

Liste des tests unitaires liés au serveur LoGeAs

NB : Ces tests sont lié à des fonction spécifique du serveur logeas-web.

Ils ne sont plus suivi (au sens qualité) dans la mesure ou aucune évolution n'a été faite.

Ils devront être repris du manière ou une autre dans les prochaines versions

Formalisme des tests unitaires

Dans le mesure du possible ont respectera les règles suivantes :

- Une unité de test pour une unité de code (ou un lot de certification)
- Les fonctions de test sont nommé suivant la règle suivant :
[“T” ; Numéro Unique de test unitaire ; “_” ; Nom de la fonction]
par exemple “T021_IsPersonnalise” présente le 21eme test unitaire du projet il test test la fonction “IsPersonnalisé”.

Serveur LoGeAs

Numéro du dernier test implémenté dans cette série : 57

Fonctions de base liées à la gestion comptable

Numero	Unité implémentation Unité concernée(s)	Explication	Information complémentaire
T001..T003	TestUnit_Scripting Unit_Scripting GestionPlanServer	Test de la fonction “ CreerCompte ” : création d'un compte sur le plan comptable officiel	Voir Function CreerCompte(Compte:String):String
T004..T010	TestUnit_Scripting Unit_Scripting GestionPlanServer	Test de la fonction “ EffaceCompte ” : effacement d'un compte sur le plan comptable officiel	Voir Function EffaceCompte(Compte :String):String
T011..T014	TestUnit_Scripting Unit_Scripting GestionPlanServer	Test de la fonction “ CreerComptePersonnalise ” : création d'un compte personnalisé sur le plan comptable officiel	Voir Function CreerComptePersonnalise(Compte,Intitule :String):String
T015..T020	TestUnit_Scripting Unit_Scripting GestionPlanServer	Test de la fonction “ DeplaceEcriture ” : déplacement de saisie (et écriture) d'un compte du plan comptable officiel sur un autre	Voir Procedure DeplaceEcriture (CompteSource, CompteDestination : String)

Numero	Unité implémentation Unité concernée(s)	Explication	Information complémentaire
T021..T024	TestUnit_Scripting Unit_Scripting GestionPlanServer	Test de la fonction " IsPersonnalise " : fonction qui indique si un compte du plan comptable officiel à été personnalisé par l'utilisateur	Voir L'unité "Unit-Scripting"
T025..T031 T046	TestUnit_Scripting Unit_Scripting GestionPlanServer	Test de la fonction " DeplaceCompte " : déplacement d'un compte vers un autre (et aussi bien sur des écritures)	Voir procedure DeplaceCompte(Compte,CompteSubstitution:string)

Fonctions de base liées à la génération comptable

Numero	Unité implémentation Unité concernée(s)	Explication	Information complémentaire
T021	TestUnit_Scripting Unit_Scripting GestionPlanServer	Test de la fonction " GenerationEcriture " : transformation d'une saisie comptable en écritures	<u>Configuration</u> : * Une suite de saisie recette et dépense sur divers comptes <u>Attente en sortie</u> : * La somme des montant crédit = Somme des montant débit * Il existe deux fois plus d'écriture que de saisie
T035	TestUnit_Scripting GestionPlanServer	Test de la fonction " GenerationEcriture " : transformation d'une saisie comptable en écritures	<u>Configuration</u> : * Une suite de saisie sur une multiligne sur divers comptes <u>Attente en sortie</u> : * La somme des montant crédit = Somme des montant débit * Il existe autant d'écriture que de saisie
T033	TestUnit_Scripting GestionPlanServer	Test de la fonction " Generation " : transformation d'une saisie comptable en écritures et remonte les totaux sur les plans comptables	<u>Configuration</u> : * Une suite de saisie sur simple et multiligne sur divers comptes <u>en sortie</u> : Cf note (1) au dessous du tableau
T034	TestUnit_Scripting GestionPlanServer	Test des fonctions " Generation " : transformation de saisie comptable en écritures et remonte les totaux sur les plans comptables	<u>Configuration</u> : * Une base de test comportant de la comptabilité <u>Attente en sortie</u> : Cf note (1) au dessous du tableau

(1) Pour tester qu'une génération est correcte on réalise les tests suivants :

- Test réalisé actuellement uniquement sur les bases de type 13 (EPUDF)
 - la variable, du plan interne, "Passif" du bilan est non nul (pour tout usage de ses variables on utilise le solde courant)
 - les variables, du plan interne, "Passif" et "Actif" sont égale
 - la différence du produit moins les charges est égal au résultat avec les définitions suivantes :
 - "Produit" = somme des variables "CR_ProduitT1", "CR_ProduitT3", "CR_ProduitT5" et "CR_ProduitT7" du plan interne
 - "Charges" = somme des variables "CR_ChargeT2", "CR_ChargeT4", "CR_ChargeT6", "CR_ChargeT8", "CR_T9", "CR_T10", "CR_T11" et "CR_T12" du plan interne
 - "Résultat" = a la variables "CR_Resultat" du plan interne
 - le solde courant du compte "7" - Solde courant du compte "6" du plan officiel est égal au solde courant de "CR_Resultat" du plan interne
- Tous les test de génération passe Cf [testgeneration](#)

Fonctions comptable de base aux travers de l'exécution d'un script de migration comptable

Numero	Unité implémentation Unité concernée(s)	Explication	Information complémentaire
T036 T037 T045	TestUnit_Script GestionPlanServer	Test de l'application d'un script "complexe" de la migration de plan comptable 2021 script2021epudf	<u>Configuration:</u> * Une base de test comportant de la comptabilité <u>Attente en sortie:</u> Cf note (1) au dessous du tableau

Fonctions liées à la signature (enregistrements et fichiers)

Fonctions de base liées à la dll Openssl

Numero	Unité implémentation Unité concernée(s)	Explication	Information complémentaire
T047	TestUnit_OpenSSL Util_OpenSSL(CreerCertificat)	Création de certificat X509 avec Clef RSA de 2048 et encrypté en AES-129-CBC	<u>Configuration:</u> <u>Attente:</u> Les fichiers des certificats existent
T052	TestUnit_OpenSSL Util_OpenSSL(GetAleatChaine)	Création de chaine aléatoires : On génère 1000 chaines de longueur 20 et on vérifie si elle sont différente	<u>Configuration:</u> 1000 chaine aléatoire <u>Attente:</u> Elles sont différentes
T048	TestUnit_OpenSSL Util_OpenSSL(GetTexteSignature & CheckTexteSignature)	Signature d'une chaine de caractères vérification de la signature	<u>Configuration:</u> * Génération de certificat Signature de la chaine <u>Attente:</u> la signature existe, elle est vérifiable

Numero	Unité implémentation Unité concernée(s)	Explication	Information complémentaire
T049	<i>TestUnit_OpenSSL</i> Util_OpenSSL(GetTexteSignature & CheckTexteSignature)	Signature d'une chaine de caractères, puis modification de la chaine	<u>Configuration:</u> * Génération de certificat Signature de la chaine <u>Attente:</u> la signature n'est pas bonne
T050	<i>TestUnit_OpenSSL</i> Util_OpenSSL(GetTexteSignature & CheckTexteSignature)	Signature d'une chaine de caractères, puis modification de la signature	<u>Configuration:</u> * Génération de certificat Signature de la chaine <u>Attente:</u> la signature n'est pas bonne
T051	<i>TestUnit_OpenSSL</i> Util_OpenSSL(GetTexteSignature & CheckTexteSignature)	Signature d'une chaine de caractères, puis vérification de la signature avec un autre certificat	<u>Configuration:</u> * Génération de 2 certificat Signature de la chaine <u>Attente:</u> la signature n'est pas bonne
T056	<i>TestUnit_NF</i> GenerationCompta	Prends une base, vérifie que son intégrité (signature) est correcte réalise la clôture comptable	Les signatures doivent toujours être correctes
T057	<i>TestUnit_NF</i> GenerationCompta	Prends une base ou les signatures sont cassé et les reconstruit	Les signatures doivent toujours être correctes

Fonctions de plus haut niveau liés à la NF

Numero	Unité implémentation Unité concernée(s)	Explication	Information complémentaire
T040	<i>TestUnit_NF</i> RecordSignedRecord (VerifieSignature) Tables comptables	Réalise une série de saisie, les génère et donc les signe. On vérifie alors que les signatures sont conformes, puis on modifie un valeur de la table et on retest	<u>Configuration:</u> * Une base de test <u>Attente en sortie:</u> Le test des écritures détecte une erreur
T042	<i>TestUnit_NF</i> RecordSignedRecord (VerifieSignature) Tables comptables	Réalise une série de saisie, les génère et donc les signe. On vérifie alors que les signatures sont conformes, puis on modifie lors une signature et on retest	<u>Configuration:</u> * Une base de test <u>Attente en sortie:</u> Les tests de cohérence passe sur PistAudit, Signature, Saisie Et après modification le test de la piste détecte une erreur

Numero	Unité implémentation Unité concernée(s)	Explication	Information complémentaire
T052	TestUnit_NF RecordSignedRecord (VerifieSignature) Tables JET	Vérifie la cohérence des signatures réalisées en version 2 (9.5 à v10) (	Configuration: * Une base de test antérieur à la version 10 Attente en sortie: Les tests de cohérence passe sur PistAudit
T043	TestUnit_NF RecordSignedRecord (VerifieSignature) Tables Personne	Vérifie la cohérence des signatures réalisées en version 2 (9.5 à v10) (	Configuration: * Une base de test antérieur à la version 10 Attente en sortie: Les tests de cohérence passe sur Personne
T044	TestUnit_NF RecordSignedRecord (VerifieSignature) Tables Famille	Vérifie la cohérence des signatures réalisées en version 2 (9.5 à v10) (	Configuration: * Une base de test antérieur à la version 10 Attente en sortie: Les tests de cohérence passe sur Famille
T053	TestUnit_NF RecordSignedRecord (VerifieSignature) Tables Famille	Vérifie la cohérence des signatures d'une base réel (	Configuration: * Une base de test antérieur à la version 10 Attente en sortie: Les tests de cohérence passe sur Famille

Fonctions liées à la sauvegarde/restauration des bases

Numero	Unité implémentation Unité concernée(s)	Explication	Information complémentaire
T038	TestUnit_NF BackupSynopse	Réalise une sauvegarde	Configuration: * Une base de test Attente en sortie: Elle existe Elle est de taille non nul
T039	TestUnit_NF BackupSynopse	Réalise une sauvegarde et la signe	Configuration: * Une base de test Attente en sortie: Elle existe Elle est de taille non nul
T047	TestUnit_NF BackupSynopse Util_OpenSSL	Reprends la sauvegarde T039, vérifie que la signature est OK	Configuration: * Une base de test Attente en sortie: Elle existe Elle est de taille non nul La signature est valide
T041	TestUnit_NF BackupSynopse Util_OpenSSL	Réalise une sauvegarde et la signe, la modifie et vérifie que la signature est fausse	Configuration: * Une base de test Attente en sortie: la signature est corrompu

Numero	Unité implémentation Unité concernée(s)	Explication	Information complémentaire
T048	TestUnit_NF BackupSynopse Util_OpenSSL	Reprends une sauvegarde V9 et vérifie que la signature est Bonne	Configuration: * Une base de test Attente en sortie: la signature est bonne
T054	TestUnit_NF BackupSynopse Util_OpenSSL	Reprends la sauvegarde T039 et la restaure	Configuration: * Une base de test Attente en sortie: Test l' "égalité" de la base initial et final(1) Vérifie qu'un enregistrement de restauration a été mis dans la nouvelle base
T055	TestUnit_NF BackupSynopse Util_OpenSSL	Reprends une sauvegarde V9 et la restaure	Configuration: * Une base de test Attente en sortie:\\Vérifie que la base existe et est non vide

(1) Pour vérifier l'égalité de deux bases on test que pour les tables "Famille", "Personne", "Saisie", "Ecriture" que l'on a le même nombre d'enregistrement et que le dernier enregistrement est le même (a faire évoluer)

From:

<https://wiki-logeas.fr/certif/> - **dokuwiki-certif**

Permanent link:

<https://wiki-logeas.fr/certif/doku.php?id=certif:test:unitaire&rev=1790859809>

Last update: **2026/10/01 15:03**

