

CAHIER DE RECETTE

Version :	1.0
Date :	12/01/17
Rédigé par :	L'équipe SmartLogger
Relu par :	L'équipe SmartLogger

HISTORIQUE DE LA DOCUMENTATION

Version	Date	Modifications réalisées
0.1	11/12/16	Création
0.2	05/01/17	Modifications de la partie 3. Stratégie de tests
1.0	12/01/17	Correctifs et restructuration du document
1.1	26/01/17	Correction liées aux remarques professorales

1. Documents applicables et de référence

- Le document de Spécifications Techniques des Besoins : STB.pdf.
- Le document d'architecture du logiciel : DAL.pdf
- Le Glossaire des termes techniques et des acronymes : Glossaire.pdf

2. Environnement de test

La plate-forme cible du projet est un serveur Linux, ainsi dans l'optique d'effectuer des tests sur un système le plus proche possible de l'original, nous allons travailler sur des Systèmes Linux uniquement, sachant que les systèmes Microsoft ont été écartés par le client.

Site de réalisation des tests :

- Réalisation des tests sur nos machines personnelles.
- Potentiellement un serveur présent directement sur le site client.
- Utilisation d'un serveur identique à celui du client afin d'avoir un environnement de travail identique pour le produit, en production ou durant son développement.

Configurations matérielles utilisées :

- Système Linux (Fedora, Ubuntu)
- Serveur basé sur un système Linux.

Outils de test mis en œuvre :

- JUnit pour les tests unitaires de la partie Java du projet.

Outils de gestion des anomalies :

- Utilisation d'un logiciel de Bug Tracker tels que MantisBT ou encore Redmine.

Jeux de données utilisés durant les phases de tests :

- Données de la forme : *niveau de criticité – informations importantes*
 - *niveau de criticité* : Valeur numérique permettant d'affilier un niveau de criticité à un format de données précis.
 - *Informations importante* : Toutes les données traités par le système sont considérés comme importante, c'est pourquoi nous stockons l'ensemble des données sans distinctions.

3. Stratégie de tests

3.1. Solution de test

Les solutions de test que nous avons choisi de définir consiste à rédiger les tests en parallèle du développement de notre solution. C'est à dire qu'une partie de l'équipe se chargera de la rédaction et de l'automatisation complètes des tests pendant que la seconde partie de l'équipe se verra attribuer la tâche de développer le contenu de l'itération.

Cela permettra ensuite, via automatisation des tests de pouvoir rejouer après chaque développement d'itération les tests unitaires de chacun des modules afin d'accélérer le processus de test unitaire, permettant ensuite de passer d'avantage de temps sur la partie Test d'intégration du projet.

En cas d'anomalie durant le développement, l'utilisation d'un outil de bug tracking permettra de facilement indiquer le problème, comment le reproduire ainsi que d'autres informations complémentaires afin de permettre aux développeurs de la partie de corriger rapidement le problème.

3.2. Réalisation des tests par cas d'utilisation

[UC-001] Fonctionnement principal				
Objectif	Vérifier le fonctionnement principal de l'application			
Éléments à tester	Réception des données externes Conversion des données dans le bon format Vérification de la validité des données Stockage des données dans la base de données Analyse du module analytique fonctionnelle Adaptation du comportement en fonction des résultats Envoi potentiel d'alertes			
Pré-requis	Un serveur fonctionnel, avec une connexion internet permettant de recevoir des flux HTTP depuis d'autres services web.			
Initialisation	On lance le script de démarrage de l'application afin qu'elle s'installe toute seule.			
Scénario :				
ID	Démarche	Modules	Comportement attendu	Exigences Vérifiées
1	On lance le script de démarrage.	Module complet	Le système se lance et le module analytique effectue correctement son travail.	OP-1 OP-4
Rapport de test		<input type="checkbox"/> Testé par :		Le :
Fonctionnalité :		Conformité :		Ergonomie :
<input type="checkbox"/> Excellente <input type="checkbox"/> Bonne <input type="checkbox"/> Moyenne <input type="checkbox"/> Faible		<input type="checkbox"/> Excellente <input type="checkbox"/> Moyenne <input type="checkbox"/> Faible		<input type="checkbox"/> Excellente <input type="checkbox"/> Moyenne <input type="checkbox"/> Faible

[UC-002] Détection de données erronées				
Objectif	Vérifier que le système détecte correctement les données erronées.			
Éléments à tester	Réception des données externes Conversion des données dans le bon format Vérification de la validité des données			
Pré-requis	Un serveur fonctionnel, avec une connexion internet permettant de recevoir des flux HTTP depuis d'autres services web La réception de données erronées depuis le flux HTTP			
Initialisation	On envoie des données invalides depuis le flux HTTP			
Scénario :				
ID	Démarche	Modules	Comportement attendu	Exigences Vérifiées
1	On envoie des données erronées depuis la vue de l'application ou par le biais d'un flux HTTP	Module E/S	Le système reçoit une donnée depuis l'application elle-même ou depuis un flux HTTP	OP-1 IN-1
2	Le système traite et convertit le flux en données exploitable	Module C/V	Le système a converti les données dans le format du système	OP-5 IN-1
3	Le système effectue la vérification des données	Module C/V	Le système vérifie la validité des données et prévient l'utilisateur que les données sont invalides (si celles-ci sont invalides)	OP-5 IN-1
4	Le système stocke les données afin de les traiter ultérieurement	Module de Gestion des Données	Le système stocke les données dans une base de données dédiée au stockage des données non traitables ou non acceptables	OP-3
5	Le système envoie une alerte à l'utilisateur pour le prévenir d'une donnée erronée	Module E/S	Le système envoie une alerte via un canal choisi	OP-2 IN-2
Rapport de test		□ Testé par :		Le :
Fonctionnalité :		Conformité :		Ergonomie :
<input type="checkbox"/> Excellente <input type="checkbox"/> Bonne <input type="checkbox"/> Moyenne <input type="checkbox"/> Faible		<input type="checkbox"/> Excellente <input type="checkbox"/> Moyenne <input type="checkbox"/> Faible		<input type="checkbox"/> Excellente <input type="checkbox"/> Moyenne <input type="checkbox"/> Faible

[UC-004] Entraînement du système

Objectif	Vérifier que le système puisse être entraîné depuis l'application web par un entraîneur.
Éléments à tester	Réception des données internes Conversion des données dans le bon format Vérification des données valides Analyse des données Apprentissage du module analytique Sauvegarde des données dans la base de données Affichage des résultats du traitement sur l'application web
Pré-requis	Le système doit être opérationnel, l'opérateur doit avoir accès au service web de l'application afin d'envoyer les données à la machine d'apprentissage
Initialisation	L'opérateur se connecte au serveur web via un navigateur web, et possède un jeu de données qu'il souhaite envoyer à la machine

Scénario :

ID	Démarche	Modules	Comportement attendu	Exigences Vérifiées
1	On se connecte au système depuis un navigateur web et la page principal de l'application s'ouvre	Module d'Interface Utilisateur	L'application web s'ouvre et l'opérateur se retrouve sur la page principal de l'application web	OP-1 IN-1 IN-3
2	L'entraîneur clique sur un lien permettant d'afficher la page d'envoi de jeux de données	Module d'Interface Utilisateur	L'application web affiche une page permettant à l'entraîneur d'envoyer des jeux de données	OP-5 IN-1 IN-3
3	L'opérateur prépare les jeux de données à envoyer et les envoie depuis l'application web	Module d'Interface Utilisateur / Module E/S	L'opérateur envoie les données qu'il souhaite faire traiter par le système	OP-5 IN-1 IN-3
4	Le système traite et convertit le flux en données exploitables	Module de C/V	Le système a converti les données dans le format du système	OP-5
5	Le système effectue la vérification des données	Module de C/V	Le système vérifie la validité des données et prévient l'utilisateur que les données sont invalides (si celles-ci sont invalides)	OP-5
6	Le système effectue l'analyse des données afin de modifier son comportement ou d'apprendre de nouveaux flux d'entrée	Module Analytique	Le système exécute l'analyse des données que lui envoie l'opérateur de l'application	OP-5
7	Le système sauvegarde les modifications de son comportement dans la base de données correspondante	Module de Gestion des Données	Le système stocke la modification de son comportement dans la base de données correspondante	OP-3
Rapport de test		<input type="checkbox"/> Testé par :	Le :	
Fonctionnalité :		Conformité :		Ergonomie :
<input type="checkbox"/> Excellente <input type="checkbox"/> Bonne <input type="checkbox"/> Moyenne <input type="checkbox"/> Faible		<input type="checkbox"/> Excellente <input type="checkbox"/> Moyenne <input type="checkbox"/> Faible		<input type="checkbox"/> Excellente <input type="checkbox"/> Moyenne <input type="checkbox"/> Faible

[UC-005] Consultation des données

Objectif	Vérifier qu'un utilisateur puisse consulter les données du module analytique
----------	--

Éléments à tester	Accès au système depuis un navigateur web Consulter l'état de la base de données			
Pré-requis	Le système doit être en fonctionnement Un navigateur web est requis, ainsi que la présence de la base de données			
Initialisation	On se connecte à l'application depuis un navigateur web présente dans le même réseau interne			
Scénario :				
ID	Démarche	Modules	Comportement attendu	Exigences Vérifiées
1	On se connecte au système depuis un navigateur web et la page principal de l'application s'ouvre	Module d'Interface Utilisateur	L'application web s'ouvre et l'opérateur se retrouve sur la page principal de l'application web	OP-1 IN-1 IN-3
2	L'entraîneur clique sur un lien affichant les données	Module d'Interface Utilisateur	L'entraîneur arrive sur la page de consultation des différentes tables	OP-5 IN-1 IN-3
3	L'entraîneur choisit les données qu'il souhaite consulter depuis une liste	Module d'Interface Utilisateur	L'entraîneur choisit les données et la page charge le contenu à afficher	OP-5 IN-1 IN-3
4	Le système interroge la table mentionnée ci-dessus	Module E/S, Module de Gestion des Données	Le système consulte la base de donnée souhaitée par l'entraîneur	OP-3
5	Le système renvoie le contenu de la table	Module d'E/S	Le système renvoie le résultat de la requête vers la page de consultation	OP-3
6	Le résultat est affiché sur la page de consultation	Module d'Interface Utilisateur	La page affiche le contenu de la table	OP-3 IN-3
Rapport de test		<input type="checkbox"/> Testé par :		Le :
Fonctionnalité :		Conformité :		Ergonomie :
<input type="checkbox"/> Excellente <input type="checkbox"/> Bonne <input type="checkbox"/> Moyenne <input type="checkbox"/> Faible		<input type="checkbox"/> Excellente <input type="checkbox"/> Moyenne <input type="checkbox"/> Faible		<input type="checkbox"/> Excellente <input type="checkbox"/> Moyenne <input type="checkbox"/> Faible

[UC-006] Ajustement du comportement

Objectif	Vérifier que le système détecte correctement les données erronées			
Éléments à tester	Réception des données externes Conversion des données dans le bon format Vérification des données valides			
Pré-requis	Un serveur fonctionnel, avec une connexion internet permettant de recevoir des flux HTTP depuis d'autres services web La réception de données erronées depuis le flux HTTP			
Initialisation	On se connecte à l'application depuis un navigateur web présent dans le même réseau interne			
Scénario :				
ID	Démarche	Modules	Comportement attendu	Exigences Vérifiées
1	On se connecte au système depuis un navigateur web et la page principal de l'application s'ouvre	Module d'Interface Utilisateur	L'application web s'ouvre et l'opérateur se retrouve sur la page principal de l'application web	OP-1 IN-1 IN-3
2	Le testeur clique sur un lien permettant la modification du comportement de la machine	Module d'Interface Utilisateur	Le testeur arrive sur une page comportant l'ensemble des options de modification du comportement de la machine	OP-5 IN-1 IN-3
3	Le testeur envoie les modifications de comportement de la machine	Module d'Interface Utilisateur Module E/S	La page de modification du comportement se met en attente de confirmation du comportement de la machine d'apprentissage	OP-5 IN-1 IN-3
4	Le module d'analyse répercute les modifications sur son comportement en lançant une nouvelle phase d'apprentissage	Module Analytique	Le système exécute une phase complète d'analyse et d'apprentissage afin de corriger son comportement.	OP-3
5	Le module d'analyse envoie une confirmation de la modification de son comportement	Module Analytique Module E/S Module d'Interface Utilisateur	La page de modification du comportement affiche le résultat de la modification	IN-1 IN-3
Rapport de test		☐ Testé par :		Le :
Fonctionnalité :		Conformité :		Ergonomie :
☐ Excellente ☐ Bonne ☐ Moyenne ☐ Faible		☐ Excellente ☐ Moyenne ☐ Faible		☐ Excellente ☐ Moyenne ☐ Faible