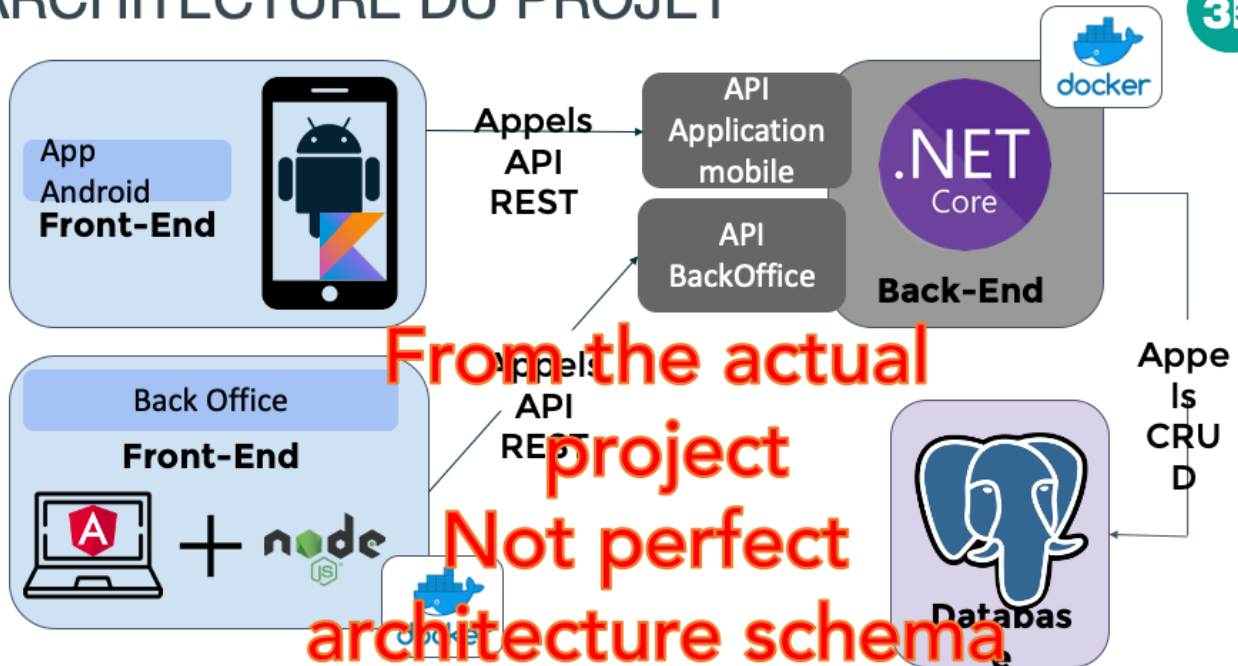
Notion de ENABLER (SAFe)

Presence Project : Sprint 0?



Recall Presence Project context and Ask to students to define priorities to user stories

ARCHITECTURE DU PROJET



En tant que	Je veux	Afin de	Critère d'acceptation	Type	Size	Sprint
Resp. Événement	lancer la session de présence sur un événement	commencer le badgeage de présence des participants	device prêt à l'opération de recueil de présence	Requirement	8	Sprint 1
Participant	pouvoir badger ma carte RFC sur un Mobile	de m'inscrire à l'événement en cours	Les badges lus sont enregistrés dans la liste des présents	Requirement	3	Sprint 1
Resp. Événement	Clore la session d'enregistrement	fermer la prise de présence sur l'événement et produire la liste de présence	Evenement clos plus de prise en compte de nouvelle présence .	Requirement	5	Sprint 1
Resp. technique	disposer d'un ensemble de documents (architecture , spec fonctionnelle , documentatation de la réalisation)	de pouvoir confier la poursuite de la réalisation et la maintenance à d'autres équipes		Non Fonction.	13	Sprint 1
Resp. Événement	pouvoir proposer un enregistrement manuel	enregistrer des participants sans carte	prêt à saisir	Requirement	5	Sprint 2
Participant	Saisir les informations me concernant sur enregistrement manuel	enregistrer ma présence	présence enregistrée	Requirement	5	Sprint 2
Resp. Événement	verrouiller toutes les actions qui me sont réservées par un code pendant l'utilisation en mode présence	d'interdire une utilisation par les participants de mes prérogatives	accès impossible sans code de déverrouillage	Requirement	3	Sprint 2
Administrateur	accéder à une interface WEB d'administration accessible sur le réseau de l'école en m'identifiant avec mes credentials habituels	d'administrer les événements	connexion à l'application Web après authentification	Requirement	8	Sprint 2
Resp. technique	soit utiliser l'interface CRI pour permettre l'authentification des administrateurs et professeurs	d'éviter de créer une source supplémentaire de gestion d'authentification		Non Fonction.	8	Sprint 2

Organization of the Dev. Team : Team charter



Evidence mais encore peut-on le préciser pour qu'une équipe fonctionne bien, il faut que quelques règles soient adoptées par tous. Pour des équipes importantes, on peut établir formellement un Team Charter qui fixe ces règles. Quoiqu'il en soit, il conviendra de discuter ensemble de quelques sujets pour organiser le travail et les thèmes peuvent être :

- Roles and responsibilities :

Le Product Owner

Le Product Owner est responsable de maximiser la valeur du produit résultant du travail de l'équipe de développement. Le Product Owner est le seul responsable de la gestion du Product Backlog.

Le Scrum Master

Le Scrum Master est chargé de promouvoir et supporter Scrum. Les Scrum Masters remplissent leur rôle en aidant tout le monde à comprendre la théorie, les pratiques, les règles et les valeurs de Scrum. Le Scrum Master est un leader-serviteur de l'équipe Scrum. Le Scrum Master assiste les personnes externes à l'équipe Scrum pour identifier quelles sont les interactions bénéfiques avec elle. Le Scrum Master aide tout le monde à adapter leurs

interactions avec l'équipe Scrum pour maximiser la valeur créée par cette équipe.

L'équipe de Développement

L'équipe de développement se compose de professionnels qui fournissent un incrément « Fini » potentiellement publiable (Releasable) à la fin de chaque Sprint.

- Values & behaviours: assez americain mais pourquoi pas en discuter
- meeting details: when , how , documentation ...
- technical practices:
 - repository (GITHUB, GITLAB...).... ,
 - Tests tools
 - ...
- communication tool:
 - Interpersonal communication tools (slack , skype..)
 - Activity tracking (Trello, GITHUB Project , JIRA...) ;

Product backlog is now splitted in
Sprints and Releases
Let us start our first Sprint

The second horizon for Agile
planning : **Sprint Planning**

Each Sprint Starts with:

SPRINT PLANNING



The start of a Sprint will be achieved by a Sprint Planning meeting that aims to define the work to be done by the team to assign it to plan it over time. This meeting includes the entire team and optionally the Product Owner in order to clarify expectations. The Sprint Planning meeting is quite long but max 8hours for a one-month sprint.

How it works:

- Explanation of stories that are in the sprint to the team
- Defining the conditions for a job to be considered finished?? So how is the realization tested?
- Review and confirmation of points awarded to the user story by Dev's team.
- For each user story: We cut into task to carry out the User Story
 - Cutting the work that will be assigned to each developer of the team:
 - Possibly estimating the duration of the load assignment to team members (predict tests, bug fixes)
 - Ordering work (what you do first, and then as well as managing dependancies)
- Review of the consistency between the Us weighting points and the dev team's sprint work plan: Possible adjustment : adding or removing user story from Sprint

Backlog

- Putting in a Scrum Board TO DO list of the US decomposed in task: Scrum Board can be supported by collaboration tool GITHUB PROJET, Trello ...

At the end of Sprint Planning, we know the features we will achieve, how to test them, how we organize the work of the dev team to achieve the result.

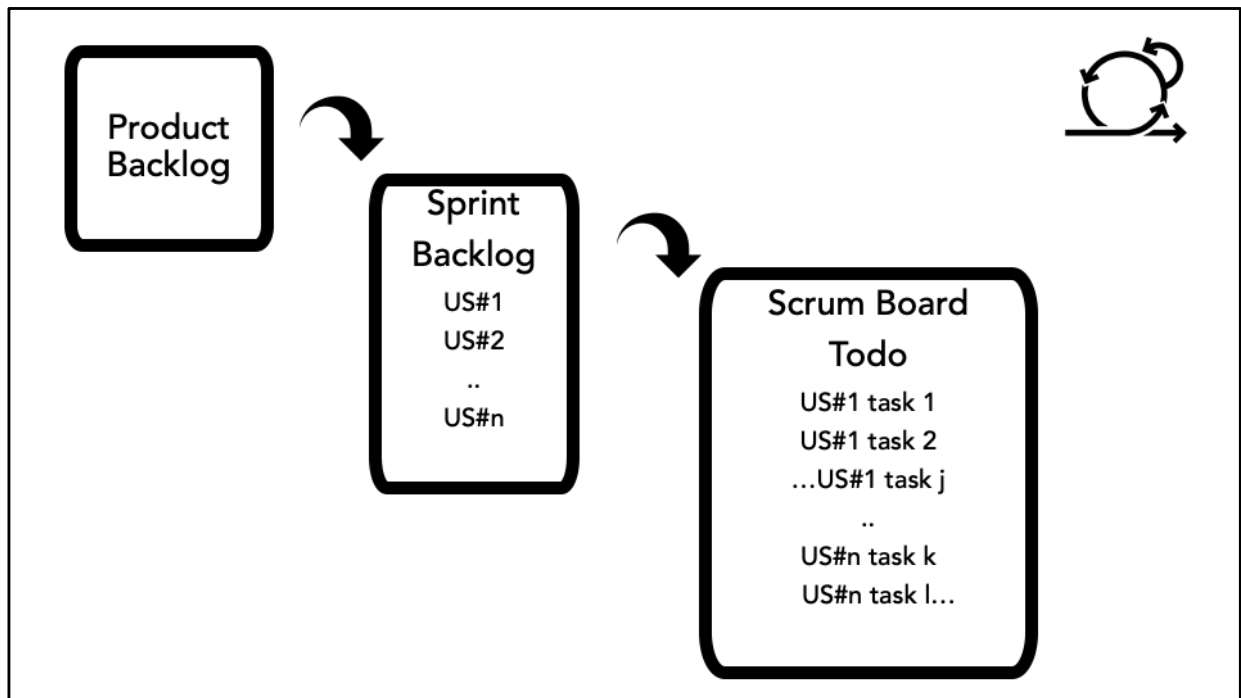
Le démarrage d'un Sprint va être réalisé par une réunion de Sprint Planning qui vise à définir le travail à réaliser par l'équipe à l'affecter à le planifier dans le temps.

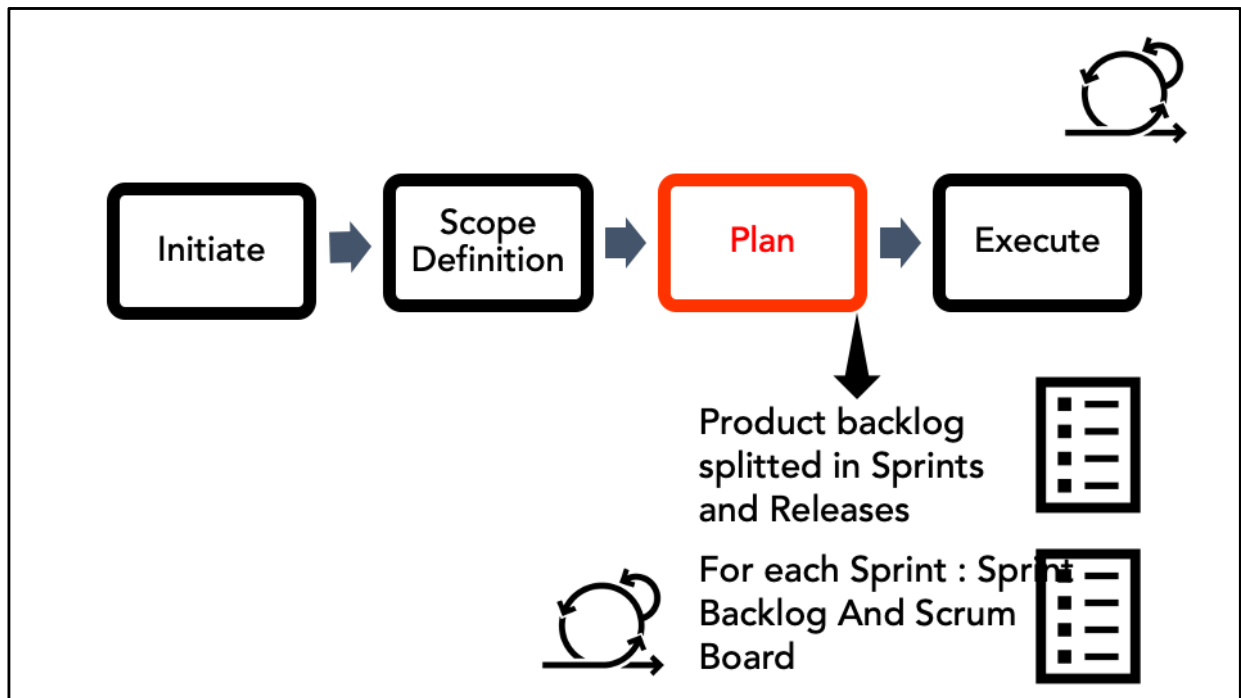
Cette réunion comprend l'ensemble de l'équipe et optionnellement le Product Owner pour clarifier les attentes .

La réunion de Sprint planning est assez longue mais max 8h pour un sprint d'un mois.

Le déroulement :

- *Explication à l'équipe de dev des stories qui sont dans le sprint*
- *Définition des conditions pour qu'un travail soit considéré comme terminé /fini ?
Donc Comment est testée la réalisation ?*
- *Revue et confirmation des points attribués à la user Story par l'équipe de Dev.*
- *Pour chaque user story : On découpe en activité à réaliser la user story Découpage des travaux qui vont être affectés à chaque développeur de l'équipe :*
 - *Possiblement estimation de la durée de réalisation des tâches*
 - *affectation aux membres de l'équipe (prévoir les tests, les corrections de bug)*
 - *Ordonnancement des travaux (ce que l'on fait en premier, puis ensuite et gestion des dépendances)*
- *Revue de la cohérence entre la pondération des US et le plan de travail de l'équipe de dev sur le sprint : Ajustement éventuel ajout ou suppression de user story du Sprint Backlog*
- *Mise dans un Scrum Board colonne TO DO des US décomposée en tâche: Scrum peut être supporté par outil de collaboration GITHUB PROJET, Trello ...*
- *En fin de Sprint Planning, on sait les fonctionnalités que l'on va réaliser, comment les tester, comment on organise le travail de l'équipe de dev pour arriver au résultat.*





At the end of the Planning Phase we have build a product backlog ordered weighted , splitted in sprints and realeases.

For each sprint we will perform planning activities to organize the scope of the specific sprint into tasks that will be followed in a scrum board

À la fin de la phase de planification, nous avons construit un Product backlog ordonné; pondéré, divisé en sprints et realeases.

Pour chaque sprint, nous effectuerons des activités de planification pour organiser la perimètre du sprint spécifique en tâches qui seront suivies dans un Scrum board par l'équipe de Dev.

Presence Project : Sprint 1 Planning ?



Recall Presence Project context and Ask to students to define priorities to user stories

En tant que Resp. d'évènement je veux
lancer la session de présence sur un
événement afin de pouvoir proposer le
badgeage de leur carte aux étudiants.

Regardons cette user stories et identifions les taches qui peuvent en découler en terme de

- Spec fonctionnelle
- Dev. Application mobile
- Dev. Back end
- Test

	Qui	Charge estimée	Pour Quand
Modification Mockup I/F appli mobile and valid client (workshop)	DEV1	0,5	Day 8
Dev Startup screen	DEV 1	1	Day 10
Dev Pincode selection	DEV 1	2	Day 12
Dev Backend endpoint	DEV 2	1	Day 11
Tests	DEV1 & DEV2	2	Day 14
Modification of Documentation	DEV1 & DEV2	0,5	Day 15



Put in Scrum Board (e.g. Trello)
with User Story identification

Next horizon of Agile planning is:

Daily Scrum (see module about execution)

Agile Planification Summary: Release Planning

➡ Sprint Planning

➡ Daily scrum