

L'application de la méthode COSMICⁱ d'analyse des points de fonction aux scénarios utilisateurs (*user stories*) fournit aux développeurs et propriétaires de produit un moyen simple et rapide pour mesurer la taille fonctionnelle de chaque scénario. Ensuite, cette mesure peut être utilisée, avec d'autres mesures, pour déterminer des indicateurs tels que la productivité, la vélocité et la densité de défauts.

La popularité croissante des méthodes Agiles de développement logiciel telles que eXtreme Programming (XP), Scrum et Kanban coïncide avec la popularité de l'utilisation des scénarios utilisateurs comme mécanisme de documentation des exigences.

Un scénario utilisateur bien élaboré décrit une caractéristique ayant une valeur pour le client. Les exigences, à différents niveaux de détail, peuvent être exprimées sous forme de scénarios utilisateurs. Par exemple, un client a défini quatre niveaux de détail distincts, soit les niveaux suivants : affaires, caractéristiques, fonctionnalités et composants logiciels. Au niveau utile le plus bas, il est attendu que chaque scénario soit la plus petite unité livrant de la valeur [aux yeux du client].

On dit qu'un scénario utilisateur est « une promesse de conversation » – il représente le développement d'une compréhension commune des besoins puisque le client et l'équipe de développement travaillent ensemble. Il combine les exigences fonctionnelles et non fonctionnelles. Des critères d'acceptation qui définissent la notion de « terminé » sont associés à chaque scénario.

Les scénarios mettent l'accent sur le « qui », le « quoi » et le « pourquoi » d'une caractéristique, et non pas sur le « comment » il sera mise en œuvre. Ils sont souvent rédigés sur des fiches de 127 mm x 76 mm (5" x 3") pour limiter la verbosité et fournir un support pratique. Ainsi, les scénarios sont pris individuellement, triés, affichés sur un tableau, etc. En passant, les grosses cartes écrites avec un feutre sont lisibles de plus loin.

La structure de chaque scénario est fondée sur le gabaritⁱⁱ suivant :

En tant que ... [rôle de l'intervenant] ...
Je veux ... [réaliser une activité ou extraire des données] ...
[avec une fréquence ou des caractéristiques de qualité]
Afin de ... [description de la valeur ou de l'avantage à obtenir].

Par exemple :

En tant qu'utilisateur d'une bibliothèque, **je veux** chercher des livres par titre avec vitesse et facilité **afin de** trouver tous les livres avec des titres similaires.

La définition opérationnelle des termes vagues tels que « vitesse » et « facilité » pourraient être écrite au verso de la carte pour inscrire les exigences non fonctionnelles.

Plusieurs équipes Agiles utilisent les *user story points* (*USP*) afin d'estimer l'effort nécessaire pour traiter un scénario jusqu'au point où il est déployable. Chaque individu (ou équipe) estime les *USP* en fonction de sa propre expérience locale, ce qui fait en sorte que les *USP* ne sont pas normalisables, ni transférables, ni stables au cours d'un même projet.

Il est important de noter que les *USP* servent d'intrant pour l'estimation de l'effort mais ils ne peuvent pas être utilisés pour comparer des indicateurs tels que la productivité, la vélocité ou la densité des défauts.

C'est ici que la méthode COSMIC peut aider.

En utilisant les concepts de la méthode COSMIC, les scénarios utilisateurs peuvent être rephrasés avec le gabarit suivant :

En tant que ... [utilisateur fonctionnel] ...
Je veux ... [répondre à un événement ou extraire des données] ...
[avec une fréquence ou des caractéristiques de qualité]
Afin de ... [description de la valeur ou de l'avantage à obtenir].

Dans l'exemple, l'utilisateur fonctionnel c'est « l'utilisateur d'une bibliothèque ». Le processus fonctionnel en réponse à un événement c'est « chercher des livres par titre ». Les exigences non fonctionnelles sont telles quelles, et la « liste des livres avec des titres similaires » représente une sortie. De plus, en quelques minutes seulement, on peut exprimer le scénario utilisateur sous forme de diagramme de séquences des mouvements de donnéesⁱⁱⁱ sur la même fiche (voir la [figure 1](#)).

ⁱ La méthode de mesure de la taille fonctionnelle COSMIC v3.0, Manuel de mesure : Le guide COSMIC d'implémentation pour ISO/IEC 19761 : 2003.

ⁱⁱ Voir le livre *User Stories Applied: For Agile Software Development* de Mike Cohn.

ⁱⁱⁱ Avec remerciements à Peter Fagg pour la notation *Data Movement Sequence Diagram* (DMSD).

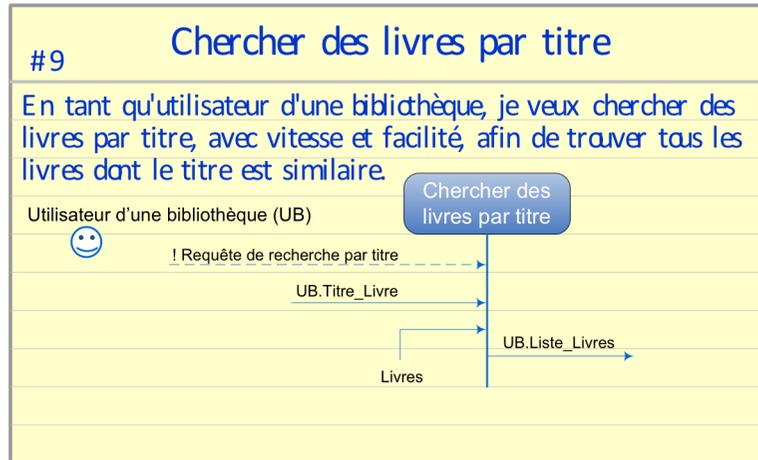


Figure 1 : Scénario utilisateur avec diagramme de séquence des mouvements de données

Ensuite, on peut déterminer la taille fonctionnelle d'un scénario en points de fonction COSMIC (PFC) en comptant les mouvements de données (Entrée, Sortie, Lecture et écriture) (voir la figure 2).

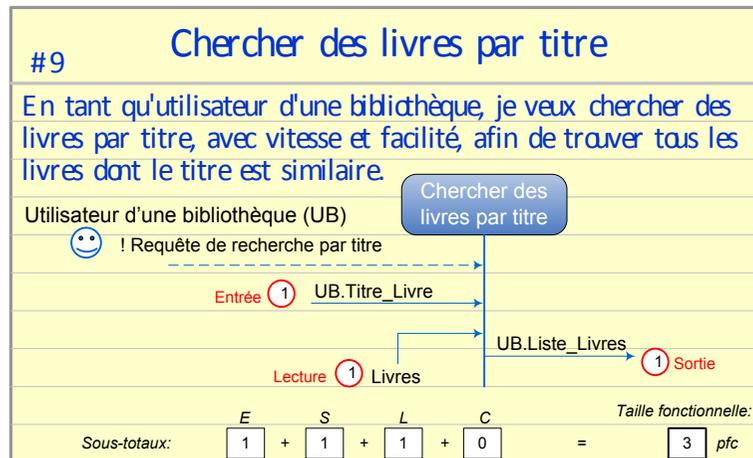


Figure 2 : Scénario utilisateur avec mesure de la taille fonctionnelle

Dans cet exemple, la taille fonctionnelle du scénario est de 3 PFC.

Naturellement, la plupart des scénarios utilisateurs sont plus complexes que dans cet exemple, qui n'inclut pas de mouvement d'écriture. Selon mon expérience, la mesure suivra quelques tendances et la majorité des scénarios seront de moins de 25 PFC. Alors, il devrait être relativement facile d'annoter le diagramme de séquence des mouvements de données sur la fiche (recto ou verso).

De cette façon, et en quelques minutes seulement, on peut visualiser chaque scénario et en mesurer la taille fonctionnelle. Avec une simple addition, on peut calculer la taille fonctionnelle du carnet de produit en entier, de l'ensemble des scénarios priorisés dans une itération spécifique ou encore de n'importe quel groupe de scénarios.

La taille fonctionnelle est une caractéristique des exigences fonctionnelles de l'utilisateur. Il est voulu que la taille fonctionnelle soit indépendante de la technologie et des méthodes utilisées ainsi que des compétences et de l'expérience des développeurs. Il s'agit d'une mesure des éléments requis, de la fonctionnalité livrée et de la quantité d'information à traiter. À ce titre, la taille fonctionnelle peut être utilisée avec les mesures d'effort et de durée (du début jusqu'à la livraison) pour calculer respectivement la productivité (taille fonctionnelle/effort) et la vélocité (taille fonctionnelle/durée). Si on garde un décompte des défauts détectés, on peut calculer la densité de défauts (#défauts détectés/taille fonctionnelle). Au fil du temps, après avoir complété le développement d'une série de scénarios, on peut déterminer la moyenne et la médiane de l'effort unitaire (heures/PFC) et du coût unitaire (\$ ou €/PFC). Il devient également simple de déterminer le nombre type d'unités COSMIC qu'une équipe peut livrer par itération (ou autre fréquence), ce qui est utile au moment de la planification.

La mesure COSMIC est normalisée et comparable dans le temps, entre les équipes, entre les projets et entre les organisations. Les développeurs, les propriétaires de produit et leurs clients ont donc un outil efficace et peu coûteux les aidant à faire des estimations, à suivre l'avancement des travaux et à évaluer la rentabilité.

Grant (PG) Rule 2010