
	Titre du document : Titre du document Coclico
	Référence : L3-3-3_ModeleProjetScrumForge
	Version du 10/10/2011

## Livrable 3.3.1

### Modèle de projet Scrum pour les forges


<b>Livrable du au titre du projet</b>	COCLICO
<b>Lot</b>	3
<b>Tache</b>	3.3
<b>Livrable</b>	L3.3.3 Modèle de projet Scrum pour les forges

<b>Rédacteur(s)</b>	<b>Vérificateur(s)</b>	<b>Approbateur(s)</b>
Séverine Rambaud		

	Titre du document : Titre du document Coclico
	Référence : L3-3-3_ModeleProjetScrumForge
	Version du 10/10/2011


Documents applicables
Annexe technique au projet COCLICO

Documents de références (pour information)

	Titre du document : Titre du document Coclico
	Référence : L3-3-3_ModeleProjetScrumForge
	Version du 10/10/2011


### Gestion des versions

N° de version	Date	Auteurs	Modification apportées
1	10/10/2011	Séverine Rambaud	Version initiale du document

	Titre du document : Titre du document Coclico
	Référence : L3-3-3_ModeleProjetScrumForge
	Version du 10/10/2011

## Sommaire

1	Introduction .....	5
1.1	Scrum & Kanban : 2 outils pour gérer le processus .....	5
1.1.1	Scrum.....	5
1.1.2	Kanban.....	6
1.2	Motivations : pourquoi intégrer un outil agile dans une forge logicielle?.....	7
2	Coclico – Construction d’un nouveau dashboard.....	9
2.1	Contexte.....	9
2.2	Les champs d’un tracker.....	10
2.3	Création de dashboard .....	10
2.3.1	Définition du dashboard .....	10
2.3.2	Cas simple – un tracker .....	11
2.3.3	Cas multi-tracker.....	14
2.3.4	Définition des postits.....	20
2.3.5	Choix des critères de filtrage .....	21
2.4.1	Dashboard final.....	24
2.4.2	Birdview .....	24
2.4.3	Affichage des critères de filtrage .....	25

	Titre du document : Titre du document Coclico
	Référence : L3-3-3_ModeleProjetScrumForge
	Version du 10/10/2011

# 1 Introduction


## 1.1 Scrum & Kanban : 2 outils pour gérer le processus

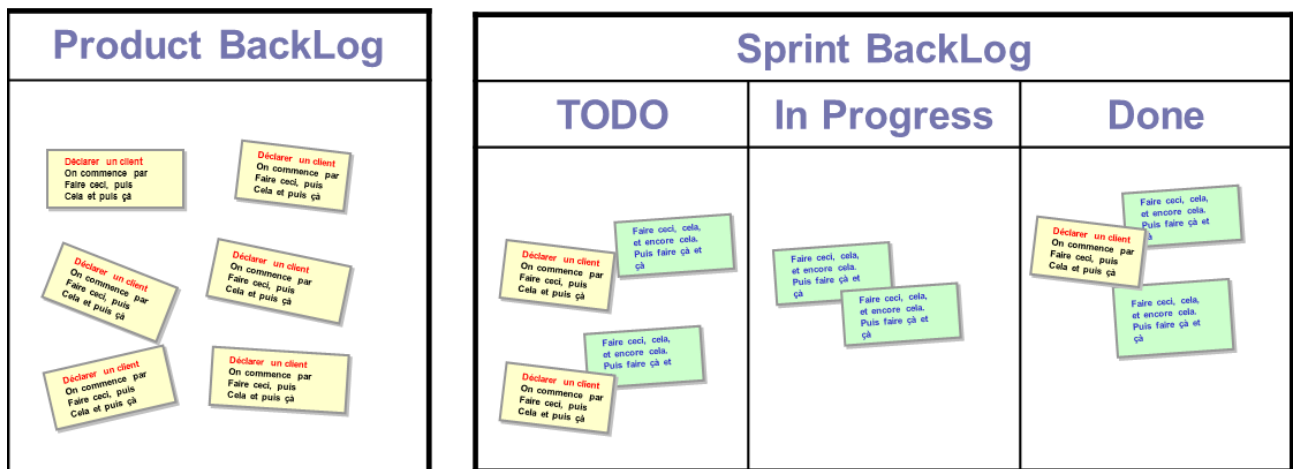
Les méthodologies agiles ont révolutionné l'ingénierie logicielle ces dernières années en modifiant radicalement les processus de développement logiciel d'un point de vue humain et organisationnel. Ces nouvelles méthodes sont dites itératives, puisqu'elles impliquent le découpage d'un projet en plusieurs sous-projets (souvent appelés itérations ou « sprints »), favorisant ainsi les échanges et l'adéquation du produit développé avec les besoins du client.

Scrum & Kanban sont deux outils agiles pour gérer le processus de développement. Comme nous le verrons ci-dessous, il existe de nombreux points communs entre les 2 approches, dont le fait de chercher à limiter le WIP (Work In Progress) et de représenter l'avancement de façon visuelle.

### 1.1.1 Scrum

**Scrum**, la plus connue de toutes, est basée sur un rythme régulier avec des itérations (sprints), implique de nouveaux rôles (Product Owner, Scrum Master) et se caractérise notamment par des artefacts partagés, des livrables réguliers (à chaque itération) et des indicateurs permettant de piloter les itérations et le projet dans son ensemble. Tous ces indicateurs reposent sur des données que les membres de l'équipe de développement doivent mettre à jour régulièrement : temps passés, estimation des temps restants et statuts d'avancement des tâches. Scrum impose de limiter le nombre d'items (tâches) en tenant compte de la capacité de production d'une équipe sur une itération. Parmi les artefacts liés à Scrum figurent 2 documents qu'on appelle « backlogs » : le Product Backlog et le Sprint Backlog. Le **Product BackLog** est le document qui recense la liste des besoins fonctionnels priorisés. Le **Sprint BackLog** est le document qui contient seulement les tâches de l'équipe pour une itération donnée (il est réinitialisé à chaque sprint).

	Titre du document : Titre du document Coclico
	Référence : L3-3-3_ModeleProjetScrumForge
	Version du 10/10/2011



La notion de sprint ou itération est donc intimement liée à l'existence de ces 2 documents :

- *A la création d'un sprint* : on associe des besoins produits au sprint en tenant compte de la capacité de l'équipe de développement
- *En fin de sprint* : on ferme les besoins produits terminés et on réaffecte les besoins produits non terminés au sprint suivant.

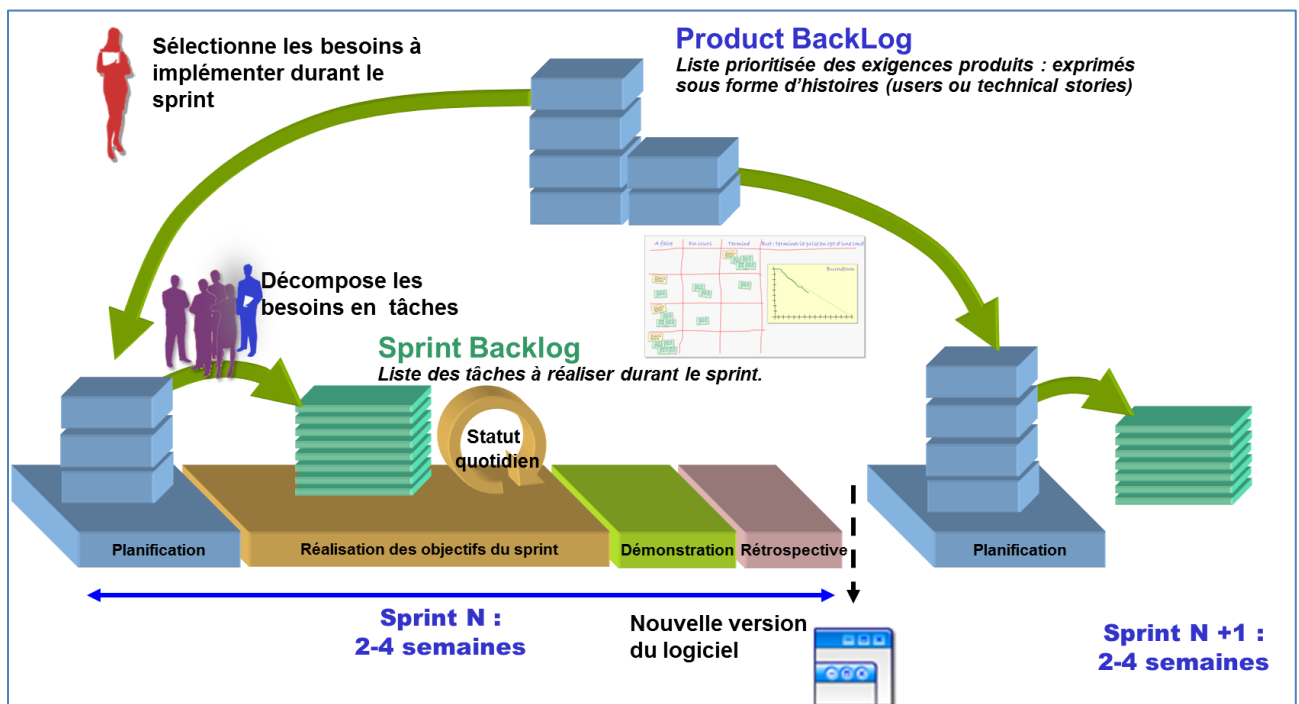



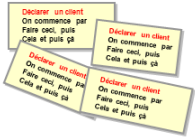

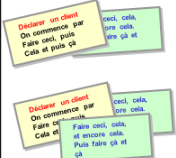

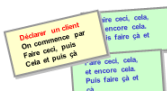
Figure 1- Cycle de vie d'un projet Scrum

### 1.1.2 Kanban

**Kanban** est à l'origine un mode de gestion de flux industriels, dont l'ingénierie logicielle s'est inspirée pour en faire une méthodologie agile épurée basée sur 2 principes fondamentaux : la visualisation des cycles de développement par tâches divisées en étapes successives d'un

	Titre du document : Titre du document Coclico
	Référence : L3-3-3_ModeleProjetScrumForge
	Version du 10/10/2011

workflow (ce workflow est défini en début de projet - le plus simpliste contient en général 3 étapes : « A faire », « En cours » et « Terminé ») et l'optimisation des cycles via le calcul du temps moyen qui est nécessaire pour traiter complètement une tâche. On cherche par ailleurs à limiter le nombre d'items que l'on peut traiter à chaque phase du flux.

Backlog	Selected 2	In Progress 3	Done 2	Deploy 1	Live
					

Kanban est donc basé sur un seul tableau de visualisation permettant de voir le flux global du besoin, de sa définition à sa mise en production. La capacité est limitée pour chaque phase par un nombre d'items (besoins) maximum, et le tableau n'est jamais réinitialisé.


## 1.2 Motivations : pourquoi intégrer un outil agile dans une forge logicielle?

Une forge logicielle permet en général de centraliser toutes les actions faites sur un projet (gestionnaire de sources, documentation, wiki, ...), d'historiser toutes les opérations et de gérer les utilisateurs de façon centralisée.


En revanche, une forge logicielle ne permet pas :

- De conserver le lien entre les différentes actions (les opérations des utilisateurs sont associées à un projet, mais il n'y a pas de lien entre les actions et le besoin auxquelles elles sont associées). Il est donc difficile d'avoir une vue globale sur l'avancement du projet et sur la prise en compte d'un besoin qui nécessite plusieurs types de tâches (documentation, code, tests, déploiement, ...)
- De gérer visuellement l'avancement des besoins ou des tâches.
- De limiter le nombre de tâches en cours (« Work In Progress ») en fonction de la capacité de l'équipe. Par conséquent, il arrive inévitablement que beaucoup de tâches ou besoins soient marqués en cours, mais en réalité leur nombre est bien supérieur à la capacité de réalisation de l'équipe. Ils restent donc en cours pendant une très longue durée.

En résumé, les forges n'intègrent en général aucun des bénéfices apportés par les méthodologies et outils agiles. Il existe de nombreuses applications web ou standalone permettant d'implémenter les méthodologies agiles dans un projet ([IceScrum](#) ou [theSCRUM](#) par exemple), mais malheureusement ces applications sont totalement dé-corrélées des forges et il n'est pas possible (ou très difficile) de les interfacier pour les rendre complémentaires.

	Titre du document : Titre du document Coclico
	Référence : L3-3-3_ModeleProjetScrumForge
	Version du 10/10/2011

L'un des enjeux du projet COCLICO réside donc dans la prise en compte des méthodologies agiles directement au sein des forges logicielles.

	Titre du document : Titre du document Coclico
	Référence : L3-3-3_ModeleProjetScrumForge
	Version du 10/10/2011

## 2 Coclico – Construction d’un nouveau dashboard

### 2.1 Contexte

Pour faciliter la gestion d’un projet de développement logiciel, il est nécessaire d’utiliser certains outils de suivi du projet, notamment pour pouvoir suivre les tâches à implémenter ou les bugs à résoudre. S’il s’agit d’une méthode de développement « Agile », la gestion des sprints et des stories peut être intéressante. Quand un membre du projet définit une nouvelle tâche dans son projet, il crée un nouveau ticket qui correspond à cette tâche. Il entre le nom de la tâche, la description, son statut etc. Les tickets sont regroupés dans des trackers. Il existe donc le tracker des tâches, le tracker de sprints, le tracker des bugs etc.

Un dashboard est une grille (tableau) où les tickets d’un tracker pour un certain projet sont affichés sous la forme de postits. Le but du dashboard est de pouvoir facilement visualiser les tickets et les modifier, en utilisant le drag and drop pour les déplacer d’une cellule à l’autre du dashboard.

Ce document définit comment construire un dashboard de centralisation de tickets ainsi que la customisation du postit et le choix des critères de filtrage des postits sur un dashboard.

Dans la Figure 1 on a un exemple de dashboard pour un tracker, dans lequel les postits représentent des tâches, ordonnées par leurs types (ligne) et statuts (colonne). Si par exemple on déplace un postit de la case Dev/ToDo dans la case Dev/Done, cela correspond au fait que la tâche a été finalisée et que son Status est modifié en Done. Donc un déplacement d’un postit d’une case à une autre correspond à la modification du ticket correspondant.





		Tache Status		
		In progress	Todo	Done
Tache Type	Dev			
	QA			
	Management			

Figure 2 Dashboard exemple

Les tickets peuvent être filtrés par des critères de filtrage : par exemple si on ne veut que les

	Titre du document : Titre du document Coclico
	Référence : L3-3-3_ModeleProjetScrumForge
	Version du 10/10/2011

tickets qui nécessitent un temps de résolution de moins de 5 heures, à côté du dashboard on aura un outil qui permettra ce type de filtrage (voir chapitre 3.3).

## 2.2 Les champs d'un tracker

Chaque ticket, de n'importe quel type (tâche, story, etc.) est caractérisé par certaines propriétés : son nom, la date de création, son auteur, éventuellement une référence vers d'autres tickets etc. Ces propriétés représentent les champs d'un tracker. Pour un tracker, les types des champs possibles actuellement dans Coclico sont :

- String – ex. le nom du ticket
- Text – ex. la description du ticket, un commentaire sur le ticket etc.
- Int – ex. l'id du ticket
- Date – ex. la date de soumission d'un ticket, la date de fin d'un sprint etc.
- Number - ex. le nombre d'heures estimées pour la résolution d'une tâche
- Url – ex. l'url pour accéder à la description d'un sprint
- Enuméré – un type qui peut avoir plusieurs valeurs ; ce type est utile pour pouvoir définir par exemple le statut des tâches (in progress, todo, done), la sévérité (minor/major/critical) etc.
- tracker-ticket-ref - référence vers un autre ticket ; s'il s'agit d'une tâche, ce champ peut représenter une autre tâche dont celle ci dépend.
- forge-ticket-ref - référence vers un autre tracker ; ex. si on a un tracker de stories et un tracker de tâches, ce champ fait référence à la story à laquelle appartient la tâche.
- Member –référence vers un membre du projet ; ex. l'auteur du ticket, la personne au quelle le ticket est attribué.


## 2.3 Création de dashboard

La création d'un dashboard est constituée de trois étapes :

- définition du dashboard : suppose d'entrer un nom et une description du dashboard ainsi que définir quelles seront les lignes et les colonnes du dashboard. Pour cela il faut choisir quel champ du tracker constitue la ligne et quel la colonne.
- définition du postit : l'utilisateur pourra configurer les informations qui seront affichées sur son postit.
- définition des critères de filtrage du dashboard : l'utilisateur choisit des critères de filtrage pour les postits ; avec l'aide de ces filtres, l'utilisateur affiche sur son dashboard seulement les postits qui remplissent ses critères.

Les étapes de la création du dashboard sont détaillées en suite.

### 2.3.1 Définition du dashboard

	Titre du document : Titre du document Coclico
	Référence : L3-3-3_ModeleProjetScrumForge
	Version du 10/10/2011

Pour la définition du dashboard, les pas à suivre sont :

1. L'utilisateur entre le nom et une description pour son nouveau dashboard.
2. L'utilisateur choisit le tracker depuis la liste des trackers disponibles

## 2.3.2 Cas simple – un tracker

Pour la création d'un dashboard il est nécessaire de savoir selon quels critères seront regroupés les tickets. Par exemple, si on veut afficher les tickets du tracker Tâches et les organiser par leur Status et leur Type, on choisira comme colonne pour notre dashboard le champ Status du tracker Tâches et pour la ligne, le champ Type du même tracker. On considère donc l'exemple suivant :

**Exemple : L'utilisateur veut afficher les tâches en fonction de leur statut et leur type. Le tracker Tâches contient les informations suivantes :**


Tableau 1 - Tickets dans le tracker Tâches

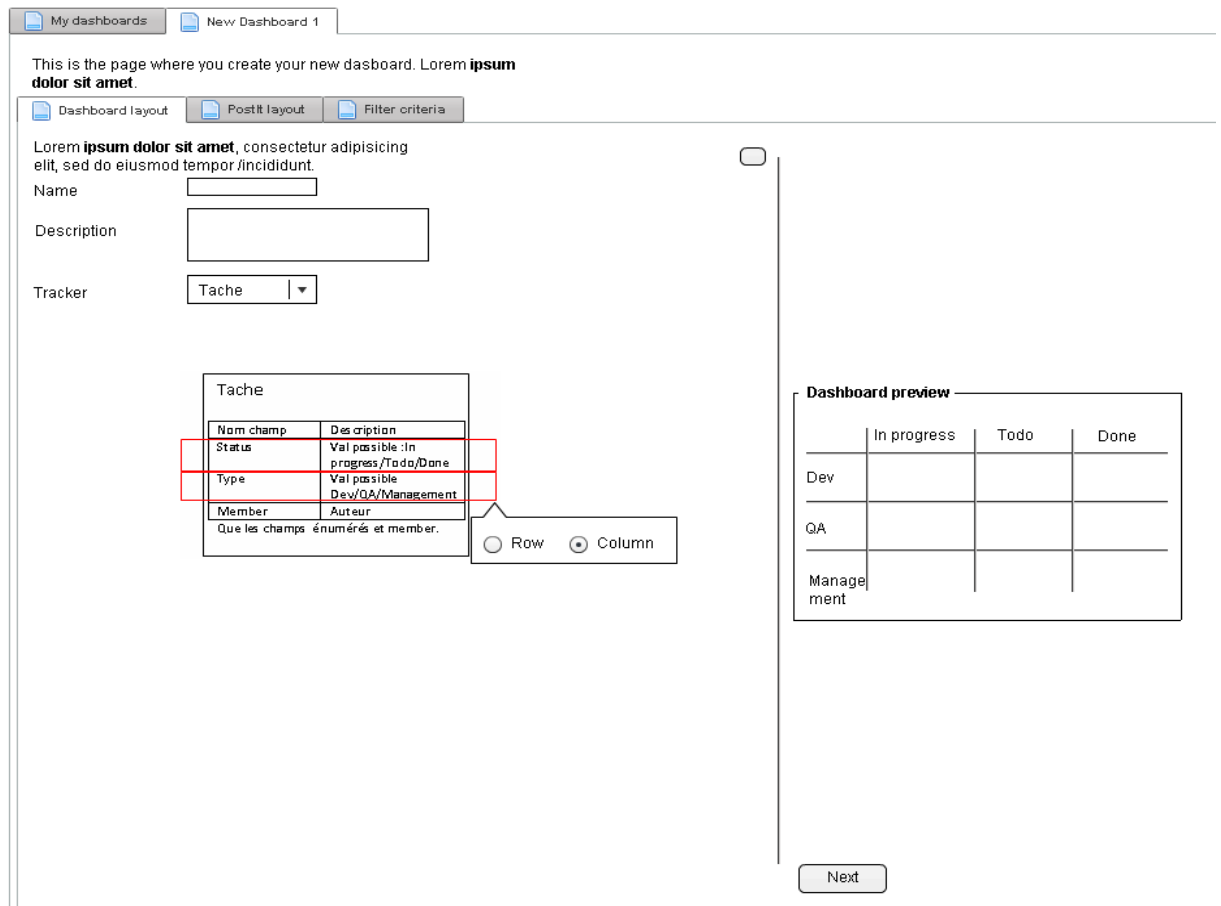
Id	Name	Description	Type	Status
1	Change mdp	Implémenter le formulaire de changement de mot de passe.	Dev	In progress
2	Ajout produit	Implémenter le formulaire d'ajout de nouveau produit (prix, photo...)	Dev	Todo
3	Mail notif mdp	Implémenter l'envoi d'un mail de notification à l'heure du changement de mdp.	Dev	Todo
4	Produit par critère	Implémenter le formulaire de recherche produit par critère.	Dev	In progress
5	Test mise à jour bd	Tester la mise à jour de la BD avec les produits.	QA	Done
6	Réunion conception bd	Etablir les tables de la BD	Management	Done
7	Documentation	Commenter la nouvelle API d'interaction avec la BD	QA	In progress
8	Réunion conception IHM ajout produit	Concevoir l'IHM d'ajout produit	Management	Done

L'écran correspondant à cet exemple se trouve dans la figure suivante (Figure 2). Les étapes à suivre pour cet exemple sont :

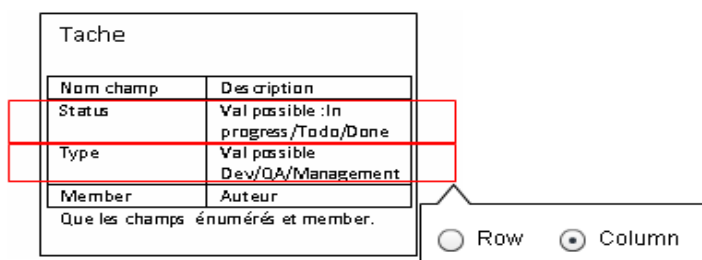
1. Après le choix du tracker Tâches (par la combo-box Tracker), un tableau avec les champs énumérés et membre du tracker choisi est affiché.
2. Choix de la ligne et de la colonne du dashboard. L'utilisateur peut sélectionner deux champs du tableau qui représente le tracker, en cliquant sur la ligne correspondante de ce tableau.

Figure 3 : Création dashboard avec des infos d'un seul tracker

	Titre du document : Titre du document Coclico
	Référence : L3-3-3_ModeleProjetScrumForge
	Version du 10/10/2011



3. Pour chaque champ sélectionné, un floating panel apparait qui permet de spécifier si le champ sera la ligne ou la colonne du dashboard (Figure 3).




4. L'utilisateur peut voir à droite de l'écran un preview de son dashboard après la sélection de la ligne et de la colonne (Figure 2 – Dashboard preview).

5. Le dashboard obtenu est le suivant :

	Tache Status		
	In progress	Done	Todo

Figure 4 : Floating pane pour choisir ligne/colonne – cas d'un seul tracker

	Titre du document : Titre du document Coclico
	Référence : L3-3-3_ModeleProjetScrumForge
	Version du 10/10/2011





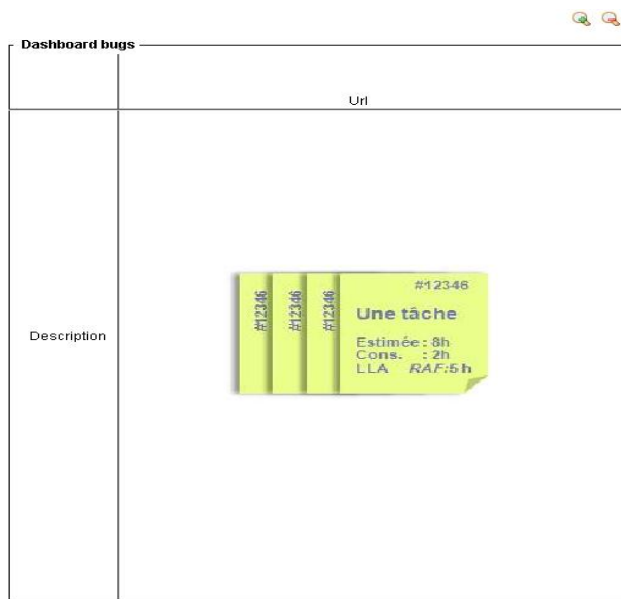
Tache Type	Management			
	QA			
	Dev			

Figure 5 Dashboard généré à partir d'un seul tracker

### Observation


Les deux champs choisis Status et Type sont énumérés (avec plusieurs valeurs possibles). En utilisant ces deux champs, le dashboard généré a plusieurs lignes et plusieurs colonnes. Si on choisit un champ qui n'est pas énuméré pour la ligne du dashboard (exemple Url/Description), cela crée un dashboard avec une seule ligne/colonne, qui n'apporte aucune valeur ajoutée à la visualisation des tickets (Figure 5).



The image shows a dashboard titled "Dashboard bugs" with a table structure. The table has two columns: "Url" and "Description". The "Description" cell contains a single task card for "#12346" with details: "Une tâche", "Estimée: 8h", "Cons.: 2h", and "LLA RAF:5h".

Figure 6 : Dashboard généré à partir des champs qui ne sont pas énumérés ou membre

En choisissant par exemple pour la ligne, le champ Submitted by (de type membre) et pour la colonne, le champ Status (de type énuméré), le dashboard généré sera celui de la Figure 6.

	Titre du document : Titre du document Coclico
	Référence : L3-3-3_ModeleProjetScrumForge
	Version du 10/10/2011

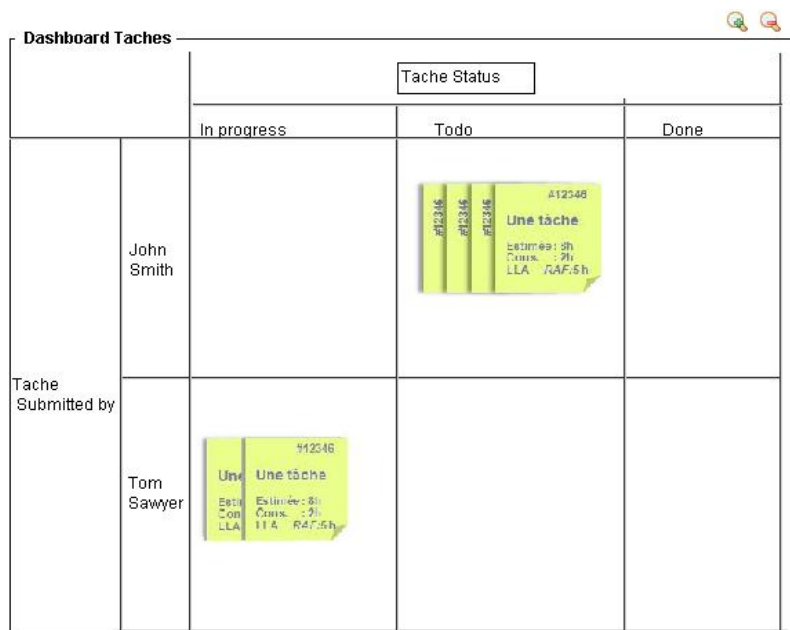


Figure 7 : Dashboard généré à partir d'un champ énuméré et un champ membre

On constate donc que, pour créer un dashboard à partir d'un seul tracker, les champs utiles sont celles de type énuméré et membre. Seuls ces champs vont être mis à la disposition de l'utilisateur pour le choix de la ligne et la colonne du dashboard.

### 2.3.3 Cas multi-tracker

Le cas le plus commun est de vouloir afficher sur un dashboard des informations à partir d'un seul tracker. Ce cas a été présenté dans la section 3.1.1.


L'utilisateur pourrait souhaiter afficher des informations à partir de plusieurs trackers, en suivant les références du tracker choisi vers les autres trackers. La démarche à suivre est différente dans ce cas.

**Exemple : L'utilisateur veut créer un dashboard pour afficher le statut des tâches par story.**

La figure suivante représente les relations entre plusieurs trackers. Dans le cas d'une gestion de projet « Agile », le projet est découpé dans des sprints (tracker des Sprints) ; chaque sprint contient plusieurs stories (tracker des Stories) et chaque tâche est rattachée à une story.

#### Tracker Tâches

Id	Name	Description	Type	Status	StoryRef
1	Change mdp	Implémenter le formulaire de changement de mot de passe.	Dev	Todo	52
2	Ajout produit	Implémenter le formulaire d'ajout de nouveau produit (prix, photo...)	Dev	Todo	85

	Titre du document : Titre du document Coclico
	Référence : L3-3-3_ModeleProjetScrumForge
	Version du 10/10/2011

3	Mail notif mdp	Implémenter l'envoi d'un mail de notification.	Dev	In progress	52
4	Produit par critère	Implémenter le formulaire de recherche produit par critère.	Dev	In progress	33
5	Mise à jour bd	Tester la mise à jour de la BD avec les produits.	QA	Done	85
6	Réunion conception bd	Etablir les tables de la BD	Management	Done	85
7	Documentation	Commenter le nouveau API d'interaction avec la BD	QA	In progress	85
8	Réunion conception IHM ajout produit	Concevoir l'IHM d'ajout produit	Management	Done	85

### Tracker Stories


Id	Name	ShortName	Description	SprintRef	Priority	Status...
52	Gestion modif mdp	Mdp	Comme utilisateur standard je veux changer mon mdp password.	10047	2	In progress
85	Gestion nouveau produit	New prod	Comme admin je veux ajouter un nouveau produit.	10785	1	Done
33	Gestion recherche produit	Search prod	Comme utilisateur standard je veux chercher un produit par critère.	10047	1	Ready

### Tracker Sprints

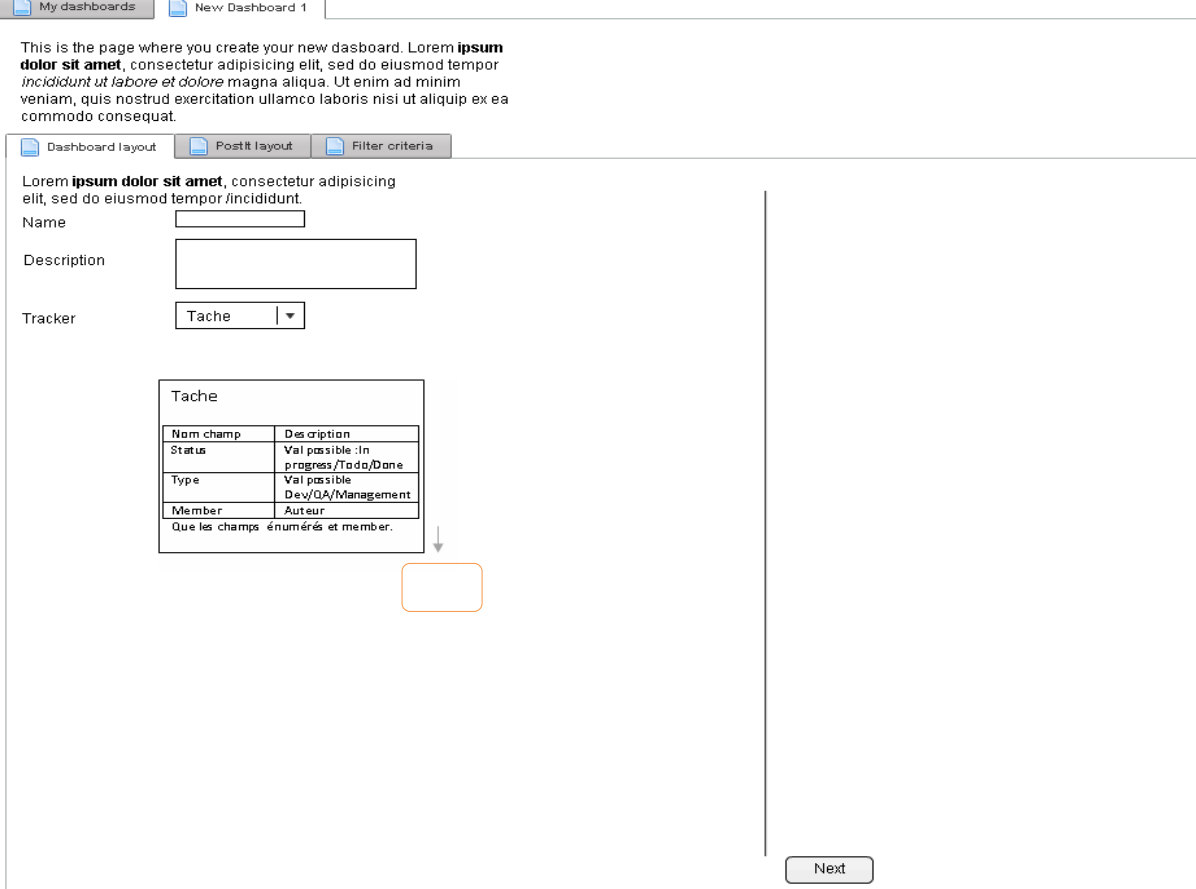
Id	Name	Description	StartDate	EndDate	Status...
10047	Sprint 1 : Développer IHM user standard				In progress
10785	Sprint2 : Développement IHM admin				Done

Figure 8 : Relations entre trackers

Le dashboard à construire nécessite des informations de deux trackers, le tracker des Tâches et un autre tracker, le tracker de Stories. Les étapes à suivre sont :

	Titre du document : Titre du document Coclico
	Référence : L3-3-3_ModeleProjetScrumForge
	Version du 10/10/2011

- Après le choix du tracker Tâches, un tableau avec les champs énumérés et member du tracker choisi est affiché. A coté de ce tableau, une flèche désactivée est affichée (Figure 8).



This is the page where you create your new dashboard. Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

Dashboard layout | PostIt layout | Filter criteria

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor /incidunt.  
 Name:   
 Description:   
 Tracker: Tache


Nom champ	Description
Status	Val possible :In progress /Tada/Done
Type	Val possible Dev/QA/Management
Member	Auteur

Que les champs énumérés et member.

Next

Figure 9 : Ecran de choix du premier champ du tracker Tâches

- Pour afficher les tickets du tracker Tâches, l'utilisateur est obligé de choisir un champ du tracker Tâches comme ligne ou colonne de son dashboard. L'utilisateur clique sur le champ Status du tableau qui représente le tracker Tâches.
- Le floating pane qui permet de choisir si le champ sélectionné est une ligne ou une colonne du dashboard est affichée.
- En même temps, le clic sur le champ active la flèche qui se trouve en bas du tableau Tâche (Figure 9 entouré en vert). On observe aussi le preview du dashboard à droite, entouré en orange.
- En appuyant sur la flèche qui vient d'être activée, l'utilisateur affichera un graphe avec les relations entre les trackers attachés au tracker Tâche (dans ce cas, le tracker Story). Cela permettra à l'utilisateur de choisir l'autre champ depuis le tracker Story (Figure 10). On observe, à côté du tableau qui représente le tracker Story une autre flèche qui permet d'étendre le graphe avec les trackers reliés au tracker Story. Ce n'est pas le cas pour le dashboard qu'on veut construire. On observe aussi une petite croix rouge à côté du tableau Story (Figure 10)

	Titre du document : Titre du document Coclico
	Référence : L3-3-3_ModeleProjetScrumForge
	Version du 10/10/2011

qui permet de cacher ce tableau pour revenir à la construction du dashboard à partir d'un seul tracker.



Titre du document : Titre du document Coclico

Référence : L3-3-3\_ModeleProjetScrumForge

Version du 10/10/2011

My dashboards New Dashboard 1

This is the page where you create your new dashboard. Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

Dashboard layout PostIt layout Filter criteria

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt.

Name

Description

Tracker Tache

Nom champ	Description
Status	Val possible :In progress/ToDo/Done
Type	Val possible /Management

Row  Column

Tous les champs énumérés et member.

Dashboard preview

	In progress	Todo	Done

My dashboards New Dashboard 1

This is the page where you create your new dashboard. Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

Dashboard layout PostIt layout Filter criteria

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt.

Name

Description

Tracker Tache

Nom champ	Description
Status	Val possible :In progress/ToDo/Done
Type	Val possible Dev/QA/Management
Member	Auteur

Que les champs énumérés et member.

Nom champ	Description
Name	Nom d'ustory
Priority	Val possible high/medium/low
Criticality	Val possible Vital/dispensable
Member	Auteur

Tous les champs


Dashboard preview

	Story1	Story2	Story3
In progress			
Todo			
Done			

Next

Next

Figure 10 Choix du premier champ (ligne) du tracker Tâches

	Titre du document : Titre du document Coclico
	Référence : L3-3-3_ModeleProjetScrumForge
	Version du 10/10/2011

**Observation**

Le choix du tracker initial oblige l'utilisateur de choisir au moins un champ (énuméré ou member) de ce tracker comme ligne ou colonne du son dashboard. Le deuxième champ peut être choisi de l'un des trackers associés au tracker initial. Ce deuxième champ peut être de n'importe quel type : String, text, Int, enum, etc (voir section 2 pour tous les types).

8. Pour obtenir un dashboard avec le nom du story en ligne, le champ Name du tableau Story est sélectionné.
9. Quand l'utilisateur sélectionne le champ Name, un floating pane apparait avec deux boutons radio pour lui demander s'il veut le champ sur la ligne ou sur la colonne, en lui proposant, par défaut ce qui n'a pas été défini au niveau du champ dans le référence (Figure 10).
10. A droite du graph, un preview aidera l'utilisateur à voir son nouveau dashboard (Figure 10). Le dashboard généré est celui en dessous (Figure 11).

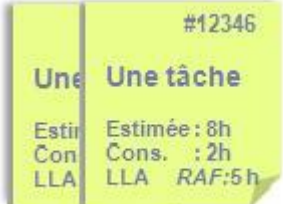



		Tache Status		
		In progress	Todo	Done
Story Name	Story 1			
	Story 2			
	Story 3			

Figure 12 : Dashboard généré pour use-case 2

	Titre du document : Titre du document Coclico
	Référence : L3-3-3_ModeleProjetScrumForge
	Version du 10/10/2011

## 2.3.4 Définition des postits

Après la définition du dashboard, l'utilisateur doit faire le choix des informations qui seront affichées sur le postit. Il a à sa disposition un graphe avec tous les champs des trackers qu'il a utilisé pour construire son dashboard. L'utilisateur peut glisser depuis le graphe les informations qui l'intéressent et les déposer sur un preview de son postit. Le postit modifiera ses dimensions en fonction des informations qu'on souhaite afficher.

On envisage de séparer le postit dans plusieurs zones distinctes pour pouvoir organiser les informations de la manière souhaitée. Une possible séparation du postit dans des zones est représentée dans la Figure 12.

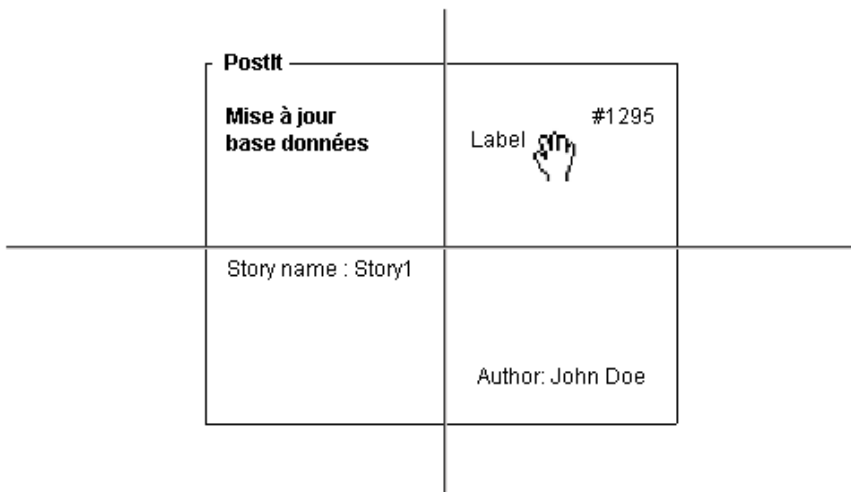



Figure 13 : Zones du postit

Dans la figure suivante on représente l'écran de configuration du postit. A gauche, on a le graphe avec tous les champs qu'on peut glisser-déposer sur le postit à droite (Figure 13).



Titre du document : Titre du document Coclico

Référence : L3-3-3\_ModeleProjetScrumForge

Version du 10/10/2011

My dashboards
New Dashboard 1

This is the page where you create your new dashboard. Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

Dashboard layout
Postit layout
Filter criteria

Configure your postit by dragging and dropping items from the graph onto the area of the ticket you want them to appear. The postit will be smaller or bigger depending on its content. Click Next to choose your Filtering criteria.

Tache

Nom champ	Description
Name	Nom de la tache
ShortName	Nom court
Status	Val possible :In progress/Tada/Done
Type	Val possible Dev/QA/Management
Comment	Commentaire
Member	Auteur

Tous les champs.

Story

Nom champ	Description
Name	Nom d'ustory
ShortName	Nom court
Description	Description du story
Status	Val possible :In progress/Tada/Done
Priority	Val possible high/medium/low
Criticality	Val possible Vital/dispensable
Member	Auteur

...

Tous les champs

**Postit**

Mise à jour base données #1295

Story name : Story1

Author: John Doe

Figure 14 : Ecran de configuration du postit

## 2.3.5 Choix des critères de filtrage


En ce qui concerne les critères de filtrage, la représentation proposée est la suivante :

- Critère énuméré : suite de check-boxes. En cochant un check-box, on choisit d'afficher tous les tickets qui remplissent ce critère.
- Critère de type date : slider avec deux curseurs
- Critère de type int ou number : slider avec deux curseurs
- Critère de type string, texte ou url : text-box de type suggestion de recherche (à discuter)

*Exemple 1* : Pour un critère de filtrage de type date, si on a trois tickets avec les EndDate suivants :

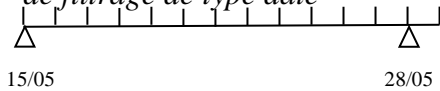
- Ticket1 (EndDate = 15/05/2010)
- Ticket2 (EndDate = 20/05/2010)
- Ticket3 (EndDate = 28/05/2010)

On veut afficher seuls les tickets qui ont une EndDate inférieure à 20/05/2010. Pour filtrer de cette manière, on aura à côté du dashboard un slider avec deux curseurs qui permettra ce type de filtrage. Le slider aura à gauche la EndDate la plus petite et à droite la EndDate la plus grande et deux curseurs qui peuvent être déplacé. Les tickets qui ont une date en dehors des deux curseurs seront filtrés et ne seront pas affichés sur le dashboard. Donc seuls le Ticket1 et le Ticket2 seront

	Titre du document : Titre du document Coclico
	Référence : L3-3-3_ModeleProjetScrumForge
	Version du 10/10/2011

finaleme nt affichés sur le dashboard.

Figure 15 Slider pour un critère de filtrage de type date



Exemple 2 : Pour un critère de type énuméré, représenté par un groupe de check-boxes, en cochant un des check-boxes, seuls les tickets qui remplissent ce critère sont affichés. Par exemple si le dashboard défini affiche les taches et le critère choisit est la priorité de la Story avec les valeurs possibles high/medium/low et que l'on sélectionne la check-box low, seulement les taches qui appartient à des stories de priorité low seront affichées.

Le choix des critères de filtrage se fait indifféremment du fait d'avoir choisi de construire un dashboard à partir d'un seul tracker ou de plusieurs. Pour choisir ces critères, l'utilisateur a à sa disposition le graphe avec tous les trackers en relation avec celui choisit au début. Le graphe utilisé est celui de la Figure 15. On observe que cette fois, tous les champs du tracker Tâches choisi sont disponibles comme possibles critères de filtrage.

L'utilisateur peut sélectionner un ou plusieurs champs des trackers du graphe comme critère. Pour chacun de ces critères de filtrage, l'utilisateur pourra visualiser sur la zone preview de son écran, la représentation de son critère. Par exemple, pour un critère de type date, il aura le slider de la Figure 14, pour le choix d'un intervalle. Les critères de type énuméré seront représentés par une suite de check-boxes. Si les valeurs des critères sont trop nombreuses, le groupe qui contient ce critère aura un scrollbar. La Figure 16 représente l'écran de choix des critères de filtrage.

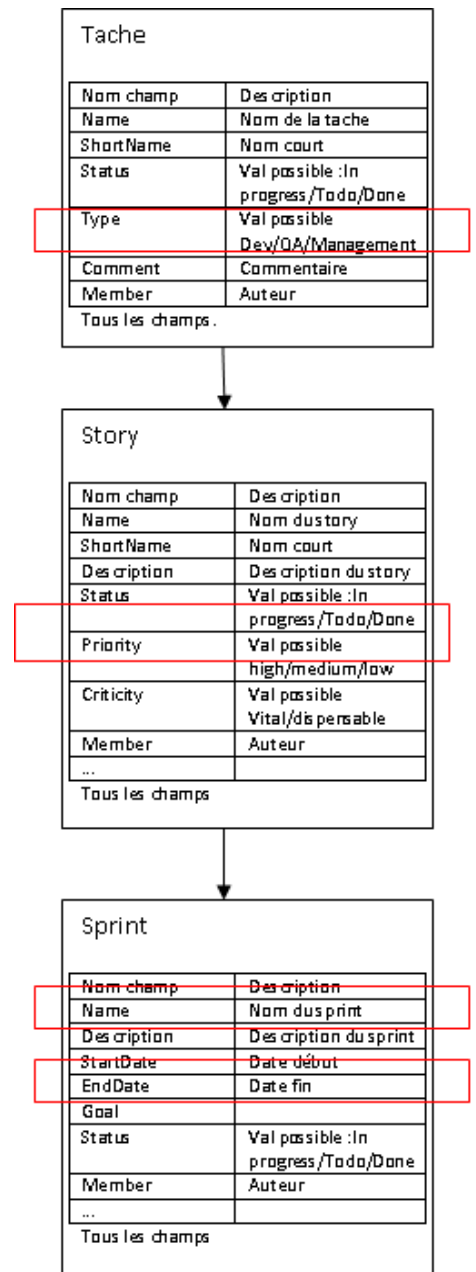


Figure 16 : Graphe de choix pour les critères de filtrage



Titre du document : Titre du document Coclico

Référence : L3-3-3\_ModeleProjetScrumForge

Version du 10/10/2011

My dashboards | New Dashboard 1

This is the page where you create your new dashboard. Lorem ipsum dolor sit amet.

Dashboard layout | PostIt layout | Filter criteria

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut.

**Tache**

Nom champ	Description
Name	Nom de la tache
ShortName	Nom court
Status	Val possible :In progress/Todo/Done
Type	Val possible :Dev/QA/Management
Comment	Commentaire
Member	Auteur

Tous les champs.

**Story**

Nom champ	Description
Name	Nom du story
ShortName	Nom court
Description	Description du story
Status	Val possible :In progress/Todo/Done
Priority	Val possible :high/medium/low
Criticality	Val possible :Vital/dé pensable
Member	Auteur

Tous les champs.

**Sprint**

Nom champ	Description
Name	Nom du sprint
Description	Description du sprint
StartDate	Date début
EndDate	Date fin
Goal	
Status	Val possible :In progress/Todo/Done
Member	Auteur

Tous les champs.

Criteria preview

**Tache**

Type

Dev  QA  Management

**Story**

Priority

Low  Medium  High

**Sprint**

Name


Sprint 1  Sprint 2  Sprint 3  Sprint 4

EndDate

Min date ————— Max date

Back Finish

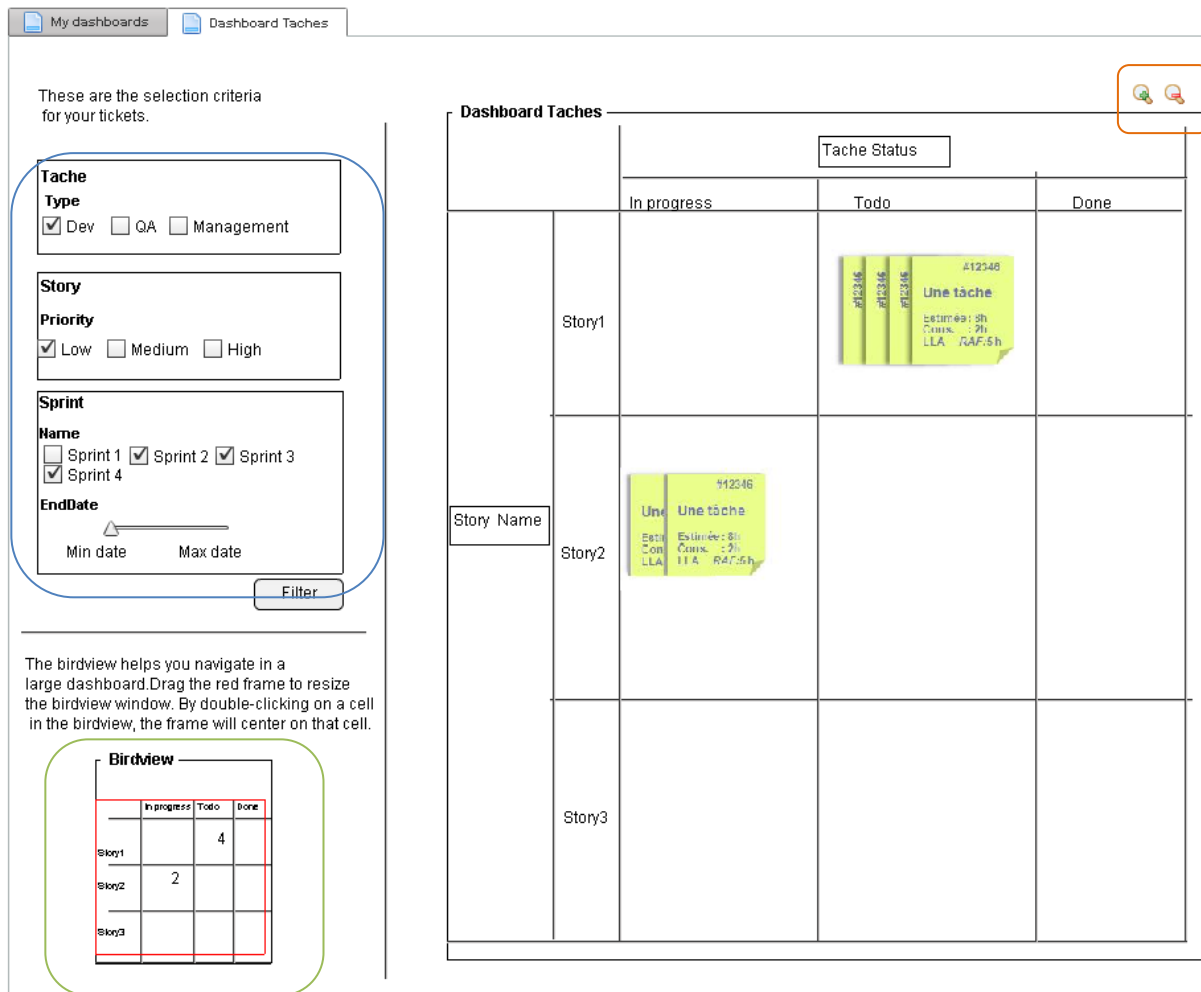
Figure 17 : Ecran du choix des critères de filtrage

	Titre du document : Titre du document Coclico
	Référence : L3-3-3_ModeleProjetScrumForge
	Version du 10/10/2011

## 2.4.1 Dashboard final

L'écran final de travail contient le dashboard construit à partir des champs des trackers, les critères de filtrage ainsi qu'une « carte » du dashboard.

Figure 18 : Ecran de travail - critères filtrage, birdview, dashboard



The screenshot shows a web interface with two main sections. On the left, there are filter criteria for tickets, and on the right, there is a Kanban board titled 'Dashboard Taches'.

**Filter Criteria:**

- Tache Type:**  Dev,  QA,  Management
- Story Priority:**  Low,  Medium,  High
- Sprint Name:**  Sprint 1,  Sprint 2,  Sprint 3,  Sprint 4
- EndDate:** A date range selector with 'Min date' and 'Max date' fields.

**Dashboard Taches (Kanban Board):**


		Tache Status		
		In progress	Todo	Done
Story Name	Story1		#12346 Une tâche Estimee: 2h Cours: 2h LLA: RAF,5h	
	Story2	#12346 Une tâche Estimee: 2h Cours: 2h LLA: RAF,5h		
	Story3			

**Birdview:**

	In progress	Todo	Done
Story1		4	
Story2	2		
Story3			

## 2.4.2 Birdview

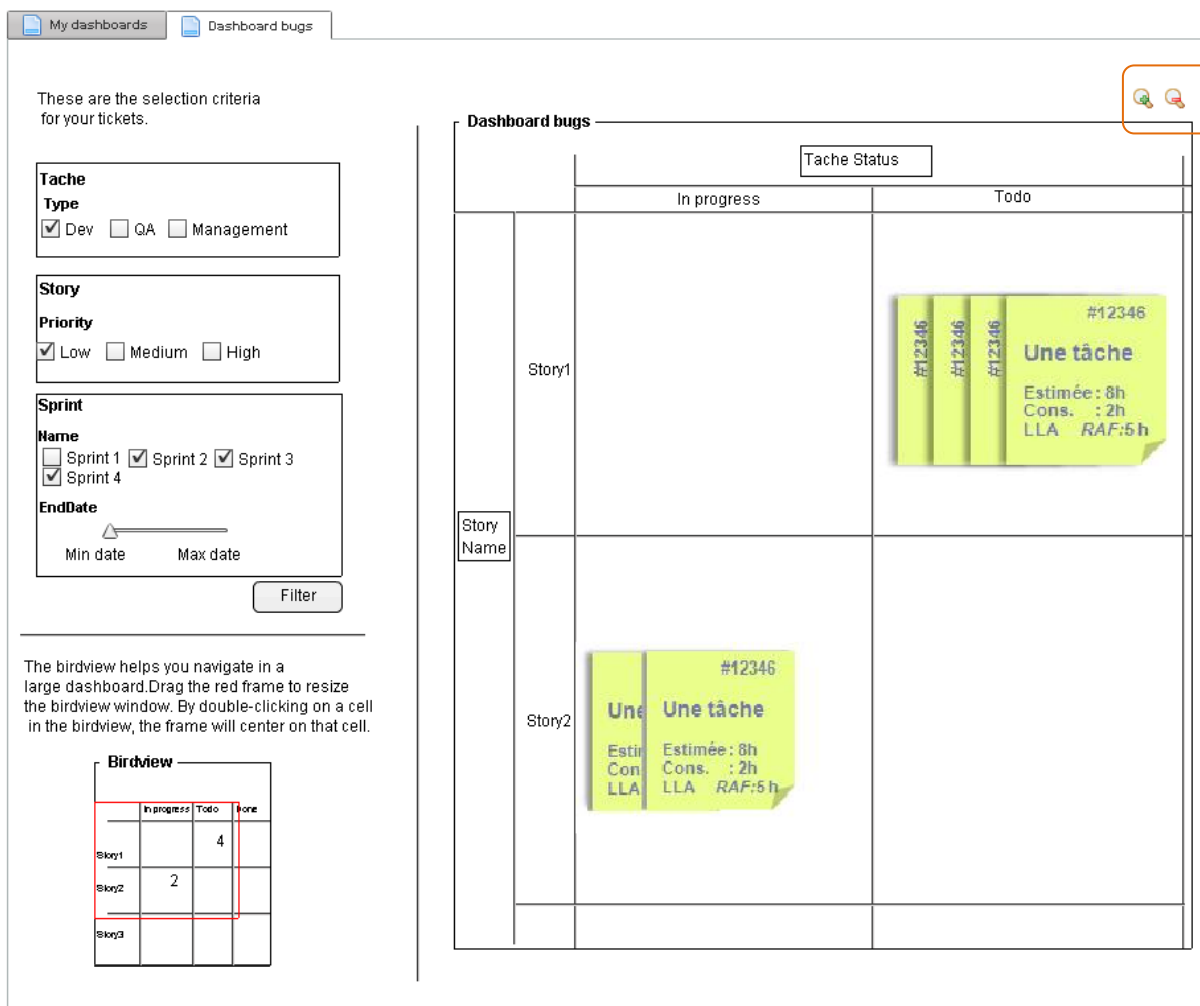
La carte du dashboard, appelée Birdview, représente une vue globale du dashboard et permet à l'utilisateur de centrer son attention sur certaines zones, dans le cas où le dashboard est trop grand et ne peut pas être affiché en totalité sur l'écran. La taille du birdview est modifiable – l'utilisateur peut tirer d'un coin du cadre rouge pour l'agrandir/réduire (le principe des fenêtres de Windows). L'utilisateur peut aussi déplacer ce cadre avec la souris sur le birdview pour voir différentes zones du dashboard. Le birdview est directement relié avec le dashboard : quand on déplace le cadre rouge sur le birdview, à droite, sur le dashboard, on affiche exactement la portion du dashboard encadrée en rouge sur le birdview. Si le cadre rouge est agrandi, on affiche une plus grande portion du dashboard (ça correspond à un zoom out). Si le cadre rouge est rétréci, on affiche une portion plus petite (zoom in).

	Titre du document : Titre du document Coclico
	Référence : L3-3-3_ModeleProjetScrumForge
	Version du 10/10/2011

On observe aussi, entouré en orange, deux boutons de zoom in/zoom out qui correspondent à la fonction de redimensionnement du cadre rouge du birdview. Si on fait un clic sur le zoom in (loupe avec +), le dashboard affiche moins de cases en plus grand et, sur le birdview, le cadre rouge se réduit. Si on clique sur le zoom out (loupe avec -), le dashboard affiche plus de cases en plus petit et, sur le birdview, le cadre rouge s'agrandit.

L'écran après le zoom in est présenté dans la Figure 16. Le dashboard n'affiche que deux lignes et deux colonnes et sur le birdview, on observe, dans le cadre rouge, la portion du dashboard affichée et en dehors du cadre rouge la portion du dashboard qui n'est pas affichée.

Sur le birdview on affiche aussi dans chaque case du dashboard le nombre de postits contenu. Si on fait un double-clic sur une case du birdview qui n'est pas affichée sur le dashboard à droite, le cadre rouge est centré sur la case du birdview et à droite, le dashboard aussi est centré sur cette case.



The screenshot shows the 'Dashboard bugs' interface. On the left, there are filter criteria for tickets:

- Tache Type:**  Dev,  QA,  Management
- Story Priority:**  Low,  Medium,  High
- Sprint Name:**  Sprint 1,  Sprint 2,  Sprint 3,  Sprint 4
- EndDate:** A range selector from 'Min date' to 'Max date'.


A 'Filter' button is located below these criteria. Below the filters, a 'Birdview' inset shows a grid with columns 'In progress', 'Todo', and 'Done'. The counts are: Story1 (4), Story2 (2), and Story3 (0).

The main dashboard area is titled 'Dashboard bugs' and has a 'Tache Status' dropdown. It displays a Kanban board with 'In progress' and 'Todo' columns. A task card is visible in the 'Todo' column for 'Story1' with details: #12346, 'Une tâche', Estimée: 8h, Cons.: 2h, LLA, RAF:5h. Another task card is visible in the 'In progress' column for 'Story2' with details: #12346, 'Une tâche', Estimée: 8h, Cons.: 2h, LLA, RAF:5h.

Figure 19 Ecran de travail - dashboard zoom in

### 2.4.3 Affichage des critères de filtrage

Les critères de filtrage sont affichés à gauche du dashboard. Pour bien profiter de l'espace de travail, la partie des critères de filtrage et du Birdview sera « pliante » (soit par un panneau

	Titre du document : Titre du document Coclico
	Référence : L3-3-3_ModeleProjetScrumForge
	Version du 10/10/2011

collapsible<sup>1</sup> soit par un « peek panel »<sup>2</sup>). Un aperçu de l'écran final de travail est présent dans la Figure 16.

---

<sup>1</sup> <http://www.iwobanas.com/2009/09/creating-collapsible-panel-in-flex-4/>

<sup>2</sup> <http://www.billdwhite.com/wordpress/?p=29>