	Titre du document : Compte rendu d'étude
	Référence : L 1.5.1
	Version du 24/09/2011

Espace de nommage

Livrable du au titre du projet	COCLICO
Lot	1
Tache	0.5
Livrable	1.5.1

Rédacteur(s)	Vérificateur(s)	Approbateur(s)
Olivier Berger	C Bayle	




	Titre du document : Compte rendu d'étude
	Référence : L 1.5.1
	Version du 24/09/2011

Table des matières

I. Etude de l'existant dans les forges	4
A. Liens existants entre entités entre outils de la forge	4
B. Profil users FusionForge	4
i. Profils RDFa (>= FusionForge 5.1)	4
ii. Profils RDF content-negotiation (FusionForge trunk)	5
C. Profil projet FusionForge	5
i. Profils RDFa (>= FusionForge 5.1)	5
ii. Profils RDF content-negotiation (FusionForge trunk)	5
D. Le moteur de références de Codendi	6
i. Présentation des références.....	6
ii. Patrons de référence prédéfinis.....	6
iii. Utilisation des références.....	8
iv. Création ou modification d'un patron de référence.....	8
E. Problèmes posés	10
i. Référencement par identifiants	10
ii. URLs ambiguës	10
iii. Accès à des données privées ?	10
iv. Versionnement des entités	11
F. Solutions envisageables	11
i. URLs REST pour l'accès aux entités de la forge	11
ii. URIs pour l'identification des ressources	11
iii. Miscellaneous	11
G. Perspectives	15

	Titre du document : Compte rendu d'étude
	Référence : L 1.5.1
	Version du 24/09/2011

	Titre du document : Compte rendu d'étude
	Référence : L 1.5.1
	Version du 24/09/2011

I. Etude de l'existant dans les forges

A. Liens existants entre entités entre outils de la forge

- Mention de correction de bugs ou de complétion de tâche dans des commits
- Mention de commits corrigeant des bugs ou complétant des tâches
- Mention de dépendances entre bugs
- Mention de dépendances entre artefacts de nature différente: une exigence peut référencer des "user stories", qui peuvent elle-même référencer des tâches.
- Mention de branches de développement, milestones, releases
- Changelogs de releases (numéros de bugs)
- Utilisateurs auteurs d'artefacts ou de changements (+ signatures)
- Références aux conteneurs (projet, outil, etc.)
- Liens entre pages wiki
- Attachements à des messages de forums, etc.
- Réponses entre emails et post de forums.
- Référence à des documents, pages wiki, fichiers, messages de forus, etc.

B. Profil users FusionForge

Les URLs des profils des users FusionForge sont accessibles via l'URL `/users/<login>`.

A partir de FusionForge 5.1, les pages HTML à cette URL contiennent des méta-données en RDFa identifiant les utilisateurs (voir ci-dessous).


A noter que dans les différents outils de la forge, un utilisateur peut être connu par différents identifiants :

- login de la partie Web (correspondant à celui de l'URL `/users/<login>`), a priori identique au login SSH, et donc aux auteurs des commits dans les outils VCS
- Identifiant wiki (version modifiée du du login précédent, avec une première lettre en majuscule)
- Identifiant sur les listes de discussion via un email appartenant au même utilisateur (pas forcément unique, le lien n'étant pas maintenu par la forge)
- ...

Pour les liens entre différents profils d'un même utilisateur dans différents outils, on se référera à l'ontologie de forge du WP2, qui modélise la notion de personne, de comptes, et permet donc d'implémenter un lien entre les différents comptes d'une même personne en cas de besoin d'unification.

Ainsi, différents comptes d'une même personne peuvent avoir des logins différents, mais des emails identiques (éventuellement encodés en SHA1), permettant de faire le lien entre ceux-ci.

i. Profils RDFa (>= FusionForge 5.1)

	Titre du document : Compte rendu d'étude
	Référence : L 1.5.1
	Version du 24/09/2011

L'URL `/users/<login>` est donc déréréférençable par des outils compatibles RDFa pour extraire des informations concernant un utilisateur FusionForge 5.1.

Cette URL standardisée est donc un bon identifiant pour les utilisateurs d'une forge FusionForge.

Il n'existe pas de fonction de renommage des logins des utilisateurs dans les forges, en général, donc une URL de ce type reste relativement pérenne.

Cf. <http://lists.fusionforge.org/pipermail/fusionforge-general/2011-June/001458.html> pour des détails sur la façon d'accéder à ces méta-données

Cependant, les outils compatibles RDFa sont peu nombreux. D'où la nécessité d'implémenter plutôt un accès réel à des données RDF via un mécanisme de content-negotiation sur ces URLs (ci-dessous).

ii.Profils RDF content-negotiation (FusionForge trunk)

Un plugin en cours de développement (via une contribution de COCLICO) permettra de fournir un profil RDF (conforme à l'ontologie de forges du WP2), en content-negotiation sur l'URL standardisée des utilisateurs.

Plus de détails sur : https://fusionforge.org/plugins/mediawiki/wiki/fusionforge/index.php/Foaf_Profiles_Plugin

C.Profil projet FusionForge

La fonctionnalité des profils d'utilisateurs présente sur `/users/<login>` est totalement équivalente à la fonctionnalité d'accès à un profil de projet sur `/projects/<nom_projet_court>`.

Les considérations sont les mêmes, concernant RDFa ou l'accès à un contenu RDF par content-negotiation.


i.Profils RDFa (\geq FusionForge 5.1)

L'URL `/projects/<nom_projet_court>` est donc déréréférençable par des outils compatibles RDFa pour extraire des informations concernant un utilisateur FusionForge 5.1.

ii.Profils RDF content-negotiation (FusionForge trunk)

Un plugin en cours de développement (via une contribution de COCLICO) permettra de fournir un profil RDF (conforme à l'ontologie de forges du WP2), en content-negotiation sur l'URL standardisée des projets.

Plus de détails sur : https://fusionforge.org/plugins/mediawiki/wiki/fusionforge/index.php/DOAP_RDF_plugin

	Titre du document : Compte rendu d'étude
	Référence : L 1.5.1
	Version du 24/09/2011

D. Le moteur de références de Codendi

i. 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.).

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.

ii. Patrons de référence prédéfinis

Voici la liste des patrons de référence prédéfinis :

- **art #num ou artifact #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 'artifact', 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,

	Titre du document : Compte rendu d'étude
	Référence : L 1.5.1
	Version du 24/09/2011

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

	Titre du document : Compte rendu d'étude
	Référence : L 1.5.1
	Version du 24/09/2011

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.

iii. 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.

iv. 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 :

	Titre du document : Compte rendu d'étude
	Référence : L 1.5.1
	Version du 24/09/2011

- \$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.

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 pointe 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'.

	Titre du document : Compte rendu d'étude
	Référence : L 1.5.1
	Version du 24/09/2011

E.Problèmes posés

Les projets, utilisateurs et artefacts stockés dans les forges sont autant d'entités qui participent à des processus de développement logiciel, à des processus d'assurance qualité, ou à différents processus d'animation des communautés de développement distribuées. Classiquement, ces entités peuvent être intégrées "nativement" dans les outils fournis par la forge, où ils sont stockés en base de données, par exemple, et rendus visible via les interfaces (Web) de la forge. Si la forge fournit des outils de support des processus décrits ci-dessus, ces artefacts sont accessibles via des vues et dialogues de sélection appropriés. Typiquement, les URL d'accès à ces vues dans les interfaces Web importent peu, et différentes URL peuvent éventuellement permettre l'accès aux mêmes entités, sans nécessité d'unification. La sélection, ou le référencement de ces entités dans les propriétés d'autres entités, ou dans les outils de conduite des processus supportés nativement par la forge se fait alors par une sélection dans des boîtes de dialogue, sans manipulation d'identifiants internes, ou d'URLs. Le nommage des entités est alors ergonomique et non-ambigü, pourvu que les dialogues en question soient correctement implémentés. Cette situation trouve sa limite lorsqu'on souhaite mettre en oeuvre des processus non-supportés nativement par les outils de la forge, ou lorsqu'on souhaite référencer des entités d'une forge depuis un autre outil.

i.Référencement par identifiants


Si les références d'accès à ces entités consistent en des identifiants internes, l'ergonomie est discutable (qu'est-ce que le bug numéro #1234, ? à quel projet appartient-il ?, etc.). Si la forge détecte des mentions littérales de tels identifiants dans les saisies faites par les utilisateurs (par la détection de conventions de syntaxe, par exemple '#' suivi par des numéros), elle est capable de reconstruire une référence plus lisible pour un utilisateur dans les vues qu'elle construit (en ajoutant le titre du rapport correspondant, par exemple). La portée de ces identifiants est alors à prendre en compte, en cas de déplacement, copier-coller, ou export des données : les identifiants contextuels doivent éventuellement être alors renumérotés. Par exemple, si un numéro de bug est unique au sein d'un même tracker, la réaffectation d'un rapport qui référence ce numéro entre deux trackers peut alors casser les liens.

ii.URLs ambigües

Les référencer à l'aide de leurs URLs est alors également problématique, si ces URLs ne sont pas constantes pour référencer la même entité. Par exemple http://maforge.com/tracker/?bug_id=1234&group_id=4 référence-t-elle la même entité (un rapport de bugs) que http://maforge.com/tracker/?group_id=4&bug_id=1234, ou même que http://maforge.com/tracker/?bug_id=1234 ? On souhaite améliorer l'accessibilité, l'identification, le nommage des entités, de façon à diminuer les ambiguïtés, améliorer l'interopérabilité entre outils, et donc permettre le support de nouveaux processus, tout en conservant une ergonomie maximale pour l'utilisateur.

iii.Accès à des données privées ?

Si l'identifiant porte une partie significative des informations privées (un titre), la mention d'une URL vers cet élément peut être problématique. Un mécanisme d'"obfuscation" peut être mis en oeuvre, mais qui nuit alors à l'ergonomie de la solution. Exemple : anonymisation des mails des utilisateurs.

	Titre du document : Compte rendu d'étude
	Référence : L 1.5.1
	Version du 24/09/2011

iv. Versionnement des entités

Référence à un élément qui change (renommage), par exemple un commit sur un fichier : le nom du fichier n'est pas constant dans un VCS. De la même manière, un document ou une page wiki peuvent exister sous plusieurs versions différentes dans une forge. On peut résoudre ce problème en permettant de préciser la version demandée, ou une date, dans le lien. Ceci permet de retourner la bonne version, et, lorsque cette dernière n'est pas précisée, de retourner la version courante.

F. Solutions envisageables

i. URLs REST pour l'accès aux entités de la forge

REST (Representational State Transfer [1]) est un paradigme de construction d'applications Web permettant de mettre en oeuvre l'accès (en lecture ou modification) aux ressources via des URL standardisées et identiques qu'on soit en lecture ou écriture. Les ressources sont alors structurées le long d'une arborescence d'accès (potentiellement non unique) aux ressources. Par exemple, l'accès au rapport de bug d'un gestionnaire de bugs d'un projet particulier peut se faire via une URL du type :

`http://maforge.com/project/<NOM_COURT_PROJET>/tracker/<NOM_COURT_TRACKER>/ticket/<NUMERO_TICKET>` ou bien directement en tant que : `http://maforge.com/project/<NOM_COURT_PROJET>/ticket/<NUMERO_TICKET>` par exemple, si le numéro de ticket est unique au sein d'un projet, permettant ainsi sa réaffectation, sans avoir à le renuméroter, entre différents trackers du même projet (par exemple en cas de conversion d'un rapport de bug en demande d'évolution)

Pour info, l'API Redmine Rest, très simple est détaillé ici : http://www.redmine.org/projects/redmine/wiki/Rest_api

exemple : Showing an issue

GET /issues/[id].xml

fonctionne aussi en json


ii. URIs pour l'identification des ressources

Mécanisme générique de référence de ressources (à la Linked Data)

iii. Miscealeanous

cf. Provide standardized path-like REST friendly URLs for tools, views, artifacts (projects, users) of the forge feature request for FusionForge


Interface WSDL de Fusionforge (voir aussi Forges_SOAP_APIs)

	Titre du document : Compte rendu d'étude
	Référence : L 1.5.1
	Version du 24/09/2011

```

login
logout
version
getGroups ('group_ids'=>'tns:ArrayOfint') : ArrayOfGroup
getGroupsByName ('group_names'=>'tns:ArrayOfstring') : ArrayOfGroup
getPublicProjectNames () : ArrayOfstring
getUsers ('user_ids'=>'tns:ArrayOfint') : ArrayOfUser
getUsersByName ('user_ids'=>'tns:ArrayOfstring') : ArrayOfUser
OK userGetGroups ('user_id'=>'xsd:int') : ArrayOfGroup
getArtifactTypes
getArtifacts
addArtifact
updateArtifact
getArtifactFiles
getArtifactFileData
addArtifactFile
artifactFileDelete
getArtifactMessages
addArtifactMessage
getArtifactTechnicians
artifactSetMonitor
artifactIsMonitoring
artifactDelete
artifactTypeIsMonitoring
artifactGetChangeLog
artifactGetViews
artifactDeleteView
artifactSetView
artifactCreateView
artifactUpdateView
getDocumentStates
getDocumentLanguages
addDocument
updateDocument
addDocumentGroup
updateDocumentGroup
getDocuments
getDocumentGroups
getDocumentGroup
getDocumentFiles
documentDelete
getProjectGroups
getProjectTasks
addProjectTask
updateProjectTask
getProjectTaskCategories
getProjectMessages
addProjectMessage
getProjectTechnicians
addTimeEntry
getPackages
getFileTypes
getFileProcessorTypes
addPackage
getReleases
addRelease
getFiles


```

	Titre du document : Compte rendu d'étude
	Référence : L 1.5.1
	Version du 24/09/2011


getFile
 addFile
 addUploadedFile
 getSCMData

Début de recensement

FusionForge function	Standardized URL	FusionForge URL	Codendi URL	SOAP Codendi function	SOAP FusionForge function
New account (Registering as a new user)		/account/register.php	/account/register.php		
Register project		/register/	/project/register.php		
User Profile	/users/<USERNAME>	/users/<USERNAME>/	/users/<USERNAME>		
User's projects				getMyProjects	userGetProjects ('user_id' : ArrayOfInt)
Project list	/projects	/softwaremap/full_list.php	/softwaremap/		
Project Summary	/projects/<PROJECTNAME>	/projects/<PROJECTNAME>/	/projects/<PROJECTNAME>/		
Project Activity		/activity/?group_id=<GROUPID>	N/A		
Project Forums	/projects/<PROJECTNAME>/forums/	/forum/?group_id=<GROUPID>	/forum/?group_id=<GROUPID>		
Project Forum (Threads)	/projects/<PROJECTNAME>/forums/<FORUMID>	/forum/forum.php?forum_id=<FORUMID>	/forum/forum.php?forum_id=<FORUMID>		
Project Forum Thread	.../threads/<THREADID> or /projects/<PROJECTNAME>/threads/<THREADID>	/forum/forum.php?thread_id=<THREADID>&forum_id=<FORUMID>			

	Titre du document : Compte rendu d'étude
	Référence : L 1.5.1
	Version du 24/09/2011

Messages)	NAME>/fthreads/<THREADID>	ORUMID>			
Project Forum Thread Message >	.../msg/<MESSAGEID> or /projects/<PROJECTNAME>/fmsg/<MESSAGEID>	/forum/message.php?msg_id=<MESSAGEID>	/forum/message.php?msg_id=<MESSAGEID>		
Project Trackers >	/projects/<PROJECTNAME>/trackers	/tracker/?group_id=<GROUPID>	/tracker/index.php?group_id=<GROUPID>	getTrackerList	
Project Tracker Artifacts >	.../trackers/<TRACKERID> or .../tracker/<TRACKERNAME>	/tracker/?atid=<TRACKERID>&group_id=<GROUPID>(&func=browse)	/tracker/?tracker=<TRACKERID>	getArtifacts	getArtifacts
Project Tracker Add artifact >		/tracker/?func=add&group_id=<GROUPID>&atid=<TRACKERID>	/tracker/?tracker=<TRACKERID>&func=new-artifact	addArtifact	addArtifact
Project Tracker Artifact >	.../artifacts/<ARTIFACTID> or /projects/<PROJECTNAME>/artifacts/<ARTIFACTID>	/tracker/index.php?func=detail&aid=<ARTIFACTID>&group_id=<GROUPID>&atid=<TRACKERID>	/tracker/?aid=<ARTIFACTID>	getArtifactById	
Project Lists >		/mail/?group_id=<GROUPID>	/mail/?group_id=<GROUPID>		
project > List About (mailman)		lists./mailman/listinfo/<LISTNAME>	/mail/?group_id=<GROUPID>&action=listinfo&id=<LISTID>		
Project Tasks Subprojects >		/pm/?group_id=6	N/A		getProjectTasks
Project Tasks Subproject Tasks >		/pm/task.php?group_project_id=<SUBPROJECTID>&group_id=<GROUPID>&func=browse	N/A		getProjectTasks
Project >		/docman/?	/plugins/docman/?		

	Titre du document : Compte rendu d'étude
	Référence : L 1.5.1
	Version du 24/09/2011


Docs		group_id=<GROUPI D>	group_id=<GROU PID>		
Project > Surveys		/survey/? group_id=<GROUPI D>	/survey/? group_id=<GROU PID>		
Project > News		/news/? group_id=<GROUPI D>	/news/? group_id=<GROU PID>		
Project > SCM		/scm/? group_id=<GROUPI D>	/svn/? group_id=<GROU PID>, /cvs/? group_id=<GROU PID>, /plugins/git/? group_id=<GROU PID>		
Project > File Release System (FRS)		/frs/? group_id=<GROUPI D>	/file/showfiles.php? group_id=<GROU PID>		
Project > FRS > Release Notes and Changelog		/frs/shownotes.php? release_id=<RELEA SEID>	/ file/shownotes.php? release_id=<RELE ASEID>		
Project > FRS > Release > Files		/frs/? group_id=<GROUPI D>&release_id=<RE LEASEID>	N/A	getFiles	
Project > FRS > Release > File		/ frs/download.php/2/< FILENAME>	/ file/download.php/ <GROUPID>/<PA CKAGEID>/<REL EASEID>/<FILEN AME>	getFile	

G.Perspectives

Les IHM permettent la saisie des liens avec des syntaxes simplifiées (type moteur de références Codendi), mais les références sont stockées sur la base de liens URI.

Lorsqu'on copie le lien vers lequel le lien pointe, on a une URL correspondant à une URI standardisée, même si l'affichage mentionne toujours "bug # 123" dans la page web.

Derrière ces URL, on a un mécanisme de compact-preview type OSLC par content-negotiation permettant l'affichage de popups contenant des représentations succincte des ressources en

	Titre du document : Compte rendu d'étude
	Référence : L 1.5.1
	Version du 24/09/2011

question.

On combine alors :

- * une saisie simplifiée et mnémotechnique
- * une URL standardisée
- * des interfaces dynamiques avec AJAX