



## FILA1 – Projet de développement

École des Mines de Nantes



# Découverte et supervision automatique des serveurs d'un centre de données

MAI / JUIN 2013

#### GROUPE

Guillaume LELOUP  
Romain GUILLAUME  
Gwenaël BEC  
Clément GUET

#### CLIENT

École des Mines de Nantes

#### AUTEURS

Guillaume LELOUP  
Romain GUILLAUME  
Gwenaël BEC  
Clément GUET

#### CONTRIBUTEURS

Guillaume LELOUP  
Romain GUILLAUME  
Gwenaël BEC  
Clément GUET



## SOMMAIRE

<b>I. Récupération du code / mise en place de l'environnement .....</b>	<b>3</b>
<b>II. Planification des tâches.....</b>	<b>4</b>
<b>III. Conformité du code .....</b>	<b>6</b>
<b>IV. Déroulement du projet .....</b>	<b>8</b>
1) Burnup en stories .....	8
2) Burnup en tâches.....	8
3) Burnup en heures .....	9
<b>V. Retour d'expérience.....</b>	<b>10</b>



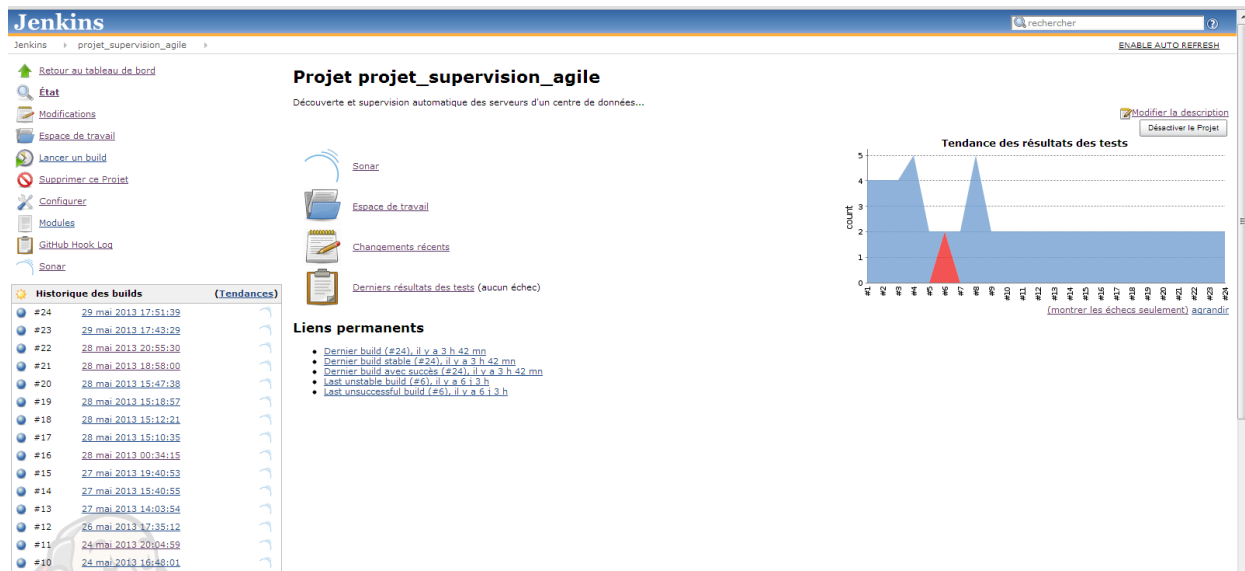
# I. Récupération du code / mise en place de l'environnement

Le projet ayant déjà été commencé par la promotion précédente, il nous a tout d'abord fallu récupérer le code existant sur le serveur SVN mis à notre disposition.

Une fois cette tâche effectuée, nous avons décidé de déployer le projet à l'aide de git, comme nous l'avons appris en cours, et de le rendre disponible sur Github à l'adresse suivante :

[https://github.com/soncoerok/projet\\_supervision\\_agile](https://github.com/soncoerok/projet_supervision_agile)

De plus, afin de rendre les phases de build plus facile, nous avons décidé d'installer Maven et Jenkins (une capture d'écran de Jenkins est disponible ci-dessous)



Enfin, nous avons également installé la solution Icescrum pour pouvoir effectuer le développement de ce projet à l'aide des méthodes agiles.



## II. Planification des tâches

Comme indiqué dans le tableau de planification ci-dessous, nous avons suivi 6 stories principales durant le sprint 1. De toutes ces stories, seul une n'est pas encore terminée à la fin du sprint. Ne prenant que très peu de ressources, elle a donc été décalée au sprint 2.

<i><b>Stories</b></i>	<i><b>Tâches</b></i>	<i><b>État</b></i>	<i><b>Complexité</b></i>	<i><b>Planifié</b></i>	<i><b>Reste à faire</b></i>	<i><b>Réel</b></i>
<b>Planification des tâches - Méthode Scrum</b>	Planification des tâches	OK	3	4	0	3
<b>Récupération du code</b>	Vérification du bon fonctionnement du code	OK	3	3.25	0	4
	Déploiement sur git	OK	3	0.5	0	0.5
	Récupération du code à partir du SVN	OK	3	0.25	0	0.25
<b>Mise en place des outils</b>	Mise en place de git	OK	13	1	0	0.7
	Mise en place de Maven	OK	13	1	0	0.5
	Mise en place de Jenkins	OK	13	7	0	7
	Mise en place Icescrum	OK	13	1	0	2
	Mise en place de sonar	OK	13	5	0	5.4
<b>Refactoring du code</b>	Refactoring du code	OK	13	10	0	10
	Correction de bug	OK	13	5	0	4

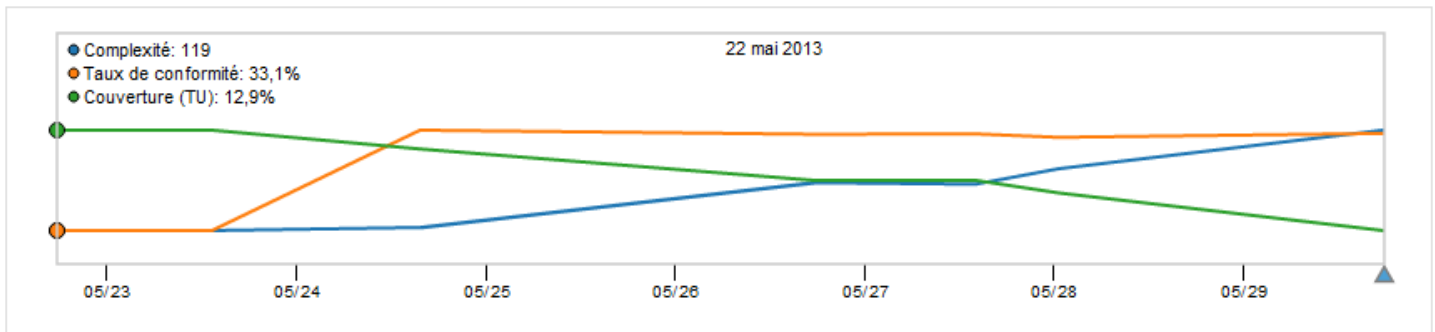


<b>Mise en place de netstat</b>	Mise en place de nmap	OK	<i>21</i>	<i>10</i>	<i>0</i>	<i>12</i>
	Recherche sur les protocoles réseaux	OK	<i>21</i>	<i>10</i>	<i>0</i>	<i>10</i>
<b>Amélioration de la reconnaissance OS</b>	Amélioration de la reconnaissance OS	En cours	<i>21</i>	<i>15</i>	<i>2</i>	<i>12</i>



### III. Conformité du code

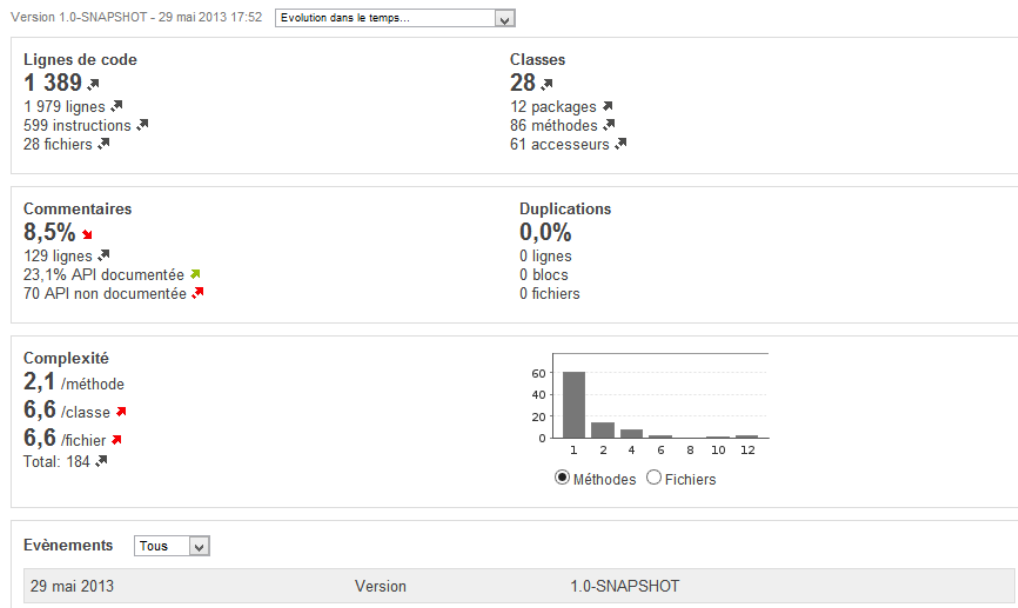
Partant d'un code existant (ayant été réalisé par la promotion précédente), nous avons utilisé l'outil de supervision Sonar afin d'effectuer un rapide état des lieux du projet.



Au début, nous avons 365 violations pour environ 1000 lignes de codes ce qui faisait un taux de conformité de 33.1 %.

A la fin de ce sprint, nous obtenons donc un meilleur taux de conformité même si notre taux de couverture a diminué (il faudra rajouter des tests unitaires dans le sprint suivant).

Voici les comptes rendus de sonar à la fin du sprint 1 :





Violations		Bloquant	0	
61		Critique	0	
Taux de conformité		Majeur	38	<div></div>
90,1%		Mineur	23	<div></div>
		Info	0	

Indice d'interdépendance entre packages	Dépendances à couper
26,1%	5 entre packages
> 9 cycles	6 entre fichiers

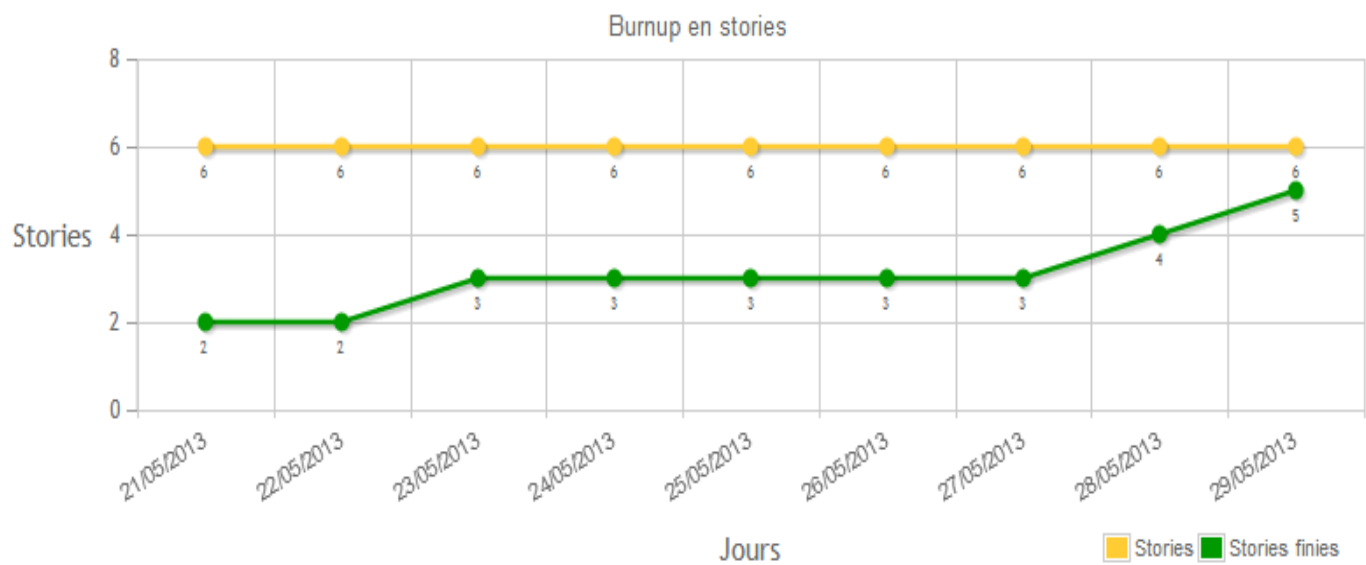
Couverture de code	Succès d'exécution des tests
9,7%	100,0%
9,6% de couverture de ligne	0 en échec
10,4% de couverture de branche	0 en erreur
	6 tests
	1.3 sec



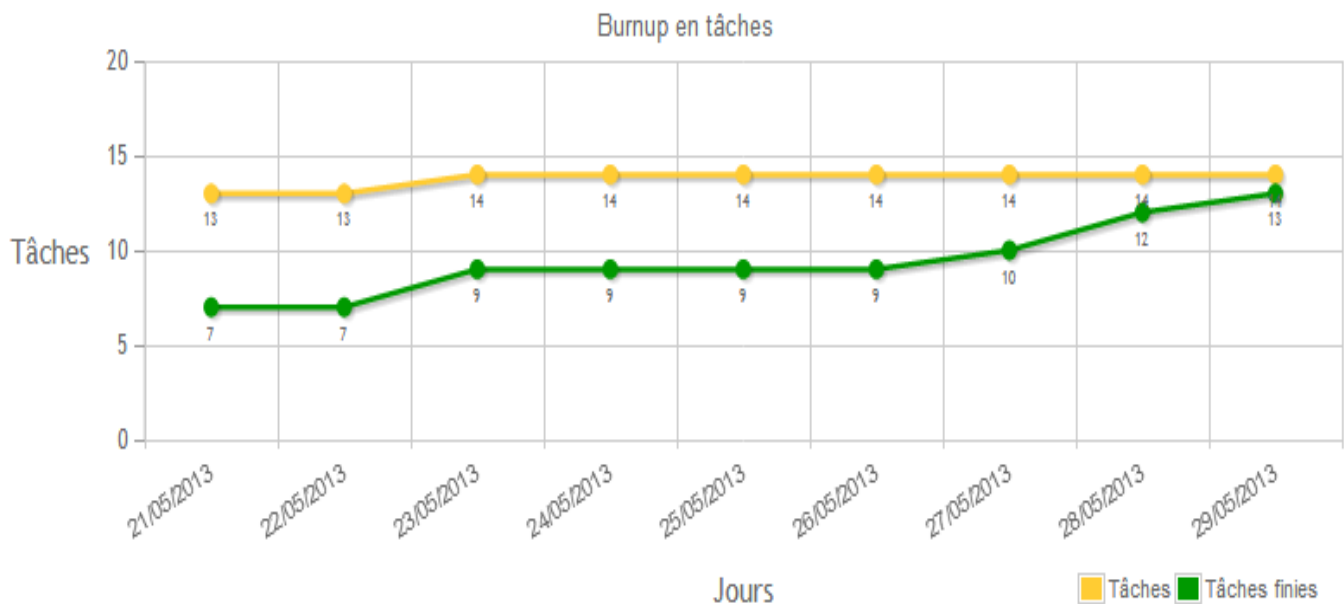
## IV. Déroulement du projet

Voici les différents graphes récapitulant le travail effectué lors du sprint 1 :

### 1) Burnup en stories



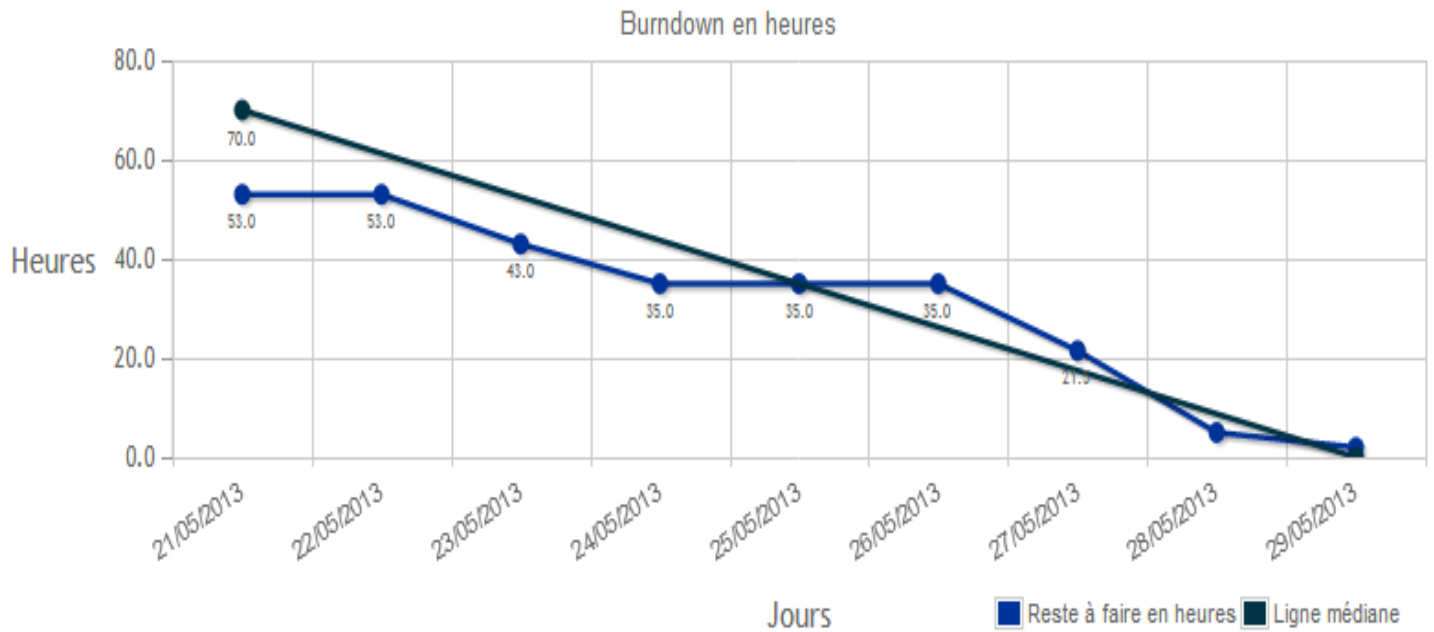
### 2) Burnup en tâches







### 3) Burnup en heures



Voici le graphe présentant l'évolution du temps passé sur le projet au cours du sprint 1. Nous pouvons constater, que nous avons à peu près respecté la moyenne de travail quotidienne.



## V. Retour d'expérience

Durant cette première phase, nous avons distingués trois parties principales :

- La compréhension du sujet et du code
- L'installation d'outils : Sonar, Jenkins, IceScrum, Git
- L'amélioration du code existant

La mise en place des outils, nous a requis beaucoup de temps, comme vous pourrez le constater sur les comptes rendus d'activité.

Concernant l'amélioration du code, nous nous sommes directement interrogés sur l'efficacité et la qualité du code existant. Ce qui nous a amenés à réfléchir aux méthodes utilisées. Nous avons donc choisis d'utiliser nmap pour récolter les informations concernant les machines.

En ce qui concerne la méthode Scrum dans le projet nous pensons qu'elle a été assez difficile à mettre en place au début, car nous n'avions pas tous les jours des heures de projets accréditées.