

PLAN DE TESTS

Ref. document VISHNU_D2_1b-UMS-PlanTests
Objet du document Référentiel du plan de tests du module UMS du projet Vishnu

TABLE DES REVISIONS

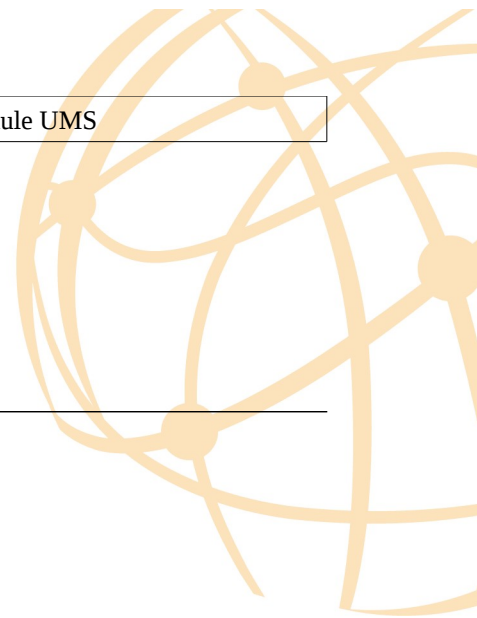
Version	Date	Modifications	Auteur
V1.0	25/01/11	Version initiale	Benjamin Isnard, Daouda Traoré, Eugène Pamba Capo-Chichi

VALIDATION DU DOCUMENT

Date	Personne

REFERENCES

Ref. document	Description
VISHNU_D1_1e-POL-TESTS	Politique de tests du projet VISHNU
VISHNU_D1_1a-SPECSGEN	Spécifications générales du projet VISHNU
VISHNU_D1_1b-STB	Spécifications Techniques des besoins



1 Présentation du document

1.1 Objectifs du document

Les objectifs de ce document sont les suivants :

- Présenter les fonctionnalités du module UMS à tester.
- Présenter les différents types de tests permettant de valider les cas d'utilisation de chaque fonctionnalité du module UMS.
- Présenter les méthodes et outils utilisés pour ces tests.
- Donner des détails de l'environnement de test

1.2 Structure du document

Le chapitre 1 présente le document.

Le chapitre 2 présente les environnements et outils de tests.

La chapitre 3 décrit la liste des tests qui s'appliquent aux fonctionnalités à tester.

La chapitre 4 présente les outils utilisés pour ces tests.

1.3 Glossaire, abréviations et acronymes

1.3.1 Glossaire

- **Script de test** : c'est un programme qui contient les instructions permettant de réaliser les tests.
- **Certificat** : correspond à la clé de session (identifiant de session crypté)

1.3.2 Acronymes

- STB : Spécification Technique des Besoins
- DSG : Document de Spécification générale
- DSD : Document de Spécification Détaillées
- API : Application Programming Interface (Interface de programmation d'applications)
- MPI : Message Passing Interface

1.3.3 Abréviations

- Id. Identifiant
- Réf. Référence
- Rel.# Numéro de release/version
- Resp. Responsable(s)

2 Environnements et outils de tests

2.1 Environnements de tests

Pour ces tests nous allons utiliser les deux environnements tests suivants :

Environnement test 1:

- Système d'exploitation : c'est le système linux qui sera installé sur toutes les machines clientes et serveurs.
- Nombre de machines clientes : au moins 1 machine cliente décrite dans [STB] pour pouvoir lancer des tests à distance (selon le cas de test).
- Deux serveurs UMS décrit dans [STB].
- Un serveur SysFera-DS dédié décrit dans [STB].
- Un serveur de base de donnée Oracle 11.2.0.2.
- Un serveur web JBOSS décrit dans [STB].

Environnement test 2:

- Système d'exploitation : c'est le système linux qui sera installé sur toutes les machines clientes et serveurs.
- Nombre de machines clientes : au moins 1 machine cliente décrite dans [STB] pour pouvoir lancer des tests à distance (selon le cas de test).
- Deux serveurs UMS décrit dans [STB].
- Un serveur SysFera-DS dédié décrit dans [STB].
- Un serveur de base de donnée PostgreSQL.
- Un serveur web JBOSS décrit dans [STB].

Pour chacun des environnements, nous allons utiliser une base de données avec 100 utilisateurs, 10 machines serveurs, 10 000 sessions et 100 000 enregistrements de commandes utilisateurs.

2.2 Outils de tests

Les outils suivants seront utilisés pour les tests :

- Hudson (serveur d'intégration) sera utilisé pour les tests d'intégration continue (compilation, installation).
- Boost test sera utilisé pour la réalisation des tests unitaires.
- Des scripts batchs seront utilisés pour la réalisation des scripts de tests, des tests fonctionnels.
- Square et Logiscope seront utilisés régulièrement pour la qualité du code.

3 Cycle de vie du test

Le schéma suivant donne une vue du plan de test sur un test à réaliser jusqu'à la réalisation du rapport de test :

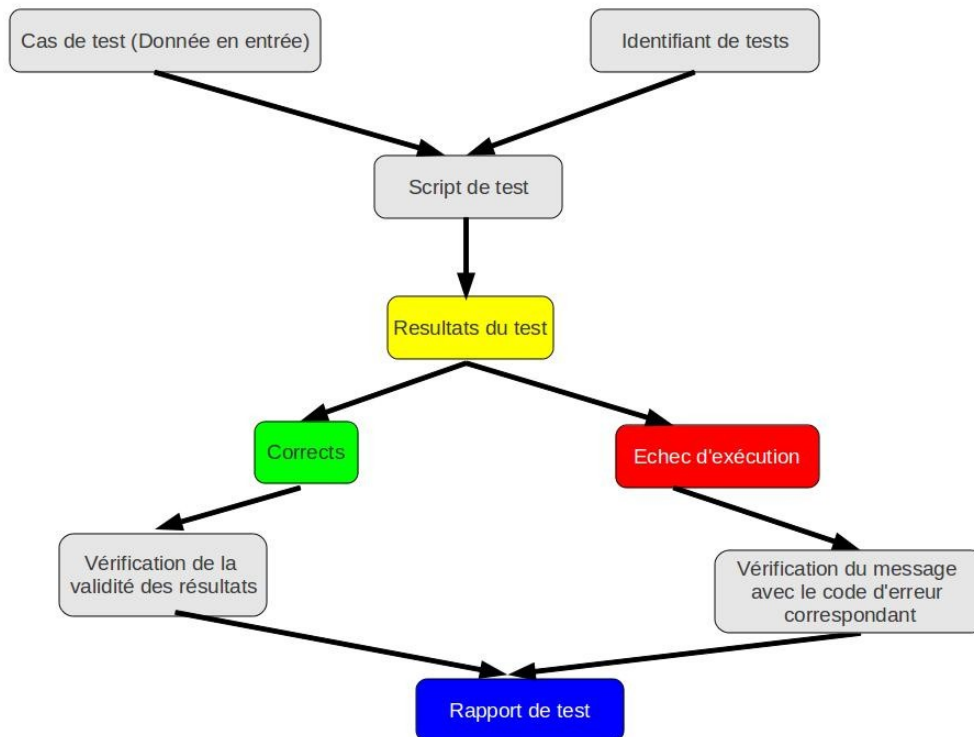


Illustration 1: Schema du test

4 Liste des tests

4.1 Tests fonctionnels

Les tests fonctionnels ou tests de boîtes noires permettent à partir des données en entrée d'observer les réactions du logiciel à tester.

Les fonctionnalités (*fonctions API/C++ et API/Python, ligne de commande utilisateur, API web service*) sont classées par catégories de tests. A chaque catégorie correspond un ensemble de fonctionnalités. Les catégories de tests sont les suivantes :

- Catégorie 1 : regroupe les fonctionnalités permettant la gestion des sessions qui sont : *connect, reconnect, listSessions, listHistoryCmd* et *close*.
- Catégorie 2 : regroupe les fonctionnalités permettant la gestion des utilisateurs qui sont : *addVishnuUser, updateUser, deleteUser, listUsers, changePassword* et *resetPassword*.
- Catégorie 3 : regroupe les fonctionnalités permettant la gestion des configurations qui sont : *addLocalAccount, updateLocalAccount, deleteLocalAccount, ListLocalAccount saveConfiguration, restoreConfiguration, configureDefaultOption* et *listOptions*.
- Catégorie 4 : regroupe les fonctionnalités permettant la gestion des machines qui sont : *addMachine, updateMachine, deleteMachine* et *listMachines*.

Nous allons dans cette section présenter les différentes stratégies de tests en utilisant des programmes exécutables ou des fonctions de l'API UMS puis en observant les résultats obtenus. Les deux conditions à vérifier sont :

- Est ce que le programme réalise les fonctions attendues en accord avec les spécifications?
- Est ce que le programme fonctionne correctement ?

4.1.1 Identifiant de tests

Les identifiants des tests sont de la forme: <ID DSG>-[B|E] où 'ID DSG' est l'identifiant du cas d'utilisation dans le document [SDG] , 'B' la séquence de base (exécution du cas d'utilisation avec des paramètres valides) et 'E' la séquence d'exécution (comportement du cas d'utilisation en cas d'erreurs)

4.1.2 Tests de la Catégorie 1

Identifiant de tests	Actions	Résultats	Considérations particulières
U1-B1	<ol style="list-style-type: none"> Cas de test : les paramètres sont corrects et enregistrés dans la base de données. Vérifier qu'il n'y a aucune session ouverte sur la machine de test. Sinon désactiver la session ouverte. Lancer le programme connect avec comme paramètres le userId (login) et le mot de passe de l'utilisateur. Un certificat de session est créé et envoyé à l'utilisateur. Lancer la commande listSessions avec la valeur du certificat retourné. Si le certificat est valide, l'exécution de listSessions réussit donc la connexion est validée. 	-Un certificat valide de session est créée.	La durée d'inactivité de la session doit avoir une valeur assez élevée pour permettre l'exécution du test dans de bonnes conditions
U1-E1 U1-E2	<ol style="list-style-type: none"> Cas de test : paramètres non valides <ul style="list-style-type: none"> ● Cas E1 : login inconnu ● Cas E2 : mot de passe incorrect Vérifier qu'il n'y a aucune session ouverte sur la machine de test. Sinon désactiver la session ouverte. Lancer le programme connect avec le cas de test correspondant (E1 ou E2) Lancer une commande (listSessions) et vérifier que son exécution échoue. 	- Un message correspondant au cas d'erreur est affiché	
U1.1-B1 U1.1-B2	<ol style="list-style-type: none"> Cas de test : les paramètres sont corrects et enregistrés dans la base de données. <ul style="list-style-type: none"> ● Cas B1-connect : l'option CLOSE_ON_DISCONNECT fixée à vrai ● Cas B2-connect : l'option CLOSE_ON_TIMEOUT est fixée à vrai Vérifier qu'il n'y a aucune session ouverte sur la machine 	<p>Cas B1-connect :</p> <ul style="list-style-type: none"> - La session a été fermée automatiquement après la fermeture du poste de travail. <p>Cas B2 :</p> <ul style="list-style-type: none"> - La session a été 	

	<p>de test. Sinon désactiver la session ouverte.</p> <ol style="list-style-type: none"> Lancer le programme connect avec comme paramètres le login et le mot de passe de l'utilisateur. Un certificat de session est créé et envoyé à l'utilisateur. Lancer la commande listSessions avec la valeur du certificat retourné. Si le certificat est valide, l'exécution de listSessions réussit. <p>Pour le cas B1 :</p> <ol style="list-style-type: none"> Fermer le poste sur lequel le programme connect a été lancé. Démarrer le poste client et vérifier l'exécution d'un programme échoue (ex : listSessions) . <p>Pour le cas B2 :</p> <ol style="list-style-type: none"> Ne lancer aucun programme pendant un certain temps (temps supérieur au temps d'inactivité fixé). Lancer la commande listSessions et vérifier que son exécution échoue. 	fermée automatiquement après le temps d'inactivité fixé	
U1.1-E1 U1.1-E2 U1.1-E3 U1.1-E4	<ol style="list-style-type: none"> Cas de test : paramètres non valides <ul style="list-style-type: none"> Cas E1 : login inconnu Cas E2 : mot de passe incorrect Cas E3 : l'option de fermeture est inconnue Cas E4 : le login utilisateur par l'administrateur pour substituer le mot de passe est inconnu. Vérifier qu'il n'y a aucune session ouverte sur la machine de test. Sinon désactiver la session ouverte. Lancer le programme connect avec le cas de test correspondant. Aucun certificat de session n'est créé. Lancer une commande (listSessions) et vérifier que son exécution échoue. 	- Le message d'erreur correspondant au cas de test est affiché	
U1.5-B	<ol style="list-style-type: none"> Cas de test : paramètres valides. Lancer le programme connect dans deux environnements capables de se connecter à VISHNU pour avoir deux identifiants de session dans la base de données par le même utilisateur. Faire un listSessions dans chacun des environnements et récupérer l'identifiant de session. Essayer de se reconnecter en utilisant l'identifiant de session de l'autre environnement. Vérifier que la reconnexion s'est bien passée dans les deux environnements. 	- Le certificat courant a été réinitialisé au certificat correspondant à l'identifiant de session et le login donné en paramètre.	
U1.5-E	<ol style="list-style-type: none"> Cas de test : paramètres non valides 	- Un message	

	<p>Cas 1 : le login incorrect et l'identifiant de session correct Cas 2 : le login correct et l'identifiant de session incorrect</p> <p>2. Lancer le programme reconnect</p> <p>3. Son exécution échoue</p>	correspondant au cas d'erreur est affiché	
U1.3.5-B	<p>1. Cas de test : connexion en tant qu'utilisateur VISHNU</p> <p>2. Ouvrir plusieurs sessions en lançant les commandes connect et close l'une après l'autre.</p> <p>3. Lancer la commande <i>listSessions</i> sans paramètre.</p> <p>4. Inspecter la base de données pour vérifier que le nombre de sessions retournées par la commande <i>listSessions</i> est égal au nombre de sessions dans la base de données, que chaque identifiant est bien enregistré dans la base de données.</p> <p>5. Désactiver la session courante en utilisant la commande <i>close</i>.</p>	- La liste des sessions de l'utilisateur est affichée.	
U1.3.5-E	<p>1. Cas de test : paramètres non valides Cas 1 : le certificat de session est incorrect Cas 2 : le login de l'utilisateur n'est pas correct</p> <p>2. Lancer le programme listSessions</p> <p>3. Son exécution échoue</p>	- Un message d'erreur correspondant au cas de test est affiché	
UA5.1-B	<p>1. Cas de test : connexion en tant qu'administrateur de VISHNU</p> <p>2. Ouvrir plusieurs sessions en lançant les programmes connect et close l'une après l'autre.</p> <p>3. Lancer le programme listSessions sans paramètre.</p> <p>4. Inspecter la base de données pour vérifier que le nombre de sessions retournées par la commande <i>listSessions</i> est égal au nombre de sessions dans la base de données, que chaque identifiant est bien enregistré dans la base de données.</p> <p>5. Désactiver la session courante en utilisant le programme <i>close</i>.</p>	- La liste des sessions de tous les utilisateurs est affichée.	
UA5.1-E	<p>1. Cas de test : paramètres non valides Cas 1 : le certificat de session est incorrect Cas 2 : le login de l'utilisateur n'est pas correct</p> <p>2. Lancer le programme listSessions</p> <p>3. Son exécution échoue</p>	- Le message d'erreur correspondant au cas de test est affiché	
UA5.2-B	<p>1. Cas de test : connexion en tant qu'administrateur de VISHNU</p> <p>2. Ouvrir plusieurs sessions en lançant les programmes connect et close l'une après l'autre.</p>	- La liste des sessions de tous les utilisateurs est affichée.	

	<ol style="list-style-type: none"> Lancer le programme listSessions sans paramètre. Inspecter la base de données pour vérifier que le nombre de sessions retournées par la commande <i>listSessions</i> est égal au nombre de sessions dans la base de données, que chaque identifiant est bien enregistré dans la base de données. Désactiver la session courante en utilisant le programme <i>close</i>. 		
UA5.2-E	<ol style="list-style-type: none"> Cas de test : paramètres non valides Cas 1 : le certificat de session est incorrect Cas 2 : le login de l'utilisateur n'est pas correct Lancer le programme listSessions Son exécution échoue 	- Le message d'erreur correspondant au cas de test est affiché	
U1.2-B	<ol style="list-style-type: none"> Lancer le programme connect pour ouvrir une session. Lancer le programme close sans paramètres. Vérifier que la commande listSession ou listHistorycmd échoue. 	- La session a été désactivée, aucune commande ne peut être lancée dans la session	
U1.2-E1 U1.2-E2	<ol style="list-style-type: none"> Cas de test : paramètres non valides Cas E1 : le certificat de session n'est valide Cas E2 : la session est déjà fermée. Cas E3 : le certificat est incompatible avec le login de l'utilisateur Lancer le programme listSessions Son exécution échoue 	- Le message d'erreur correspondant au cas de test est affiché	
U1.3.4-B	<ol style="list-style-type: none"> Ouvrir une session et lancer la commande <i>listSession</i> plusieurs fois. Lancer la commande <i>listHistoryCmd</i> sans paramètres Inspecter la base de données pour vérifier que le nombre de commandes retournées par la commande <i>listHistoryCmd</i> est égal au nombre de <i>listHistoryCmd</i> est égal au nombre de commandes dans la base de données, que chaque l'identifiant de chaque commande est bien enregistré dans la base de données. Désactiver la session courante en utilisant la commande <i>close</i>. 	- L'historique des commandes est bien affichée.	
U1.3.4-B	<ol style="list-style-type: none"> Cas de test : paramètres non valides Cas E1 : le certificat de session n'est valide Lancer le programme listHistoryCmd Son exécution échoue 	- Un message correspondant au cas d'erreur est affiché	

4.1.3 Tests de la catégorie 2

Identifiant de tests	Actions	Résultats	Considérations particulières
U4-B	<ol style="list-style-type: none"> Cas de test : paramètres valides et l'utilisateur est un administrateur du système VISHNU. Ouvrir une session sur la machine à l'aide du programme connect. Lancer la commande listUsers et vérifier que l'utilisateur n'est pas déjà enregistré dans la base. Lancer la commande avec l'identifiant qui sera affecté au futur utilisateur. Lancer la commande listUser pour vérifier l'ajout de l'utilisateur. Désactiver la session courante à l'aide de la commande close. 	- Le nouvel utilisateur est créé et ajouté dans la base de données.	
U4-E1 U4-E2 U4-E3 U4-E4 U4-E5	<ol style="list-style-type: none"> Cas de test : paramètres non valides <ol style="list-style-type: none"> Cas E1 : le login de l'utilisateur est inconnu Cas E2 : le nom de la machine est inconnu Cas E3 : l'e-mail de l'utilisateur est invalide Cas E4 : l'utilisateur existe déjà Cas E5 : l'utilisateur n'est pas un administrateur VISHNU Lancer le programme addVishnuUser avec les paramètres correspondant à chaque cas. L'exécution de la commande échoue et un message d'erreur est affiché. Pour les cas : E1, E2, E3 <ol style="list-style-type: none"> Lancer la commande listUsers et vérifier que l'utilisateur n'est pas dans la liste. Pour le cas : E4 <ol style="list-style-type: none"> Lancer la commande listUsers et vérifier que l'utilisateur n'est pas ajouté deux fois dans la liste. 	- Un message d'erreur correspond au cas d'erreur est affiché	
U4.1-B	<ol style="list-style-type: none"> Cas de test : paramètres valides et l'utilisateur est un administrateur du système VISHNU. Ouvrir une session sur la machine à l'aide du programme connect. Lancer la commande listUsers pour vérifier la présence de l'utilisateur dans la base de données. Lancer la commande avec le nouvel identifiant à donner à l'utilisateur. L'identifiant de l'utilisateur est changé Désactiver la session courante à l'aide de la commande close. 	- Les informations sur l'utilisateur ont été mises à jour	

U4.1-E1 U4.1-E2 U4.1-E3	<ol style="list-style-type: none"> Cas de test : paramètres non valides <ol style="list-style-type: none"> Cas E1 : le login de l'utilisateur est inconnu Cas E2 : le nom de la machine est inconnu Cas E5 : l'utilisateur n'est pas un administrateur VISHNU Lancer le programme updateUser avec les paramètres correspondant à chaque cas. L'exécution de la commande échoue et un message d'erreur est affiché. Lancer la commande listUsers et vérifier que l'utilisateur n'est pas dans la liste. 	- Un message correspondant au code d'erreur est affiché	
U4.2-B	<ol style="list-style-type: none"> Cas de test : paramètres valides et l'utilisateur est un administrateur du système VISHNU. Ouvrir une session sur la machine à l'aide de la commande connect. Lancer la commande listUsers pour vérifier que l'utilisateur existe bien dans la liste. Lancer le programme deleteUser avec les paramètres de test Lancer le programme listUsers pour vérifier que l'utilisateur n'existe plus dans la liste. 	- L'utilisateur est supprimé dans VISHNU.	
U4.2-E1 U4.2-E2	<ol style="list-style-type: none"> Cas de test : paramètres non valides <ol style="list-style-type: none"> Cas E1 : le nom de la machine est inconnu Cas E2 : l'utilisateur n'est pas un administrateur VISHNU Lancer le programme deleteUser avec les paramètres correspondant à chaque cas. L'exécution de la commande échoue et un message d'erreur est affiché. 	- Un message correspondant au cas d'erreur est affiché	
U4.3-B	<ol style="list-style-type: none"> Cas de test : paramètres valides et l'utilisateur est un administrateur du système VISHNU. Ouvrir une session sur la machine à l'aide de la commande connect. Lancer la commande listUsers, récupérer le nombre d'utilisateurs déjà enregistrés dans la base. Enregistrer cette liste dans un fichier. Créer plusieurs utilisateurs à l'aide de la commande addVishnuUser. Ajouter la liste de ces nouveaux utilisateurs dans la liste du fichier. Exécuter la commande listUsers et vérifier que la taille de la liste enregistrée dans le fichier est égale à celle trouvée par listUsers, le contenu des listes est le même. 	- La liste des utilisateurs enregistrés est affichée.	

	6. Désactiver la session courante à l'aide de la commande close . 7. Créer (<i>addVishnuUser</i>) plusieurs utilisateurs avec toutes les informations nécessaires sur chaque utilisateur. Enregistrer chaque utilisateur avec ces informations dans un fichier. 8. Exécuter la commande listUsers avec l'option -i et l'identifiant d'un utilisateur choisi aléatoirement dans la liste du fichier sauvegardé. Vérifier que les informations obtenues sont les mêmes que celles de l'utilisateur dans le fichier. 9. Désactiver la session courante à l'aide de la commande close .		
U4.3-E1 U4.3-E2	4. Cas de test : paramètres non valides <ol style="list-style-type: none"> Cas E1 : le nom de la machine est inconnu Cas E2 : l'utilisateur n'est pas un administrateur VISHNU 5. Lancer le programme deleteUser avec les paramètres correspondant à chaque cas. 6. L'exécution de la commande échoue et un message d'erreur est affiché.	- Le message d'erreur correspondant au cas d'erreur est affiché	
U1.3.3-B	1. Cas de test : paramètres valides et l'utilisateur est un administrateur du système VISHNU. 2. Lancer la commande listUsers pour vérifier la présence de l'utilisateur dans la base de données. 3. Lancer la commande changePassword avec l'identifiant de l'utilisateur, son ancien mot de passe et son nouveau mot de passe. 4. Vérifier que l'utilisateur peut lancer une commande (par exemple connect) avec son login et son nouveau mot de passe.	- Le mot de passe de l'utilisateur est changé	
U1.3.3-E	7. Cas de test : paramètres non valides <ol style="list-style-type: none"> Cas E1 : le nom de la machine est inconnu 8. Lancer le programme ChangeUser avec les paramètres correspondant à chaque cas. 9. L'exécution de la commande échoue et un message d'erreur est affiché.	- Un message correspondant au cas d'erreur est affiché	
UA2-B	1. Cas de test : paramètres valides et l'utilisateur est un administrateur du système VISHNU. 2. Ouvrir une session sur la machine à l'aide de la commande connect . 3. Exécuter les étapes 1 à 2 de l'Id <i>changePassword en</i>	- Le mot de passe de l'utilisateur a été mis à jour	

	<i>remplaçant la commande <code>changePassword</code> par <code>resetPassword</code>.</i> 4. Le mot de passe de l'utilisateur est réinitialisé. 5. Vérifier que l'utilisateur peut lancer une commande (par exemple <code>connect</code>) avec son login et son nouveau mot de passe. 6. Désactiver la session courante à l'aide de la commande <i>close</i> .		
UA2-E1 UA2-E2	10. Cas de test : paramètres non valides 1. Cas E1 : le nom de la machine est inconnu 2. Cas E2 : l'utilisateur n'est pas un administrateur VISHNU 11. Lancer le programme <i>deleteUser</i> avec les paramètres correspondant à chaque cas. 12. L'exécution de la commande échoue et un message d'erreur est affiché.	- Un message correspondant au cas d'erreur est affiché	

4.1.4 Tests de la catégorie 3

Note : Lors du test de chacune des fonctions, une session sera ouverte (si aucune session n'est ouverte) à l'aide de la commande ***connect*** et fermer (à l'aide de la commande ***close***) à la fin des actions à mener pour réaliser le test.

Identifiant de tests	Actions	Résultats	Considérations particulières
U4-B	1. Cas de test : paramètres valides. 2. Lancer la commande <i>ListLocalAccount</i> et enregistrer les configurations déjà ajoutées. 3. Lancer la commande avec le login, le chemin vers la clé et le chemin vers le répertoire personnel de l'utilisateur et l'identifiant de la machine à laquelle la configuration sera rattachée. 4. Vérifier qu'une nouvelle configuration a été ajoutée à l'aide de la commande <i>ListLocalAccount</i> et la liste enregistrée précédemment à l'étape 2.	- Une nouvelle configuration a été créée pour l'utilisateur donné	
U4-E1 U4-E2	1. Cas de test : paramètres non valides 1. Cas E1 : le nom de la machine est inconnu 2. Cas E2 : l'utilisateur n'est pas connu 2. Lancer le programme <i>addLocalAccount</i> avec les paramètres correspondant à chaque cas.		

	3. L'exécution de la commande échoue et un message d'erreur est affiché.		
U4.1-B	<ol style="list-style-type: none"> Cas de test : paramètres valides. A l'aide de la commande <i>ListLocalAccount</i>, enregistrer les informations sur la configuration qui sera mise à jour. Pour chaque information à mettre à jour faire : <ul style="list-style-type: none"> Lancer la commande avec le login de l'utilisateur et l'identifiant de la machine à laquelle la configuration sera rattachée. A l'aide de la commande <i>listLocalAccount</i>, vérifier la mise à jour des informations en les comparant à celles enregistrées. 	- Les informations sur la configuration donnée ont été mises à jour	
U4.1-E	<ol style="list-style-type: none"> Cas de test : paramètres non valides <ol style="list-style-type: none"> Cas E1 : le nom de la machine est inconnu Cas E2 : l'utilisateur n'est pas connu Lancer le programme updateLocalAccount avec les paramètres correspondant à chaque cas. L'exécution de la commande échoue et un message d'erreur est affiché. 		
U4.2-B	<ol style="list-style-type: none"> Cas de test : paramètres valides. Lancer la commande avec le login de l'utilisateur et l'identifiant de la machine à laquelle la configuration sera rattachée. Vérifier que la configuration a été supprimée à l'aide de la commande <i>listLocalAccount</i>. 	- La configuration donnée a été supprimée.	
U4.2-E	<ol style="list-style-type: none"> Cas de test : paramètres non valides <ol style="list-style-type: none"> Cas E1 : le nom de la machine est inconnu Cas E2 : l'utilisateur n'est pas connu Lancer le programme deleteLocalAccount avec les 	- Un message correspondant au cas d'erreur est affiché	

	paramètres correspondant à chaque cas. 3. L'exécution de la commande échoue et un message d'erreur est affiché.		
UA3-B	<ol style="list-style-type: none"> Cas de test : paramètres valides et l'utilisateur est un administrateur du système VISHNU. Créer un fichier vide. Lancer la commande de sauvegarde. Lancer la commande <code>listLocalAccount</code>. Vérifier que le fichier contient l'ensemble des configuration. 	- La configuration donnée a été sauvegardée dans un fichier.	
UA3-E	<ol style="list-style-type: none"> Cas de test : paramètres non valides Cas E1 : le nom du fichier n'est pas connu Lancer le programme saveConfiguration avec les paramètres correspondant à chaque cas. L'exécution de la commande échoue et un message d'erreur est affiché. 	- Un message correspondant au cas d'erreur est affiché	
UA4-B	<ol style="list-style-type: none"> Cas de test : paramètres valides et l'utilisateur est un administrateur du système VISHNU. Lancer la commande listLocalAccount pour vérifier le status des configurations enregistrées. Lancer restoreConfiguration avec le fichier contenant la sauvegarde. Lancer la commande <code>listLocalAccount</code> pour vérifier que les configurations ont été restaurées. 	Le contenu du fichier donné a permis de restaurer la configuration.	
UA4-E	<ol style="list-style-type: none"> Cas de test : paramètres non valides 1. Cas E1 : le nom du fichier n'est pas connu Lancer le programme restoreConfiguration avec les paramètres correspondant à chaque cas. L'exécution de la commande échoue et un message d'erreur est affiché. 	- Un message correspondant au cas d'erreur est affiché	
U4.3-B	<ol style="list-style-type: none"> Cas de test : paramètres valides et l'utilisateur est un administrateur du système VISHNU. 	- La liste de l'ensemble des configurations est affichée.	

	<ol style="list-style-type: none"> Lancer la commande sans paramètre Inspecter la base de données pour vérifier que le nombre d'utilisateurs retourné par la commande listLocalAccount est égal au nombre de configurations dans la base de données, que chaque configuration est bien enregistrée dans la base de données. 		
U4.3-E	<ol style="list-style-type: none"> Cas de test : paramètres non valides <ol style="list-style-type: none"> Cas E1 : le nom de la machine est inconnu Cas E2 : l'utilisateur n'est pas un administrateur VISHNU Lancer le programme listLocalAccount avec les paramètres correspondant à chaque cas. L'exécution de la commande échoue et un message d'erreur est affiché. 	-Un message correspondant au cas d'erreur est affiché	
UA7-B	<ol style="list-style-type: none"> Cas de test : paramètres valides et l'utilisateur est un administrateur du système VISHNU. Lancer la commande <i>listOptions</i> pour avoir la liste des configurations par défaut. Lancer la commande <i>configureDefaultOption</i> avec le nom de l'option et sa valeur. Lancer la commande <i>listOptions</i> pour vérifier si la configuration par défaut a été mise à jour. 	- La configuration est fixée comme configuration par défaut.	
UA7-E1 UA7-E2	<ol style="list-style-type: none"> Cas de test : paramètres non valides <ol style="list-style-type: none"> Cas E1 : le nom de l'option n'est valide Cas E2 : la valeur de l'option n'est pas valide 	- Un message correspondant au code d'erreur est affiché	

	<ol style="list-style-type: none"> Lancer le programme configureDefaultOption avec les paramètres correspondant à chaque cas. L'exécution de la commande échoue et un message d'erreur est affiché. 		
U1.3.2-B	<ol style="list-style-type: none"> Cas de test : paramètres valides Lancer la commande sans paramètres Inspecter la base de données pour vérifier que le nombre d'options retourné par la commande <code>listOptions</code> est égal au nombre d'options dans la base de données, que l'identifiant de chaque option est bien enregistré dans la base de données. 	- L'ensemble des options est affiché.	
U1.3.2-E	<ol style="list-style-type: none"> Cas de test : paramètres non valides <ol style="list-style-type: none"> Cas E : le nom de l'option n'est pas connu Lancer le programme listOptions avec les paramètres correspondant à chaque cas. L'exécution de la commande échoue et un message d'erreur est affiché. 	- Un message correspondant au cas d'erreur est affiché	
U1.3.1-B	<ol style="list-style-type: none"> Cas de test : paramètres valides . Lancer la commande <code>listOptions</code> pour avoir la liste des options. Lancer la commande <code>configureOption</code> avec le nom de l'option et sa valeur. Lancer la commande <code>listOptions</code> pour vérifier si la configuration a été mise à jour. 	- L'option a été configurée	
U1.3.1-E1 U1.3.1-E2	<ol style="list-style-type: none"> Cas de test : paramètres non valides <ol style="list-style-type: none"> Cas E1 : le nom de l'option n'est valide Cas E2 : la valeur de l'option n'est pas valide Lancer le programme configureOption avec les paramètres correspondant à chaque cas. L'exécution de la commande échoue et un message d'erreur est affiché. 	- Un message d'erreur correspondant au cas d'erreur est affiché	

4.1.5 Tests de catégorie 4

Identifiant de tests	Actions	Résultats	Considérations particulières
UA6.1-B	<ol style="list-style-type: none"> Cas de test : paramètres valides et l'utilisateur est un administrateur du système VISHNU. Lancer la commande <i>listMachines</i> et enregistrer le nombre de machines déjà enregistrées. Lancer la commande avec le nom de la machine. Lancer la commande <i>listMachines</i> pour vérifier l'ajout de la machine. 	La nouvelle machine est créée.	
UA6.1-E	<ol style="list-style-type: none"> Cas de test : paramètres invalides -Cas E1 : le nom de la machine existe déjà Lancer la commande <i>addMachine</i> Son exécution échoue 	-Un message d'erreur est affiché	
UA6.4-B	<ol style="list-style-type: none"> Cas de test : paramètres valides et l'utilisateur est un administrateur du système VISHNU. Lancer la commande <i>listMachines</i> avec l'option -m pour avoir toutes les informations sur une machine. Lancer la commande avec le nom de la machine et l'information à mettre jour. Lancer la commande <i>listMachines</i> pour vérifier la mise à jour des informations sur la machine. 	Les informations sur la machine ont été mises à jour	
UA6.4-E	<ol style="list-style-type: none"> Cas de test : paramètres invalides -Cas E1 : le nom de la machine existe déjà Lancer la commande <i>updateMachine</i> Son exécution échoue 	- Un message d'erreur est affiché	
UA6.2-B	<ol style="list-style-type: none"> Cas de test : paramètres valides et l'utilisateur est un administrateur du système VISHNU. Lancer la commande <i>listMachines</i> avec l'option -t pour vérifier si la machine est présente. Lancer la commande avec l'identifiant de la machine Lancer la commande <i>listMachines</i> pour vérifier que la machine n'existe plus dans la liste. 	La machine est supprimée dans VISHNU.	
UA6.2-E	<ol style="list-style-type: none"> Cas de test : paramètres invalides -Cas E1 : le nom de la machine existe déjà 	- Un message d'erreur est affiché correspondant au cas	

	<ol style="list-style-type: none"> Lancer la commande <i>deleteMachine</i> Son exécution échoue 	d'erreur	
UA6.3-B	<ol style="list-style-type: none"> Cas de test : paramètres valides et l'utilisateur est un administrateur du système VISHNU. Lancer la commande sans paramètres Inspecter la base de données pour vérifier que le nombre de machines retourné par la commande <i>listMachine</i> est égal au nombre de machines dans la base de données, que chaque l'identifiant de machine est bien enregistré dans la base de données. 	La liste des machines est affichée	
UA6.3-E	<ol style="list-style-type: none"> Cas de test : paramètres invalides -Cas E1 : le nom de la machine existe déjà Lancer la commande <i>listMachines</i> Son exécution échoue 	- Un message d'erreur est affiché correspondant au cas d'erreur	

4.2 Tests de stress

4.2.1 Présentation

Ces tests permettent de vérifier que la cible de test peut réussir à basculer et à récupérer pour différents dysfonctionnements matériels, logiciels ou réseaux sans qu'il y ait perte de données. Ils vérifient que lorsqu'une défaillance survient un système alternatif ou un système de sauvegarde prends le relai correctement sans perte de données ou de transactions.

4.2.2 Environnement des tests

TODO préciser qu'on utilise l'environnement avec PostgreSQL

4.2.3 Identifiants de tests

Les identifiants des tests sont de la forme: STR-<Element stressé>-<Type de stress>.

4.2.4 Liste des tests

Identifiant de tests	Technique	Critère de complétion
STR-SERV-DOWN	<ul style="list-style-type: none"> Lancer la commande connect Arrêter le serveur UMS après l'exécution de la commande Redémarrer le serveur UMS Lancer la commande close 	- Terminaison normale de la commande close
STR-AGENT-DOWN	<ul style="list-style-type: none"> Lancer la commande connect Arrêter l'agent SysFera-DS après l'exécution 	- Terminaison normale de la commande close

	de la commande - Redémarrer l'agent SysFera-DS - Lancer la commande close	
--	--	--

4.3 Contrôle des performances

4.3.1 Présentation

Ces tests ont pour objectif de vérifier que les exigences de performance quantifiées sont effectivement satisfaites.

4.3.2 Environnement de tests

Nous utiliserons l'environnement de test 2 dans un environnement isolé (réseau et systèmes non partagés).

4.3.3 Identifiants des tests

Les identifiants des tests sont de la forme: CDP-<ID STB>-[NomdeCommande|IdentifiantDeTest] où 'ID STB' est l'identifiant de l'exigence spécifiée dans le document [STB] au paragraphe « performances, fiabilité et robustesse ».

4.3.4 Liste des tests

Identifiant de tests	Technique	Critère de completion
CDP-3.1-connect	- Lancer une commande 'ssh <host> uname' sur une machine serveur en mesurant le temps de réponse (temps de référence) - Lancer la commande connect en mesurant le temps de réponse	temps inférieur au temps de référence (valeur moyenne sur 10 itérations)
CDP-3.1-close	- Lancer la commande connect - Lancer une commande 'ssh <host> uname' sur une machine serveur en mesurant le temps de réponse (temps de référence). - Lancer la commande close en mesurant le temps de réponse	temps inférieur au temps de référence (valeur moyenne sur 10 itérations)
CDP-3.1-listSession, CDP-3.1-listHistoryCmd, CDP-3.1-listUsers, CDP-3.1-listMachines, CDP-3.1-listOptions	- Lancer la commande connect - Lancer une commande 'ssh <host> uname' sur une machine serveur en mesurant le temps de réponse (temps de référence). - Lancer la commande listSession/listHistoryCmd/listUsers/listMachines/listOptions en mesurant le temps de réponse	temps inférieur au temps de référence (valeur moyenne sur 10 itérations)
CDP-3.2.1-connect	- Lancer la commande connect 100 fois	Consommation en mémoire virtuelle sur le serveur UMS inférieure à 500 Mégaoctets.
CDP-3.2.2-connect	- Lancer le serveur UMS	Consommation CPU inférieure à 10%.

4.4 Tests de charge

4.4.1 Présentation

Les tests de charge ont pour objectif de vérifier que les exigences de capacité du système sont vérifiées.

4.4.2 Environnement de tests

Nous utiliserons l'environnement de test 2 avec un nombre de clients ou de serveurs tels que spécifié dans le cas de tests. Ces clients ou serveurs seront simulés sur une seule machine.

Par ailleurs, les requêtes envoyées en simultanée seront en fait envoyées par une seule machine cliente avec un programme de synchronisation (MPI).

4.4.3 Identifiants de tests

Les identifiants des tests sont de la forme: **LOAD-<ID STB>-[NomdeCommande|IdentifiantDeTest]** où 'ID STB' est l'identifiant de l'exigence spécifiée dans le document [STB].

4.4.4 Liste des tests

Identifiant de tests	Technique	Résultats attendus
LOAD-2.1-connect	- Lancer 100 commandes connect simultanées	- Terminaison normale de toutes les commandes - Etat normal du serveur UMS
LOAD-2.1-listSessions, LOAD-2.1-listHistoryCmd, LOAD-2.1-listUsers, LOAD-2.1-listMachines, LOAD-2.1-listOptions	- Lancer 100 commandes listSessions/ listHistoryCmd / listUsers / listMachines/ listOptions simultanées	- Terminaison normale de toutes les commandes - Etat normal du serveur UMS
LOAD-2.2-connect	- Ajouter 1000 utilisateurs dans la base de données - Lancer la commande connect	- Terminaison normale de toutes les commandes - Etat normal du serveur UMS
LOAD-2.3-connect	- Configurer 10 machines serveurs dans la base de données Vishnu - Lancer la commande connect	- Terminaison normale de toutes les commandes - Etat normal du serveur UMS
LOAD-2.4-connect	- Configurer 1000 machines serveurs dans la base de données Vishnu - Lancer la commande connect	- Terminaison normale de toutes les commandes - Etat normal du serveur UMS

4.5 Contrôle de la qualité du code

4.5.1 Présentation

L'objectif de ces tests est de vérifier que le code source (C/C++ ou Java) respecte les contraintes de qualité logiciel fixées dans les spécifications et le cahier des charges.

4.5.2 Identifiants des tests

Les identifiants des tests sont de la forme: [IdentifiantDeTest]-[IdentifiantDesModules] où les identifiants sont les suivants:

- Identifiants de tests
 - CRP: Conformité aux Règles de Programmation
 - MCC: Mesure de la Complexité du Code
 - DDC: Détection de Duplication du Code
- Identifiants de modules
 - CSC : C/C++ source code
 - JSC : Java source code

4.5.3 Liste des tests

Identifiant de tests	Technique	Résultats attendus
CRP-CSC	Audit du code C/C++ avec l'outil LOGISCOPE	Le code respecte toutes les règles de codage de catégorie «REQUIRED » spécifiées pour le projet [SCL] sauf dérogation clairement documentée dans le code.
CRP-JSC	Audit du code Java avec l'outil LOGISCOPE	Le code respecte toutes les règles de codage de catégorie «REQUIRED » spécifiées pour le projet [SCL] sauf dérogation clairement documentée dans le code.
MCC-CSC	Audit du code C/C++ avec l'outil SQUORE	Pas plus de 1% de composants classés « A risque »
MCC-JSC	Audit du code Java avec l'outil SQUORE	Pas plus de 1% de composants classés « A risque »
DDC-CSC	Audit du code C/C++ avec l'outil SQUORE	Moins de 10% de code dupliqué.
DDC-JSC	Audit du code Java avec l'outil SQUORE	Moins de 10% de code dupliqué.

4.6 Tests de compilation et d'installation

4.6.1 Présentation

La compilation des différents exécutables et bibliothèques composant le module UMS sera réalisée par un serveur d'intégration continue (du type HUDSON).

Par ailleurs les tests d'installation ont pour objectif de valider la procédure d'installation de l'ensemble des différents composants d'une plateforme VISHNU (clients, serveurs, agents SysFera-DS).

4.6.2 Environnement des tests

Nous utiliserons les deux environnements de tests pour la compilation et l'installation.

4.6.3 Liste des tests

Identifiant de tests	Technique	Résultats attendus
COMP-CSC	Compilation du code C/C++ avec l'outil GCC/CMake	Compilation correcte de tous les exécutables
INS-SERVER-PGSQL INS-SERVER-ORACLE	Installation des différents éléments serveur de la plateforme VISHNU <ul style="list-style-type: none"> • serveur de calcul • serveur de stockage • serveur dédié VISHNU • serveur de base de données 	- Tous les processus démarrés et opérationnels - Documentation disponible (pages man)
INS-CLI	Installation du client VISHNU en ligne de commande	- Toutes les commandes installées - Documentation disponible (pages man)
INS-PGAPI	Installation des librairies API VISHNU (C++ et Python)	- Librairie C++ installée - Librairie Python installée et pouvant être chargée dans l'interpréteur Python 2.x
INS-WSAPI	Installation des services web VISHNU sur le serveur JBoss	- Services opérationnels (contrôle via la console webservices de JBoss)

4.7 Contrôle de la documentation utilisateur

4.7.1 Présentation

L'objectif est de vérifier que la documentation utilisateur permet à l'utilisateur de comprendre et d'utiliser les fonctions spécifiées .

4.7.2 Liste des tests

Identifiant de tests	Technique	Résultats attendus
CDU-MAN	Vérifier les pages de manuel attachées à chaque commande	Toutes les pages de manuel contiennent toutes les informations nécessaires à l'utilisateur pour utiliser les commandes.
CDU-USERSGUIDE	Suivre le guide de l'utilisateur	Le guide permet une prise en main rapide des fonctions du logiciel