



CAHIER DE RECETTE DU NOYAU AMDA-NG

Rédigé par : <i>Architecte : Freddy CASIMIR</i>	Diffusé à : CNES/ IRAP
Approuvé par : Chef de projet CNES : DUFOURG Nicolas	

LISTE DES MODIFICATIONS DU DOCUMENT

Vers.	Date	Paragraphe	Description de la modification
01.00	15/01/2013		Création du document
01.01	05/02/2013		Livraison V1.1.0

SOMMAIRE

1	INTRODUCTION	5
1.1	But du document.....	5
1.2	Domaine d'application.....	5
2	RAPPEL DES FONCTIONNALITES DU PRODUIT	6
3	ARCHITECTURE DU PRODUIT	15
4	COUVERTURE DE LA RECETTE	16
4.1.1	LES LOGICIELS.....	16
4.1.2	LES DOCUMENTS.....	16
4.1.3	LES MATERIELS.....	17
5	LES ESSAIS DE RECETTE	18
5.1	INSTALLATION – PARAMETRAGE.....	18
5.2	LES TESTS FONCTIONNELS.....	18
5.3	LES CONTRAINTES.....	18
5.3.1	CONTRAINTES D'ENVIRONNEMENT.....	18
5.3.2	PERFORMANCES.....	18
6	LOGIQUE D'ENCHAINEMENT DES ESSAIS	19
6.1	Dossier de référence.....	19
6.2	Liste des tests.....	19
7	ORGANISATION DE LA RECETTE	20
7.1	ENVIRONNEMENT.....	20
7.1.1	CONFIGURATION ET SITE.....	20
7.1.2	MOYENS D'ESSAIS.....	21
7.2	PLANNING.....	21
7.3	INTERVENANTS.....	21
7.4	RESULTATS DE LA RECETTE.....	21
7.4.1	Conditions d'acceptation.....	21
7.4.2	Gestion des anomalies détectées en recette.....	22

8	ANNEXE 1 - MATRICE DE COUVERTURE DES USERS STORIES.....	23
9	ANNEXE 2 - FAITS TECHNIQUES INCLUS.....	24
10	ANNEXE 3 - FAITS TECHNIQUES NON COUVERTS	25
11	DOCUMENTS APPLICABLES ET DE REFERENCE (A/R).....	26
12	GLOSSAIRE ET ABREVIATIONS.....	27
12.1	Glossaire.....	27
12.2	Abréviations	27

1 INTRODUCTION

1.1 BUT DU DOCUMENT

Le présent document constitue le cahier de recette du noyau AMDA-NG.

Ce document décrit les modalités de déroulement de la phase de recette. Il précise notamment les prérequis, le périmètre, les moyens mis en œuvre, les conditions de passage des tests, les critères d'acceptation et le planning.

1.2 DOMAINE D'APPLICATION

Le présent document concerne la recette site.

Les descriptions des tests effectués ainsi que les résultats de ces tests sont décrits dans l'outil web de test d'acceptation FitNesse qui sera exécuté durant la recette par l'IRAP/CNES. Une impression pdf de la page de résultat fera office de journal d'essai.

Le périmètre applicatif concerné est le suivant :

- ☞ Logiciel AMDA-NG Module paramètres (Cf DA1)

Note : La prestation ne couvre pas l'intégration de ce module à l'IHM AMDA, par conséquent les tests se font en ligne de commande, par le lancement de trois exécutables autonomes instanciant ce module.

2 RAPPEL DES FONCTIONNALITES DU PRODUIT

Release n°- Sprint n°	ID	Name	Description	Type	Feature
R1-S1	31	Connexion à DD_Server	Connexion à DD_server à partir de la librairie libDD.a fournie par l'IRAP.	Technique	Couverture fonctionnelle de l'existant
R1-S2	36	Création dynamique de paramètre	<p>As Utilisateur pilote</p> <p>I can donner en entrée argument ligne de commande un identificateur de paramètre et un intervalle de temps et option format.</p> <p>Auparavant, j'ai précisé dans un fichier de configuration de propriétés clé=valeur, l'emplacement du répertoires des fichiers d'information des paramètres (format xml) à l'endroit où on lance l'exécutable.</p> <p>To donner en entrée argument ligne de commande un identificateur de paramètre et un intervalle de temps et option format.</p> <p>Auparavant, j'ai précisé dans un fichier de configuration de propriétés clé=valeur, l'emplacement du répertoires des fichiers d'information des paramètres (format xml) à l'endroit où on lance l'exécutable.</p>	Fonctionnel	Couverture fonctionnelle de l'existant
R1-S3	54	Simple parameter with processing, one getParam	Given parameter ID and StartTime and Time interval calculate and output parameter values using ParamGet and ParamProcess definitions from parameter XML descriptor. Type and size of parameter is to be defined by ParamProcess class. ParamProcess makes calculations with the use of the predefined functions from dynamically linked library.	Fonctionnel	Couverture fonctionnelle de l'existant
R1-S3	49	Time Validation	Test new DD_Client function DD_SetTimeInfo(); Validate that time set by DD_SetTimeInfo() is in the requested time interval, otherwise return OUTOFTIME or NODATAATTIME error	Fonctionnel	Couverture fonctionnelle de l'existant

Release n°- Sprint n°	ID	Name	Description	Type	Feature
R1-S3	4	Paramètre integer	<p>As Utilisateur pilote</p> <p>I can lancer le kernel AMDA en ligne de commande avec une requête sur un paramètre scalaire ou paramere 1D tableau integer en connaissant son identifiant et sur un intervalle de temps</p> <p>To lancer le kernel AMDA en ligne de commande avec une requête sur un paramètre scalaire ou paramere 1D tableau integer en connaissant son identifiant et sur un intervalle de temps</p>	Fonctionnel	Couverture fonctionnelle de l'existant
R1-S3	1 2	Tableau 1D floatparameter simple	<p>En tant que Utilisateur pilote, je peux lancer le kernel AMDA en ligne de commande avec une requête sur un simple paramètre de type 1D tableau en connaissant son identifiant et sur un intervalle de temps afin de obtenir un fichier ASCII avec les différentes valeurs du paramètre sur l'intervalle de temps demandé et information supplémentaire sur le parameter (getInformfonction de DDServer).</p>	Fonctionnel	Couverture fonctionnelle de l'existant
R1-S3	3 5	Transformation temps format 'YYYYDDHMMSSm ls' en double en format ISO	<p>As Utilisateur pilote</p> <p>I can donner en entrée en argument un identificateur d'instrument virtuel, un nom de paramètre, un temps debut et un intervalle de temps, et l'option format de sortie double ou ISO par défaut format fourni par DDServer.</p> <p>To donner en entrée en argument un identificateur d'instrument virtuel, un nom de paramètre, un temps debut et un intervalle de temps, et l'option format de sortie double ou ISO par défaut format fourni par DDServer.</p>	Fonctionnel	Couverture fonctionnelle de l'existant
R1-S4	1 9	gestion des erreurs	il faut prévoir : LOG, messages des erreurs, mode 'debug', ...	Fonctionnel	Maintenabilité
R1-S4	4 6	traitement de grande intervalle du temps	Construct and output values of parameter for large time interval when several calls to DDServer are needed. Size of 'time block' is to be defined in the system.	Fonctionnel	Couverture fonctionnelle de l'existant
R1-S4	3 7	Parametercomposee 'systeme' sans echantillonnage	<p>Construct and output values of composed parameter based on 2 simple scalar parameters without resampling (from one dataset) using internal standard function</p> <p>Functions list and one test per fonction définir tag processing du fichier XML à trouver dans le dossier de conception.</p>	Fonctionnel	Couverture fonctionnelle de l'existant

Release n°- Sprint n°	ID	Name	Description	Type	Feature
R1-S4	20	message des erreurs utilisateur	Acceptance le dossier de conception doit contenir les codes erreurs et leurs significations	Fonctionnel	Maintenabilité
R2-S1	17	Resampling d'un paramètre	As Utilisateur pilote I can faire une requête sur le paramètre avec une entréesupplémentaire de temps d'échantillonnage (argument sampling-mode=classic) To faire une requête sur le paramètre avec une entréesupplémentaire de temps d'échantillonnage (argument sampling-mode=classic)	Fonctionnel	Couverture fonctionnelle de l'existant
R2-S1	29	Composante de paramètre simple	En tant que Utilisateur pilote, je peux lancer le kernel AMDA en ligne de commande avec une requête sur un simple paramètre de type tableau ou vecteur en connaissant son identifiant, numéro de composante (indice de tableau 1D) et sur un intervalle de temps afin de obtenir un fichier ASCII avec les différentes valeurs du paramètre composante sur l'intervalle de temps demandé.	Fonctionnel	Couverture fonctionnelle de l'existant
R2-S1	47	Multi-Paramrequest	Output values of two parameters for the same time interval (using one request) into different ASCII files. Faire fichier requête.	Fonctionnel	Couverture fonctionnelle de l'existant
R2-S1	16	Resamplingparameter avec 'gaps'	Définir l'intervalle de temps à partir duquel l'absence de données est considéré comme 'gap' .Faire échantillonnage de parameter. Input args: Sampling : new sampling time. note that original param sampling is to be got by DD_getInform('minSampling') GapsThreshold: min gap value (in original sampling time) Put NaN for large gaps	Fonctionnel	Couverture fonctionnelle de l'existant
R2-S1	61	Composed parameter with time shifting processing function	Construct and output values of composed parameter obtained by application of time shift function to simple parameters	Fonctionnel	Couverture fonctionnelle de l'existant
R2-S1	59	Multi-request: different types of output for the same parameter	Run multi-request with two different types of output for the same parameter: 1d table and component of 1d table. Generatetwo ascii files.	Fonctionnel	Couverture fonctionnelle de l'existant

Release n°- Sprint n°	ID	Name	Description	Type	Feature
R2-S2	40	Two composed parameters: one simple parameter	construct and output as ASCII file[s] values of 2 composed parameters which are constructed from one simple parameter. Implement mechanism of validation if data for parameter creation already exists in the system to avoid unneeded calls to DDServer	Fonctionnel	Couverture fonctionnelle de l'existant
R2-S2	42	Paramètre composé sur scalaire avec fonctions	En tant que Utilisateur pilote, je peux créer un paramètre composé défini par une expression utilisant des paramètres scalaires simples et des fonctions mathématiques (ie. sin, cos, tan, acos, asin, atan, sinh, cosh, tanh, abs, sqrt, exp, log10, log), afin de récupérer un fichier ASCII des valeurs correspondantes à ce paramètre.	Fonctionnel	Couverture fonctionnelle de l'existant
R2-S2	7	Paramètre composé utilisateur sur scalaire, avec opération simple et expression simple	construct and output values of composed parameter based on user defined expression with one arithmetic operator and parameters simple. Available operators for this US are: +, -, *, /, ^ The expression contain only one operator (for example: paramA + paramB) Validation formule	Fonctionnel	Couverture fonctionnelle de l'existant
R2-S2	41	Paramètre composé utilisateur sur vecteur	Construct and output values of composed parameter based on user defined expression with arithmetic operators and parameters simple vectoriels	Fonctionnel	Fonctionnalités nouvelles
R2-S2	63	Paramètre composé utilisateur sur composante de 1d tableau ou vecteur	construct and output values of composed parameter based on user defined expression with arithmetic operators and parameters simple and with the use of parameter component Validation formule	Fonctionnel	Couverture fonctionnelle de l'existant

Release n°- Sprint n°	ID	Name	Description	Type	Feature
R2-S2	68	Paramètre composé utilisateur sur scalaire, avec opérations simples et expression complexe	<p>construct and output values of composed parameter based on user defined expression and parameters simple.</p> <p>Available operators for this US are: +, -, *, /, ^</p> <p>Very close to US7 but for this one, the expression can be complex and use several operators.</p> <p>Validation formule</p>	Fonctionnel	Couverture fonctionnelle de l'existant
R2-S2	21	<u>Compilation mode release</u>	En tant qu'Installateur, je peux deployer le noyau AMDA compilé en mode release, afin d'améliorer les temps d'exécution de l'application.	Fonctionnel	Performances
R2-S2	38	Parameter composee 'systeme' avec reechantillage	<p>Construct and output values of composed parameter based on 2 simple parameters with resampling (from different datasets) using internal standard function</p> <p>Exemplesiparamcomposéunebalise sampling optionnelle à définir dans le fichier XML.</p>	Fonctionnel	Couverture fonctionnelle de l'existant
R2-S2	23	<u>Documentation "Ajout de fonction mathématique"</u>	En tant que Développeur de maintenance, je peux me référer à une documentation concernant l'ajout d'une fonction mathématique, afin de la rendre disponible aux utilisateurs du noyau AMDA.	Fonctionnel	Evolutivité & Extensibilité
R2-S3	75	DD_GetMultiParam()	New DD_Client function: get multi parameter data in one request to DD Server. Update paramGet class	Fonctionnel	Fonctionnalités nouvelles
R2-S3	60	complexparameter calibration processing	<p>Calculate and Output parameter values and calibration information (see US12)</p> <p>Needed calibration information is constructed with the use of internal library function. Function is time UNDEPENDENT : only DD_getInform() is used to obtain calibration data</p> <p>energy_bounds(E,Emin,Emax) : e_mgs_er_info[0,*] = E[*]-Emin[*]; e_mgs_er_info[1,*] = E[*] + Emax[*]; return e_mgs_er_info;</p>	Fonctionnel	Couverture fonctionnelle de l'existant

Release n°- Sprint n°	ID	Name	Description	Type	Feature
R2-S3	22	ajout de nouveau type de parameter	pouvoir rajouter un nouveau type de paramètre	Fonctionnel	Evolutivité & Extensibilité
R2-S3	64	Paramètre composé utilisateur avec opérateurs logiques	construct and output values of composed parameter based on user defined expression with arithmetic and logic operators and parameters simple . Output parameter values are: 0 (false), 1 (true), 2 (no data) Relational operators: < and > Logical operators: & and Validation formule	Fonctionnel	Couverture fonctionnelle de l'existant
R2-S3	66	Composed parameter with boxcar averaging processing function	En tant que Développeur de maintenance, je peux me référer à une documentation concernant l'implémentation d'un nouveau type de "getParam", afin de pouvoir récupérer les données d'un paramètre depuis un autre service que "DD server".	Fonctionnel	Couverture fonctionnelle de l'existant
R2-S3	24	Documentation "Ajout d'un nouveau type de getParam"	En tant que Développeur de maintenance, je peux me référer à une documentation concernant l'implémentation d'un nouveau type de "getParam", afin de pouvoir récupérer les données d'un paramètre depuis un autre service que "DD server".	Fonctionnel	Evolutivité & Extensibilité
R2-S3	45	Resampling d'un paramètre en considérant la "filling value"	As Utilisateur pilote I can faire une requête sur le parametre avec un argument supplémentaire de temps d'échantillonnage. (Filling value est defini par DD_GetInform() appele a DD_Server) To faire une requête sur le parametre avec un argument supplémentaire de temps d'échantillonnage. (Filling value est defini par DD_GetInform() appele a DD_Server)	Fonctionnel	Couverture fonctionnelle de l'existant
R2-S3	65	Composed parameter with running average	Construct and output values of composed parameter obtained by application of running average function to simple parameters	Fonctionnel	Couverture fonctionnelle de l'existant
R2-S3	62	time dependentparameter calibration	Calibration information is time dependent.	Fonctionnel	Couverture fonctionnelle de l'existant

Release n°- Sprint n°	ID	Name	Description	Type	Feature
R2-S3	25	Documentation "Ajout d'un nouveau type d'output"	En tant que Développeur de maintenance, je peux me référer à une documentation concernant l'implémentation d'un nouveau type d'output, afin de fournir aux utilisateurs une nouvelle sortie pour les paramètres.	Fonctionnel	Evolutivité & Extensibilité
R3-S1	70	User defined parameter: expression with time dependent function	Complex parameter : expression with time dependent function. (magnitude(shift(imf,T)) - magnitude(imf)) / magnitude(imf)	Fonctionnel	Couverture fonctionnelle de l'existant
R3-S1	74	Optimisation of parameter Calculation 2	Create *.so for parameter at the moment when parameter is defined by user. Implement new DD Server call getDataType()	Fonctionnel	Fonctionnalités nouvelles
R3-S1	79	Function 'derivative'	Numerical differentiation. 3 point Lagrangian interpolation (points {x0,y0; x1,y1; x2,y2}): $df/dx = y0*(2x-x1-x2)/(x01*x02) + y1*(2x-x0-x2)/(x10*x12)+y2*(2x-x0-x1)/(x20*x21),$ where x01= x0-x1; x02=x0-x2, x12=x1-x2... {x0,y0; x1,y1; x2,y2} => {x[i-1],y[i-1]; x[i],y[i]; x[i+1],y[i+1]} calculate value (df/dx)[i] using three points i-1, i, i+1 First point (df/dx)[0] is calculated using points 0, 1, 2 Last point (df/dx)[n-1] is calculated using points n-3,	Fonctionnel	Couverture fonctionnelle de l'existant

Release n°- Sprint n°	ID	Name	Description	Type	Feature
			n-2, n-1		
R3-S1	81	Traitement des paramètres non trouvés	<p>A l'exécution d'une requête, il se peut qu'un paramètre réclamé ne soit pas trouvé.</p> <p>L'appel de la nouvelle fonction 'getMultiData' de 'DD server' ne renvoie plus le code erreur NOVAR lorsqu'un des paramètres demandés n'existe pas. Dorénavant, la structure de data renvoyée par 'DD server' correspondant à un paramètre qui n'existe pas est remplie de la manière suivante:</p> <pre>.type == NOVAR .DimNumber == 0 .Dimensions == NULL .VarNumber == 0 .Variables == NULL</pre> <p>Cette tâche consiste à faire tout de même le output des paramètres existants.</p> <p>Dans le cas du output ASCII, la colonne d'un paramètre qui n'existe pas affichera pour chaque lignes de données 'NOVAR'.</p> <p>De plus, un log et un code retour seront renvoyés</p>	Fonctionnel	Couverture fonctionnelle de l'existant

Release n°- Sprint n°	ID	Name	Description	Type	Feature
R3-S1	76	VI-dependentgetParam()	Minimize calls to DD Server by using one getMultiData() call for ALL parameters of the same VI (virtual instrument)	Fonctionnel	Fonctionnalités nouvelles
R3-S1	58	Complexparameterprocessing	Construct and output values of parameter using several basic parameters (<baseParam>), standard library function and time dependent calibration. Finalizeparameter XML descriptor els_calculate(ELS,TimeCompres,EnergyCompres,vex_els_energy,Angle) for (iEn = 0; iEn< 128; iEn++) // 128 EnergySteps (vex_els_flux[t])[iEn] = (float)(ELS[t])[iEn + Angle *128]/(float)(T_Compres[t])/float)(E_Compres[t])/(vex_els_energy[t])[iEn]; Angle = 10	Fonctionnel	Couverture fonctionnelle de l'existant
R3-S1	80	Optimisation of parametercalculation 1	Implement mechanism to check if *.so for parameter exists already and compilation is not needed.	Fonctionnel	Fonctionnalités nouvelles

3 ARCHITECTURE DU PRODUIT

Cf DA2 et DA3

4 COUVERTURE DE LA RECETTE

Les tests prévus pour l'acceptation reprennent ceux passés lors de la validation, augmentés d'un test dont le but est l'installation des logiciels sur le serveur fourni (Cf DA5).

Ces tests permettent de valider le logiciel par rapport aux « Users Stories » (équivalent des exigences fonctionnelles).

La stratégie de validation est entièrement décrite dans le Plan de Test (DA4) et la stratégie de test pour la recette est de générer, installer puis de rejouer tous les tests (fonctionnels) automatisés sur la plateforme cible.

Tous les tests qui seront passés ont été validés lors des diverses revue de Sprint.

4.1 - FOURNITURES A RECETTER

4.1.1 LES LOGICIELS

Voir le manuel d'installation DA5 qui présente la liste des logiciels standards §3 à installer et logiciels spécifiques §4 à installer et « recetter ».

La livraison d'une nouvelle version de AMDA-Kernel se fait sous forme d'un tgz (archive tar compressé avec gzip) avec pour nomenclature :

<date AAAA-MM-JJ>-AMDA_Kernel_V<version>.tgz

Exemple: 2013-01-15-AMDA_Kernel_V1.0.0.tgz

Cette archive contient les sources complets de l'application, les tests automatisés ainsi que le client et le serveur de test.

Les données de test sont de la responsabilité de l'IRAP/CNES.

Les COTS sont livrés dans leurs conteneurs d'origine.

4.1.2 LES DOCUMENTS

Référence	Nom du document	Version	Date de la version	Emetteur
CDPP-AR-32500-382-SI	Dossier d'architecture	V02.01	29/11/12	AKKA IS
CDPP-CD-32500-436-SI	Dossier de conception	V01.0		AKKA IS
CDPP-IF-32500-438-SI	Dossier de contrôle des interfaces du noyau AMDA-NG	V01.0		AKKA IS

CDPP-MI-32500-440-SI	Manuel d'installation du noyau AMDA-NG	V01.0	AKKA IS
CDPP-PE-322500-439-SI	Noyau AMDA - Plan de test	V01.01	AKKA IS

4.1.3 LES MATERIELS

Sans objet

4.2 - CONTEXTE DE LA RECETTE

La recette portera sur toutes les « Users stories » (fonctionnalités) constituant la version du produit livré. Ces « Users stories » ont été acceptées lors des revues de « Sprint ». Elles sont listées accompagnées de leurs tests d'acceptation dans l'outil web icescrum

(<http://yahscrum.akka.eu:8080/icescrum/p/AMDAKERNEL/#project>) et seront récapitulées dans le journal des essais.

Liste des « Users Stories » non couvertes :

Toutes les « Users Stories » sont couvertes.

5 LES ESSAIS DE RECETTE

5.1 INSTALLATION – PARAMETRAGE

Cf DA5 §4 Installation et §5 Post-Installation

Les tests devront être passés sur le produit compilé en mode « Release » puis en mode « Debug ».

5.2 LES TESTS FONCTIONNELS

Comme mentionné dans le DA4 (Plan de Test), tous les tests fonctionnels ont été automatisés et sont livrés avec les sources de l'application.

5.3 LES CONTRAINTES

5.3.1 CONTRAINTES D'ENVIRONNEMENT

Une instance de DDServer doit être démarrée avec les jeux de données correspondants aux tests d'acceptation des Users Stories disponibles (responsabilité de l'IRAP).

Le serveur doit pouvoir communiquer/joindre DDServer.

Les pré-requis de DA5 doivent être respectés.

5.3.2 PERFORMANCES

Sans objet

6 LOGIQUE D'ENCHAINEMENT DES ESSAIS

6.1 DOSSIER DE REFERENCE

Les tests d'acceptation sont menés sur les versions figées du logiciel.

La version logicielle figée actuelle est : 1.1.0.

6.2 LISTE DES TESTS

Pendant le déroulement de la procédure d'installation (cf DA5), des impressions pdf des résultats FitNesse obtenus en mode Debug et en mode Release constitueront le journal des essais et identifieront clairement la liste des tests passés en recette.

Cette liste de tests sera en issue de IceScrum et l'IRAP/CNES pourra, dans la limite du planning défini, effectuer des tests supplémentaires. Dans ce cas, la fiche de test devra être rédigée et complétée par l'IRAP/CNES sous IceScrum et FitNesse, puis consignée dans le journal des essais.

7 ORGANISATION DE LA RECETTE

7.1 ENVIRONNEMENT

7.1.1 CONFIGURATION ET SITE

7.1.1.1 Configuration matérielle

Les machines utilisées pour les tests de recette sont les suivantes :

Code	Type de machine	Système d'exploitation	Description
cdpp1.cesr.fr	Dell PowerEdge R620 Bi-proc intel Xeon E5- 2640 2.50GHz – 15M Cache – Turbo 6C – 48Go de mémoire	Cent OS 6.3 64bits	Machine située à l'IRAP. Le noyau n'est pas hébergé sur la machine cdpp1.cesr.fr, mais sur une machine virtuelle Virtualbox (qui elle tourne sur cdpp1), dont voici les caractéristiques : <ul style="list-style-type: none"> • amdakernel.cesr.fr • Système d'exploitation Cent OS 6.3 64 bits • 1 processeur utilisé • 1 Go de mémoire allouée Cette machine virtuelle contient : <ul style="list-style-type: none"> • Le noyau • Les outils de développement pour le noyau La plateforme de test fitnessse
manunja.cesr.fr			Machine ou s'exécute DDServer

7.1.1.2 Outils logiciels

Ce chapitre présente les outils logiciels nécessaires sur l'environnement de recette en dehors des logiciels fournis et objet de cette recette. Il s'agit par exemple des outils de génération ou des logiciels pré-requis.

Outil	Version	Plate-forme
-------	---------	-------------

Firefox	Sans grande importance Version 17.0.1 a été testé	Machine accédant au FitNesse de la machine d'intégration
---------	--	--

7.1.2 MOYENS D'ESSAIS

Sans Objet

7.2 PLANNING

La recette de la version 1.1.0 du logiciel se déroule sur **5 jours ouvrés**.

L'ensemble des tests prévus est déroulé sur cette période. Ce délai ne peut être remis en cause si l'ensemble des tests prévus n'a pu être passé complètement. Toutefois, des anomalies bloquantes pour les tests peuvent ajourner la recette et nécessiter de poursuivre celle-ci une fois les corrections apportées.

Les dates de recette et de CRE sont les suivantes :

- ☞ Recette : du 06/02/2013 au 13/02/2013 à AKKA
- ☞ CRE : le 14/02/2013 à 14h à l'IRAP

7.3 INTERVENANTS

L'IRAP procédera à la recette.

7.4 RESULTATS DE LA RECETTE

7.4.1 Conditions d'acceptation

Suite à la livraison complète des éléments de recette, l'IRAP/CNES dispose d'un délai fixe pour effectuer la recette. Ce délai est défini pour chaque version recettée en fonction du périmètre fonctionnel soumis à acceptation.

En fin de période de recette, l'analyse des anomalies résiduelles permet de donner les éléments de prononciation ou non de la recette :

- S'il subsiste des Anomalies Bloquantes, la prononciation de la recette peut être refusée ;
- S'il subsiste des Anomalies non-bloquantes, la recette est prononcée avec réserves et conditionnée par un plan d'actions de corrections, annexé au « Procès Verbal de Recette » ;
- Sinon, la recette est prononcée sans réserve

Une fois les anomalies corrigées et après recette par l'IRAP/CNES, le responsable de la réalisation procède à la livraison définitive (bon de livraison) dans un espace dédié pour mise en exploitation.

Les Anomalies Externes ne peuvent pas entraîner de refus de recette.

Anomalie externe : anomalie due à une cause étrangère à la fourniture de AKKA. Dans cette définition d'anomalie entrent par exemple :

- les erreurs consécutives aux éléments manquants autres que ceux dont AKKA a la charge,
- les anomalies relatives aux pannes matérielles,
- les anomalies liées à une mauvaise configuration (variables d'environnement, fichiers de paramétrage).

NB : Une mise en exploitation entraîne la prononciation de la recette.

7.4.2 Gestion des anomalies détectées en recette

En cas d'anomalie détectée lors de la recette, une « user story » de type « défaut » est créée sous IceScrum avec le test qui a remonté cette anomalie. Un plan d'action est alors proposé par l'équipe AKKA Technologies.

En cas d'anomalie bloquante, la recette peut être ajournée, le temps pour AKKA Technologies de corriger l'anomalie et livrer un patch correctif.

8 ANNEXE 1 - MATRICE DE COUVERTURE DES USERS STORIES

Fait sous IceScrum : à chaque User Stories correspond un ou plusieurs tests d'acceptation.

(Egalement visible dans le cahier des essais)

9 ANNEXE 2 - FAITS TECHNIQUES INCLUS

Sans objet

10 ANNEXE 3 - FAITS TECHNIQUES NON COUVERTS

Sans objet

11 DOCUMENTS APPLICABLES ET DE REFERENCE (A/R)

A/R	Référence	Titre
A1	CDPP-ST-32500-417-CNES	SPECIFICATION TECHNIQUE INDUSTRIALISATION DU NOYAU AMDA
A2	CDPP-AR-32500-382-SI	Dossier d'architecture du noyau d'AMAD-NG
A3	CDPP-IF-32500-438-SI	Dossier de contrôle des interfaces du noyau AMDA-NG
A4	CDPP-PE-322500-439-SI	Noyau AMDA - Plan de test
A5	CDPP-MI-32500-440-SI	Manuel d'installation du noyau AMDA-NG

12 GLOSSAIRE ET ABREVIATIONS

12.1 GLOSSAIRE

Terme	Définition

12.2 ABREVIATIONS

Abréviation	Nom détaillé