

# 420-KBH-LG, PROJET D'INTÉGRATION

Saliha Yacoub

## Objectifs de cette séance:

- › Rappels Backlog du produit
- › User-stories: la propriété INVEST
- › Backlog de sprint
- › Plan de release.

# Le backlog du produit

Id_story	Énoncé	Test d'acceptation	Détails	Priorité	Estimation
1	En tant que client de la banque , je veux me connecter au site pour choisir une opération bancaire	Connexion réussie  Menu d'option affiché	1 – réaliser le formulaire de connexion  2- afficher les option du menu	M	4 heures pour une personne.  10 heure pour une personne.
2	En tant que client, je souhaite consulter mes transaction du mois	Selon la période, la liste des transactions est affichée	1- choisir l'option: voir « mes transactions »  2- choisir la période pour les transactions	M	5  5
3	En tant que client je souhaite réaliser une opération bancaire	N'est pas une bonne user-story !!!			

# Planifier une Release

- › Étape 1, définir la date fin. → le 02 décembre.
- › Étape 2, estimer les stories → fait dans la backlog
- › Étape 3, définir la taille d'un sprint → un sprint a trois semaines.
- › Étape 4 estimer la capacité de l'équipe → pas évident. On pourrait se baser sur le temps (en heures) pour un sprint , mais encore là pas facile à déterminer. Le plus important est que l'équipe soit motivée

# Planifier une Release

## Étape 5: Produire un plan de Release

- › Après avoir fait les étapes précédentes produire un plan de release devient facile, un jeu d'enfant. On procède comme suit :
  - On prend le backlog du produit priorisé et estimé.
  - On commence par le premier sprint de la release. On y associe les stories en commençant par les prioritaires
  - On continue dans ce sprint en additionnant la taille en points de stories jusqu'à atteindre la capacité de l'équipe
  - Quand on y arrive, on passe au sprint suivant.
- › Et Si on ne tombe pas exactement sur la capacité de l'équipe à l'intérieur d'un sprint ?
  - › On va légèrement au dessus , ou en dessous ou une story un peu moins prioritaire et qui rentre dans le sprint.
- › **Garder du « lousse »**

# Planifier une release

Sprint 1 :

M,M,M,M,M,  
M,M,S,S ,C

Sprint 2 :

M,M,M,S,S,S,  
S,S,C,C

Sprint 3 :

M, S, S, S,  
S, C, C, C,  
C, C

- › Dans le sprint 1, en principe on devrait retrouver plus de stories priorisées M que S ou C. Parfois un story priorisée C complète bien le sprint.
- › Ce n'est pas rare, que des stories priorisée M se retrouvent dans le dernier sprint. Par contre il faut faire en sorte de les terminer en premier.
- › Si on suit la méthode qui consiste à remplir le sprint d'abord par les M, et on applique la même règle aux sprint suivant, en principe votre dernier sprint ne contiendrait pas trop de stories M
- › Attention: S ou C, ne veut pas dire abandonnée. (Exemple: tourner la roue pour le jeu Trivial Crack).

# Planifier une release.

Concrètement :

Le cours de 420-KBH-LG, est pondéré (1-7-6) ce qui veut dire que vous avez 14h par semaine et par étudiant.

Pour une équipe de 4, si vous avez estimé en temps vos user-stories, alors pour **le sprint 1**, vous devez avoir des stories totalisant:

- › 168 heures pour l'équipe
- › 42 heures pour un étudiant. (pour le sprint 1)

On vous demande de **laisser du lousse**, car il y'aura certainement des imprévus.

Je pense que si chaque étudiant met 36 à 38 heures, par sprint , l'équipe va arriver et surtout pas essoufflée

# Planifier un sprint: sprint 1

Après avoir planifié votre release, en principe vous avez les backlog de TOUS les sprints. On vous demande le backlog du sprint 1.

Après le backlog, il faudra indiquer les personnes (les étudiants membres de l'équipe) qui devront s'occuper de la réalisation des user-stories

Exemple:

Raphaël et Thomas réaliseront la story 1

Félix réalisera la story 2

etc....

Vous l'avez compris: On ne peut pas vous demander le backlog des autres sprints, car il est possible que leurs contenus soient changés après la revue du sprint 1





CONCLUSION



QUESTIONS ??