

Livrable 4.3.1 – Spécification des fonctionnalités de liens/traçabilité

Livrable du au titre du projet	COCLICO
Lot	Sous-Projet 4 : Traçabilité et intégration continue
Tache	3 – Lien entre les artefacts projet gérés dans une Forge
Livrable	L4.3.1 Spécification des fonctionnalités de liens/traçabilité

Rédacteur(s)	Vérificateur(s)	Approbateur(s)
Nicolas Guérin		

Documents de références	
	Annexe technique du projet COCLICO

Gestion des versions

N° de version	Date	Auteurs	Modification apportées
1.0	07/10/2011	Nicolas Guérin	Version finale

Copyright : ©2011 Xerox

Ce document est diffusé sous les termes de la licence « Creative Commons Attribution 2.0 France ».

Ce document est le produit de travaux effectués dans le cadre du projet COCLICO, avec l'aide financière de collectivités publiques, dans le cadre des appels à projets des pôles de compétitivité SYSTEM@TIC et MINALOGIC.

Table des matières

1 - Présentation du document.....	3
1.1 - Objectifs	3
1.2 - Précisions importantes sur le livrable.....	3
2 - Introduction	4
2.1 - Traçabilité des exigences	4
2.2 - Application au domaine du logiciel	4
3 - Lien entre les artefacts de la forge	6
3.1 - Référencer un artefact	6
3.2 - Le moteur de références de Codendi	6
3.3 - Liens bi-directionnels	11
4 - Visualisation et navigation entre artefacts liés	12
4.1 - Création de lien	12
4.2 - Visualisation	12
4.3 - Aperçu	13
4.4 - Utilisation des références dans les trackers	15
5 - Conclusion	18

1 Présentation du document

1.1 Objectifs

Ce document présente un des résultat du sous-projet 4: Traçabilité et intégration continue. Il décrit les spécifications demandées dans la tâche 3 : Lien entre les artefacts projet gérés dans une Forge.

1.2 Précisions importantes sur le livrable

Les spécifications ont été définies au fur et à mesure de implémentation, dans une démarche agile.

De fait, nous n'avons pas produit dans un premier temps un document de spécifications (livrable L4.3.1), puis dans un deuxième une implémentation (livrable L4.3.2). Nous avons avancé les deux chantiers simultanément et itérativement.

Ce document présente donc les spécifications des fonctionnalités de lien/traçabilité telles qu'elles sont maintenant implémentées dans Codendi, à l'issu du projet Coclico.

C'est la raison pour laquelle elles sont illustrées de captures d'écran effectuées sur un serveur Codendi.

2 Introduction

2.1 Traçabilité des exigences

(ce qui suit s'inspire de l'article Wikipedia sur la [gestion des exigences](#))

La traçabilité des exigences est un concept clé mis en avant dans la gestion des exigences.

2.1.1 Anecdote

Un jour un constructeur automobile décida de réduire les coûts sur l'un de ses modèles phares. Une équipe se pencha sur les spécifications du modèle et chercha des axes de réduction des coûts. Quelqu'un s'avisait que le modèle était conçu pour résister à un vent arrière, avec de la pluie, de 200km/h ce qui entraînait des coûts de fabrication importants. On décida donc de changer cela pour produire un phare plus léger. Ce n'est qu'à l'automne, chez les concessionnaires qui trouvaient de l'eau dans les coffres, que l'on se rendit compte que les voitures étaient acheminées par train Express.

(Cette anecdote n'a pas de valeur historique mais seulement illustrative.)

2.1.2 Définition

La **traçabilité** est la possibilité de lire facilement ce qu'il est advenu et ce qu'il est censé advenir de quelque chose.

La traçabilité des exigences est le fait de pouvoir à tout instant connaître facilement l'origine et les liens entre les exigences, ainsi qu'entre les exigences et le reste du projet ou le contexte (notamment les besoins utilisateur, réalisation et tests).

Elle aide à répondre aux questions du type :

- D'où vient une exigence ? (Quel besoin cette exigence couvre-t-elle ? Pourquoi a-t-on conçu la solution de cette manière et quelles étaient les autres possibilités ?)
- Où met-on en œuvre cette exigence ?
- Comment interpréter cette exigence ?
- Quelle décision de conception affecte la mise en œuvre de l'exigence ?
- Cet élément de conception est-il nécessaire ?
- La solution réalisée est-elle conforme aux exigences ?
- Comment testera-t-on cette exigence ?
- Toutes les exigences sont-elles prises en compte ? (peut se lire : est-ce que le projet est terminé ?)
- Quel est l'impact du changement d'une exigence ?

2.2 Application au domaine du logiciel

Dans le cas du logiciel, le "produit" est principalement composé de code source et de documentation. Mais entre l'exigence et le code, il y a nécessairement plusieurs étapes.

Par exemple, une équipe de développement utilisant la méthode Scrum pourra fonctionner avec cette chaîne d'artefacts:

- Les exigences exprimées par le client - les "user stories" qui en sont dérivées - les tâches créées à partir des stories - le code développé pour chaque tâche

Grâce à l'utilisation d'une forge logicielle, on peut bénéficier de l'ensemble des outils nécessaires au sein de la même plateforme. Cela doit permettre de couvrir la traçabilité de l'intégralité du processus, de l'expression des exigences jusqu'au code source développé.

On va donc décrire ici comment créer ces liens entre les différentes étapes du processus, puis comment naviguer à l'aide de ces liens, et comment les visualiser.

3 Lien entre les artefacts de la forge

3.1 Référencer un artefact

Pour modéliser une exigences dans une forge logicielle, il existe plusieurs solutions. On peut en lister deux qui sont particulièrement utilisées: - une page wiki : cela permet de supporter l'édition par plusieurs personnes, et de versionner facilement l'exigence. - un artefact de tracker : en personnalisant un tracker, on peut modéliser les exigences de manière plus formelle en spécifiant les champs nécessaires, et en ajoutant un workflow par exemple.

Pour les tâches, user stories, epics, ou autre approche de l'implémentation de l'exigence, le recours à des trackers dédiés semble immédiat.

Enfin, il faut pouvoir référencer le code source généré pour pouvoir expliquer quel code correspond à quelle exigence.

A ce sujet, il est utile de se référer au livrable 4.6.1 qui présente une solution de traçabilité des exigences dans le code.

On voit par ailleurs qu'il peut être utile de référencer d'autres objets. Par exemple, une exigence peut être modifiée suite à un échange dans le forum du projet. Pouvoir référencer un message dans un forum semble donc une bonne idée. De même, une exigence peut se baser sur un document publié dans le gestionnaire de documents du projet, et donc référencer un document est également utile.

Il semble donc utile de pouvoir référencer un maximum d'objets de la forge, et pas seulement les artefacts de tracker. Pour cela on va définir un système de référence permettant de créer simplement un lien entre un objet de la forge et un autre. Ce mécanisme va gérer les objets suivants: - artefacts de trackers (tâches, user stories, exigences, bugs, etc.) - documents - pages wiki - fichiers (livraisons) - code source (révisions SVN, commits CVS ou Git) - messages dans les forums de discussion et dans la messagerie instantanée - un build dans le système d'intégration continue

Nous allons donc décrire comment fonctionne ce système de référence, et comment il peut être étendu à des objets qui ne seraient pas dans la forge.

3.2 Le moteur de références de Codendi

(note: ce chapitre est partagé avec le livrable « L1.5.1 Compte rendu d'étude de l'espace de nommage »)

3.2.1 Présentation des références

Le moteur de références de Codendi est un outil permettant de créer des références croisées entre objets de Codendi. En fonction d'un patron (*pattern*) de texte, Codendi peut

créer automatiquement un hyperlien vers l'objet souhaité.

Lorsqu'on tape un commentaire, ou un message de commit, tout morceau de texte qui suit le patron "XXX #NNN" est interprété comme une référence à l'objet XXX numéro NNN. Ainsi, 'art #123' sera détecté comme une référence à l'artefact numéro 123, et Codendi créera automatiquement un hyperlien qui pointe sur la page décrivant précisément cet artefact (bug, tache, etc.).

En fait, **ce mécanisme est disponible dans pratiquement tous les services de Codendi**. Ainsi, il est possible de créer des références dans les champs ou les commentaires d'un artefact d'un tracker, mais aussi dans un message posté dans un forum, dans une page Wiki, dans un document HTML, dans un champ de description (d'un projet, d'un fichier, ...), dans un message posté dans la 'chat-room', dans les messages de commit SVN, CVS ou Git, etc.

On peut également référencer un objet appartenant à un autre projet. A cette fin, on utilise soit 'XXX #num_projet:NNN', soit 'XXX #nom_projet:NNN' pour référencer l'objet XXX numéro NNN appartenant au projet numéro 'num_projet', ou dont le nom est 'nom_projet'. Par exemple, 'wiki #monproj:Accueil' est une référence à la page wiki 'Accueil' du projet 'monproj'.

Certains patrons de référence peuvent nécessiter plusieurs paramètres. Dans ce cas, on utilise le caractère '/' comme séparateur entre les paramètres. Par exemple, 'wiki #Accueil/2' crée une référence vers la page wiki 'Accueil' à la version 2.

Il existe deux types de patrons de référence :

- **Les patrons de référence système** : ces patrons de référence sont définis par l'administrateur du site et sont communs à l'ensemble des projets. Ils ne peuvent pas être modifiés, excepté leur statut (activé/désactivé). La plupart des patrons de référence système sont liés à un service spécifique. Ainsi 'art', 'doc', 'file' ou 'wiki' sont liés respectivement aux services outils de suivi, gestionnaire de documents, gestionnaire de fichiers et Wiki. Afin de faciliter l'utilisation de ces patrons de référence, ils sont automatiquement activés ou désactivés lorsque le service correspondant est activé/désactivé. Cependant, il est toujours possible de les activer ou de les désactiver manuellement.
- **Les patrons de référence projet** : ces patrons de référence peuvent être entièrement personnalisés par les administrateurs du projet.

3.2.2 Patrons de référence prédéfinis

Voici la liste des patrons de référence prédéfinis :

- **art #num ou artefact #num** : référence à l'artefact numéro 'num'. C'est un numéro global au système (l'artefact 'num' est unique, et n'existe que dans un projet). Cette référence crée un lien vers la page de description et de mise à jour de l'artefact. En sus des mots clés 'art' et 'artefact', les artefacts peuvent être référencés à l'aide du nom court

de l'outil de suivi. Ainsi, un artefact issu d'un outil de suivi de bugs (anomalies) pourra être référencé par bug #NNN, une demande de support (support request) par sr #NNN, une tâche par task #NNN et un patch avec patch #NNN. Ainsi, lorsque l'on crée un outil de suivi personnalisé, il est important de bien choisir son nom court.

- **commit #num ou cvs #num** : référence au commit CVS numéro 'num'. C'est un numéro global au système. Cette référence crée un lien vers la page de description du commit, qui contient le message du commit, la liste des fichiers modifiés, les liens vers les modifications, ...

- **rev #num ou revision #num ou svn #num** : référence à la révision Subversion numéro 'num'. C'est un numéro propre au projet : si on souhaite référencer une révision Subversion d'un autre projet, il faut spécifier le projet dans la référence (par exemple: 'rev #monproj:123'). Cette référence crée un lien vers la page de description de la révision, qui contient le message du commit, la liste des fichiers modifiés, les liens vers les modifications, ...

- **git #num**: référence au commit Git dont le hash est 'num'. Cette référence crée un lien vers la page de description du commit, qui contient le message du commit, la liste des fichiers modifiés, les liens vers les modifications, ...

- **build #num ou build #job/num**: référence au build Hudson 'num'. On peut préciser le job Hudson si plusieurs ont été déclarés dans le projet. Le lien pointe vers la page Hudson contenant la description du build, son résultat, ses tests, etc.

- **wiki #pagewiki et wiki #pagewiki/num** : référence à la page Wiki nommée 'pagewiki'. Le second format permet de spécifier la version de la page Wiki. Les pages Wiki sont propres à un projet, donc si on doit référencer une page appartenant à un autre projet, il faut spécifier le projet dans la référence.

- **doc #num ou document #num** : référence au document numéro 'num'. C'est un numéro global au système. Les numéros de documents sont visibles dans la page principale du service 'Documentation' : l'identifiant est visible lorsqu'on place la souris sur le titre. La référence pointe sur le document lui-même.

- **dossier #num ou folder #num** : référence au dossier numéro 'num' dans le gestionnaire de documents. C'est un numéro global au système. Les numéros de dossier sont visibles dans la page principale du service 'Documentation' : l'identifiant est visible lorsqu'on place la souris sur le titre. La référence pointe sur le dossier lui-même.

- **news #num** : référence vers l'annonce numéro 'num'. C'est un numéro global au système. Cette référence crée un lien vers la page d'annonce, où il est possible d'ajouter des commentaires.

- **forum #num et msg #num** : référence vers le forum numéro 'num' ou vers le message de forum numéro 'num'. Ce sont des numéros globaux au système. La première référence crée un lien vers la page d'accueil du forum, tandis que la deuxième crée un lien vers un message précis.

- **chat #num** permet de référencer une discussion archivée de l'outil de messagerie instantanée. Il suffit de préciser le numéro de la conversation correspondant.

- **file #num** : référence vers le fichier numéro 'num'. C'est un numéro global au système. Ce type de référence permet de télécharger directement un fichier faisant partie d'une version. Les numéros de fichiers sont visibles dans la page principale du service 'Fichiers' : l'identifiant est visible lorsqu'on place la souris sur le nom du fichier. La référence pointe sur le fichier lui-même, et le navigateur peut alors demander à quel endroit on souhaite le sauvegarder.

- **release #num** : référence vers la version (release) numéro 'num'. C'est un numéro global au système. Les numéros de version sont visibles dans la page principale du service 'Fichiers' : l'identifiant est visible lorsqu'on place la souris sur le nom de la version. La référence pointe sur la page d'accueil du gestionnaire de fichier, où la version référencée est mise en valeur dans la liste.

3.2.3 Utilisation des références

Lors des phases de développement ou de maintenance d'un projet, il est essentiel de garder la trace des changements effectués dans le code source. C'est ce que font les systèmes de gestion de configuration comme CVS ou Subversion. En complément de ce suivi, il est tout aussi critique de pouvoir relier ces changements de code aux artefacts (une tâche, une anomalie ou une demande d'assistance) qui ont amené les développeurs à modifier le code ou la documentation. Et inversement, à la lecture d'un artefact il est très utile de voir quels changements il a engendré dans le code source.

L'intégration de CVS et de Subversion avec Codendi fournit précisément ce mécanisme de références croisées. Ceci est rendu possible par l'utilisation des références à utiliser dans les commentaires des artefacts ou dans les messages de commits que Codendi reconnaît automatiquement.

Le système n'est pas limité aux références à des commits ou à des artefacts, puisqu'il est également possible de référencer un message posté dans un forum, une documentation, ou un fichier. Le système est suffisamment flexible pour permettre le référencement d'objets qui ne sont pas dans Codendi. Il suffit de créer ses propres patrons de référence pour relier des messages à un gestionnaire de document externe comme DocuShare, ou à un outil de gestion de code source comme ClearCase.

3.2.4 Création ou modification d'un patron de référence

Pour créer ou modifier un patron de référence, il faut remplir les champs suivants :

- **Mot-clé** : C'est le mot-clé qui va déclencher la création d'une référence lorsqu'il est détecté. Il est préférable de le choisir court et explicite pour favoriser la lecture des références.

- **Description** : Courte description du patron de référence. Elle est affichée lorsqu'on place

la souris sur une référence extraite.

- **Lien** : C'est l'URL vers lequel pointe la référence. L'utilisateur sera automatiquement redirigé vers cette adresse s'il clique sur la référence. L'URL ne pointe pas nécessairement vers le serveur Codendi : on peut créer des patrons de référence qui pointent vers des pages externes. La nouvelle page sera chargée dans le fenêtre courante.

Certaines chaînes de caractères seront interprétées lors de la création de la référence :

- \$projname : nom court du projet.
- \$group_id : numéro du projet.
- \$0 : le mot-clé extrait par ce patron de référence.
- \$1 : le premier paramètre de la référence.
- \$2 : le deuxième paramètre de la référence.
- \$3...\$9 : du troisième au neuvième paramètre de la référence.

3.2.5 Exemples:

artifact #25

'\$0' vaut 'artifact', '\$1' vaut '25'. C'est une référence vers l'artifact numéro 25.

wiki #codendi:Accueil/1

'\$0' vaut 'wiki', '\$1' vaut 'Accueil', '\$2' vaut '1', '\$projname' vaut 'codendi'. C'est une référence vers la première version de la page wiki 'Accueil'.

myref #123:1/23/456

'\$0' vaut 'myref', '\$1' vaut '1', '\$2' vaut '23', '\$3' vaut '456' et '\$group_id' vaut '123'. C'est une référence définie par l'utilisateur qui point un objet appartenant au groupe dont l'id est 123.

google #codendi/xerox

'\$0' vaut 'google', '\$1' vaut 'codendi', '\$2' vaut 'xerox'. Si vous avez défini le patron de référence 'google', avec un lien pointant vers [http://www.google.com/search?hl=en&q=\\$1+\\$2](http://www.google.com/search?hl=en&q=$1+$2), cliquer sur la référence 'google #codendi/xerox' lancera une recherche Google de 'codendi xerox'.

ds #123

'\$0' vaut 'ds', '\$1' vaut '123'. Si vous avez défini le patron de référence 'ds', avec un lien pointant vers [http://docushare/dsweb/Get/Document-\\$1](http://docushare/dsweb/Get/Document-$1), cliquer sur la référence 'ds #123' lancera le téléchargement du document '123' de votre serveur Docushare local.

Il faut noter que le nombre de paramètres est important : si le nombre de paramètres dans

le texte ne correspond pas au nombre de paramètres définis dans le patron de référence, la référence ne sera pas extraite. Ceci permet de créer plusieurs patrons de référence avec le même mot-clé, mais un nombre de paramètres différent. Voyez par exemple les références suivantes : 'wiki #Accueil' est une référence à un seul paramètre pointant vers la page Wiki 'Accueil', alors que la référence 'wiki #Accueil/2' est une autre référence à deux paramètres pointant vers la page Wiki 'Accueil' à la version '2'.

3.3 Liens bi-directionnels

Lorsqu'on crée un lien dans un objet A vers un objet B, il semble indispensable que l'objet B "sache" qu'il est référencé par l'objet A. Et par ailleurs, on veut éviter d'avoir à créer ce lien via une nouvelle référence.

De fait, le système de référence va devoir créer des liens bi-directionnels: tout objet référencé doit savoir par qui il est référencé, et si possible, proposer un lien vers ces objets qui le référencent. Pour que cela soit possible, il est nécessaire d'enregistrer chaque référence dans la base de données de la forge pour que ce lien soit exploitable de partout.

Nous verrons plus bas un cas concret où, en référençant une tâche dans un message de commit, la tâche est mise à jour automatiquement avec un lien vers la révision SVN correspondante.

4 Visualisation et navigation entre artefacts liés

4.1 Création de lien

Pour créer un lien, la méthode la plus simple est de taper une référence utilisant un modèle défini dans le projet dans un des multiples champs acceptant les références.

Exemple dans une page wiki:

Éditer : **ExempleDeReference**

```
Comme discuté dans le forum (en particulier dans le msg #2), il
semble indispensable de compléter les spécifications techniques
de la plateforme (doc #24). Les modifications du code source (rev
#3 et rev #4) ne sont pas suffisantes.
Il faudra ensuite mettre à jour l'exigence #19.
```

4.2 Visualisation

4.2.1 Liens


La manière initiale de visualisation d'une référence par la création d'un lien HTML dans la page contenant la référence. Cette affichage se fait automatiquement par application d'une expression régulière sur le texte affiché afin de détecter les expressions possible de référence. Le système vérifie également que le mot clé utilisé (dans l'expression *mot_clé #identifiant*) correspond à une référence définie dans le projet. Sui c'est le cas, la référence est transformée en lien, comme on peut le voir dans l'exemple ci-dessous (page wiki).

ExempleDeReference

```
Comme discuté dans le forum (en particulier dans le msg #2), il
semble indispensable de compléter les spécifications techniques
de la plateforme (doc #3). Les modifications du
code source (rev #3 et rev #4) ne sont pas suffisantes. Il faudra
ensuite mettre à jour
l'exigence #19.
```

4.2.2 Espace dédié

Les références à un objet peuvent également être regroupées dans une espace dédié dans la page de description de l'objet. C'est le cas pour les objets de type: artefact de tracker, page wiki, document ou révision de code par exemple.

	Titre du document : Livrable 4.3.1 – Spécification des fonctionnalités de liens/traçabilité
	Référence : L4.3.1
	Version 1.0 du 07/10/2011

Dans la cas d'une page wiki, ces références sont listées au dessus de la page dans un espace modulable:


Références

Liste des objets étant référencés par cet objet ou le référençant. [Cacher les objets référencés par cet objet](#)


Artefact

 [exigence #5](#) ✕, [exigence #19](#) ✕

Document

 [doc #3](#) ✕

Message de forum

 [msg #2](#) ✕

Révision Subversion

 [rev #3](#) ✕, [rev #4](#) ✕

ExempleDeReference

Comme discuté dans le forum (en particulier dans le [msg #2](#)), il semble indispensable de compléter les spécifications techniques de la plateforme ([doc #3](#)). Les modifications du code source ([rev #3](#) et [rev #4](#)) ne sont pas suffisantes. Il faudra ensuite mettre à jour l'[exigence #19](#).

4.3 Aperçu

4.3.1 Cas général

Certains types de référence proposent un aperçu de l'objet référencé. Il suffit de positionner le pointeur de la souris sur le lien correspondant. Une bulle (tooltip) apparait alors présentant un résumé de l'objet en question.

Suivant la nature de l'objet, le tooltip affiche des informations différentes:

Exemple 1: Document

ExempleDeReference

Comme discuté dans le forum (en particulier dans le [msg #2](#)), il semble indispensable de compléter les spécifications techniques de la plateforme ([doc #24](#)). Les modifications du code source ([rev #3](#) et [rev #4](#)) ne sont pas suffisantes. Il faudra ensuite mettre à jour l'exigence #19.

Propriétaire: "Nicolas Guérin (guerin)" Dernière modification le octobre 6, 2011 11:45 par "Nicolas Guérin (guerin)"

Réunion plénière du 14 juin
Compte rendu de la réunion du 14 juin 2011

Exemple 2: Artefact

ExempleDeReference

Comme discuté dans le forum (en particulier dans le [msg #2](#)), il semble indispensable de compléter les spécifications techniques de la plateforme ([doc #24](#)). Les modifications du code source ([rev #3](#) et [rev #4](#)) ne sont pas suffisantes. Il faudra ensuite mettre à jour l'exigence #19.

Propriétaire: "Nicolas Guérin (guerin)" Dernière modification le octobre 6, 2011 11:45 par "Nicolas Guérin (guerin)"

Artifact ID: 19
Product: Product1
Summary: Fournir support pour revue.
State: New

[Imprimer la page](#) | [HistoriqueDeLaPage](#) | [Diff](#) |

Exemple 3: Révision SVN

ExempleDeReference

Comme discuté dans le forum (en particulier dans le [msg #2](#)), il semble indispensable de compléter les spécifications techniques de la plateforme ([doc #24](#)). Les modifications du code source ([rev #3](#) et [rev #4](#)) ne sont pas suffisantes. Il faudra ensuite mettre à jour l'exigence #19.

Propriétaire: "Nicolas Guérin (guerin)" Message de commit: Add documentation files. le octobre 6, 2011 11:45 par "Nicolas Guérin (guerin)"

Date: 01/09/2011 12:00

4.3.2 Définition de l'aperçu: le cas des trackers

Dans le cas des trackers, la nature de l'artefact manipulé peut être très diverse. On propose donc une interface permettant à l'administrateur du projet de définir le contenu de l'aperçu, en choisissant parmi les champs du tracker.

Voir l'exemple ci-dessous:

Tooltip

Les champs suivants seront affichés dans le tooltip :

Artifact ID ✘
Product ✘
Summary ✘
State ✘

Ajouter un champ au tooltip :

4.4 Utilisation des références dans les trackers

4.4.1 Description

On vient de le voir, le cas des trackers est particulier car c'est un outil central dans la gestion du projet logiciel. Du fait de ses possibilités de configuration, le tracker peut être employé quel que soit la méthode de développement utilisée. Ainsi, on utilisera des trackers pour modéliser bugs, exigences, tickets, tâches, user stories, risques, tests, patches, etc. Ainsi, c'est un outil important pour gérer la traçabilité dans le projet;

De fait, il est possible d'utiliser le mécanisme décrit plus haut (*mot_clé #identifiant*) pour créer des références dans un artefact (objet manipulé par le tracker), que ce soit dans un champ ou dans un commentaire.

Mais pour aller plus loin, nous avons besoin de définir un type de champ "référence à artefact" qui facilite la gestion de ces références.

Les avantages attendus sont: - possibilité de lister les artefacts liés dans un rapport de tracker - possibilité de créer une référence de manière graphique (à la souris) sans utiliser les mots clés ni avoir besoin de connaître les identifiants - possibilité de créer un artefact lié directement depuis la page de description d'un artefact.

4.4.2 Exemple

Dans cet exemple, on va lier deux "user stories" existantes à une exigence.


Première étape: affichage de l'exigence

exigence #19 - Fournir support pour revue.

Identification

Artifact ID #19 Product * Product1 Version * 1.0 State * New

Type * specification Nature * Fonctionnal Complexity Medium Priority High

Summary: Fournir support pour revue. Satisfy:  ou  Sélection d'un artefact existant

Entrez des ids d'artefacts séparés par une virgule

Deuxième étape, sélection des artefacts à lier

Projet de demonstration Coclico » Admin Site Web Forums Listes Documents Wiki Annonces Subversion

Sélectionner les artefacts à lier à exigence #19 - Fournir support pour revue.

Manuellement

Si vous connaissez l'id de l'artefact, entrez-le manuellement dans le champs texte ci-dessous. Le champs attends une liste d'ids séparés par des virgules.

Artefact id(s)

Rechercher des artefacts

Projet de demonstration Coclico

User Stories > Default

Default

Results

6 artefact(s)

	Artifact ID	As a	I want to	So that	Business value	Story points	Remaining Effort	Source Requirement	S #
<input type="checkbox"/>	7	User	launch a Chrome web browser	I can use the application	8	8	0	Requirement #4	
<input type="checkbox"/>	8	User	have a new feature	it works	5	3	1		
<input type="checkbox"/>	9		have 2 new featruess	it works better	13	13	8		
<input type="checkbox"/>	10	Admin	add feature3	it improves the product	20	3	15		
<input type="checkbox"/>	11	User	add feature	it becomes	5	5	13		

On voit que l'on peut sélectionner n'importe quel type d'artefact du projet (choix du tracker), ainsi que le rapport qui sera utilisé pour l'affichage. De plus, l'outil offre une sélection rapide basée sur les artefacts récemment modifiés.

Troisième étape: Affichage des artefacts liés

Satisfy

8,9



ou








Créer un nouvel artefact

Entrez des ids d'artefacts séparés par une virgule

User Stories

2 artefact(s)

		Artifact ID	As a	I want to	So that	Business value	Story points	Remaining Effort	Source Requirement	Sprint #
		8	User	have a new feature	it works	5	3	1		2
		9		have 2 new featrues	it works better	13	13	8		2
							Nombre : 2	Somme : 9		

On peut maintenant voir les artefacts liés dans un tableau similaire à un rapport de tracker.

Certains champs du tableau peuvent être eux-même des références vers d'autres artefacts.

5 Conclusion

Les mécanismes décrits dans ce document permettent de faciliter la traçabilité dans un projet hébergé par une forge. Le système de références qui est décrit permet de créer, visualiser ces références, et naviguer facilement entre artefacts liés. Ainsi relier une révision dans le code source à une tâche, elle même liée à une exigence, est maintenant facilité.

Par ailleurs, les captures d'écran utilisées dans ce document sont extraites d'un serveur Codendi implémentant les mécanismes décrits. Ces développements ont eu lieu dans le cadre du livrable L4.3.2.