Etude sur la sécurité des logiciels médicaux

Réalisée par Serma Ingenierie à la demande de l'ANSM

Juillet 2016

TABLE DES MATIERES

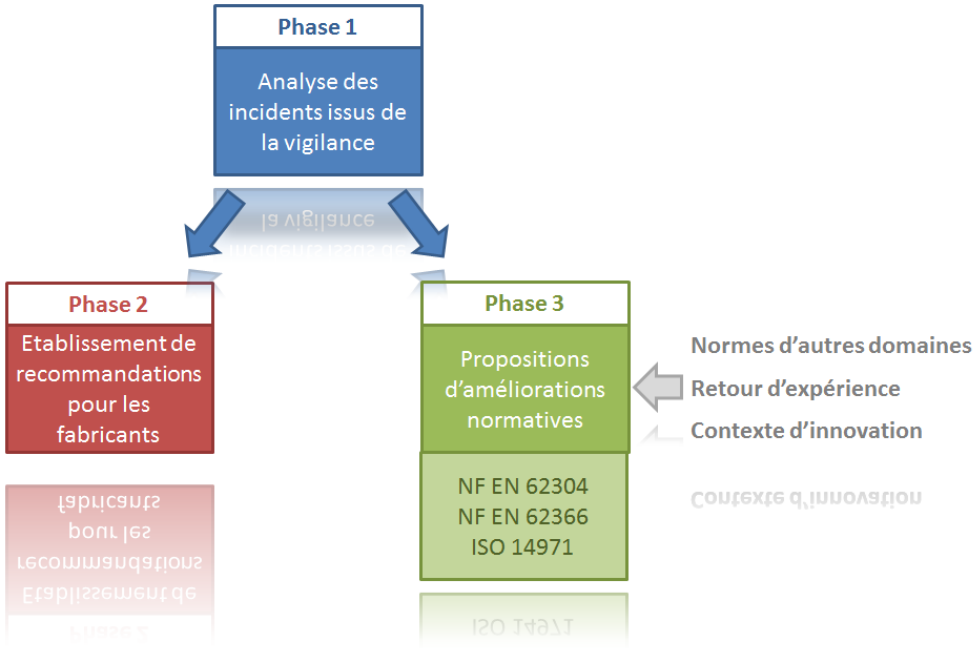
RESUME

Afin de répondre à l'importance prise par le Logiciel dans les applications médicales, l'ANSM a souhaité lancer une étude sur la sécurité des Logiciels. Cette étude réalisée entre août 2014 et novembre 2015 par la société SERMA INGENIERIE a pour principaux objectifs :

- de compléter les réflexions sur la sécurité des Logiciels au niveau normatif,
- de mesurer la pertinence et la suffisance de l'environnement normatif pour le développement Logiciel,
- d'apporter des recommandations sur l'application des normes aux fabricants de Logiciels.

La démarche d'analyse retenue pour cette étude a comporté 3 grandes phases et s'est appuyée sur :

- les incidents issus des vigilances (matériovigilance, réactovigilance, pharmacovigilance),
- les référentiels existants du domaine médical,
- l'état de l'art normatif d'autres domaines,
- les retours d'expérience SERMA INGENIERIE et ANSM,
- les innovations dans le domaine médical.



Les principales **recommandations** à l'attention des fabricants de Logiciels de DM (Phase 2) détaillées dans ce rapport ont porté sur :

- Les bases de données
- La vérification des Spécification / Conception Architecturale
- L'analyse d'impact d'une modification
- La gestion et suivi des risques

Des **propositions d'amélioration** au niveau normatif (Phase 3) ont également été formulées. Celles-ci se sont focalisées sur les normes existantes (NF EN 62304, NF EN 62366, ISO 14971). Ces propositions d'amélioration sont de 2 natures :

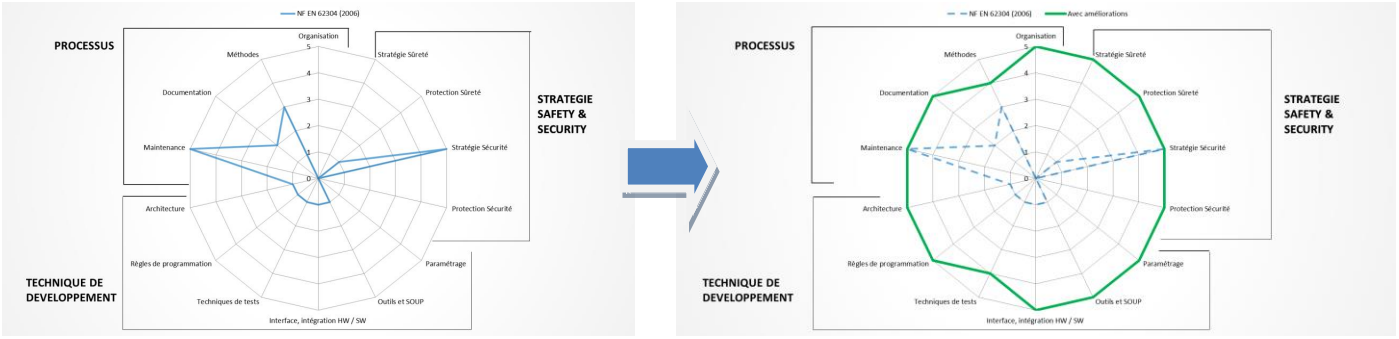
- Des compléments aux exigences des normes actuelles
- La création de nouvelles exigences

Les principales améliorations proposées portent :

- Pour la norme NF EN 62304 (Logiciels de dispositifs médicaux – Processus du cycle de vie du Logiciel), sur :

Processus	- La structure organisationnelle, l'indépendance des équipes et les compétences des intervenants - La définition du processus et de la stratégie d'Evaluation - Les précisions à apporter dans l'activité de conception détaillée (Unités Logicielles)
Techniques de développement	- L'ajout d'exigences sur les interfaces du Logiciel et son intégration - L'ajout d'exigences pour les Logiciels configurés par données d'application - La définition de contraintes architecturales - L'apport de techniques présentées sous forme de tableau - La définition de règles de programmation et des vérifications du code associées - La présentation de techniques de tests - La clarification liée à la réalisation des activités de tests unitaires - L'ajout de la qualification des outils
Safety et Security	- La définition de la stratégie concernant la Sûreté du Logiciel - La définition des techniques concernant la Sûreté du Logiciel

L'amélioration de la couverture[1] de la norme pour cette norme peut être symbolisée par les graphes ci-après :



- Pour la norme NF EN 62366 (Application de l'ingénierie de l'aptitude à l'utilisation aux dispositifs médicaux), sur :

  - L'amélioration des exigences relatives à la formation pour les utilisateurs
  - L'apport de précisions sur le contenu de la notice d'utilisation

- Pour la norme ISO 14971 (Application de la gestion des risques aux dispositifs médicaux), sur :

  - L'ajout des activités de Sûreté (Security) dans la Gestion de Risques

L'ensemble des recommandations et améliorations formulées dans ce rapport ont été établies afin d'apporter une meilleure maîtrise de la sécurité des Logiciels médicaux. Elles considèrent également l'impact des innovations futures dans ce domaine.

[1] Pour les critères et notes présents sur le diagramme en radar, se reporter à l'annexe 2 de ce rapport

1. PRESENTATION DE L'ANSM

L'Agence nationale de sécurité du médicament et des produits de santé (ANSM), créée par la loi du 29 décembre 2011 relative au renforcement de la sécurité sanitaire du médicament et des produits de santé, a été mise en place le 1er mai 2012.

Elle est chargée d'évaluer les bénéfices et les risques liés à l'utilisation des produits de santé tout au long de leur cycle de vie.

Au sein de l'ANSM, la direction des Dispositifs Médicaux de Diagnostics et des Plateaux Techniques (DMDPT) est en charge des dispositifs médicaux de diagnostic, de radiothérapie, des Logiciels, ainsi que des plateaux techniques (dispositifs d'anesthésie réanimation, de suppléance fonctionnelle, de bloc opératoire).

Outre son intervention sur l'évaluation des bénéfices et des risques liés à l'utilisation des produits de santé entrant dans son champ de compétence, la DMDPT assure également la veille scientifique et technique ainsi que la production de l'information scientifique. Elle participe également aux activités européennes et internationales en accord avec les axes stratégiques de l'Agence et se fait le porte-parole de l'Agence, en lien avec la direction de la communication, sur l'information à apporter sur ces sujets.

2.OBJET DU DOCUMENT

2.1. INTRODUCTION

Ce document constitue le rapport final de l'étude réalisée par SERMA INGENIERIE sur la sécurité des Logiciels médicaux (marché ANSM n°2014C029 du 04/08/2014) entre Août 2014 et Novembre 2015.

Afin de répondre à l'importance prise par le Logiciel dans les applications médicales, l'ANSM a souhaité lancer une étude sur la sécurité des Logiciels intégrant :

- les Logiciels de dispositifs médicaux,
- certains Logiciels utilisés dans les laboratoires de biologie médicale qui entrent désormais dans le champ de compétence de l'agence,
- les Logiciels d'aide à la prescription de médicaments pour lesquels l'ANSM reçoit des signalements.

En effet, les Logiciels sont de plus en plus présents dans le domaine des dispositifs médicaux, tant au niveau des Logiciels autonomes, ayant le statut de dispositif médical, qu'au niveau des Logiciels embarqués dans des dispositifs médicaux.

Les principaux objectifs de cette étude sont :

- de compléter les réflexions européennes sur la sécurité des Logiciels aux niveaux réglementaires et normatifs à partir de l'expérience acquise dans d'autres secteurs industriels ayant également recours à des Logiciels spécifiques,
- de mesurer la pertinence et la suffisance de l'environnement normatif à disposition des éditeurs de ces Logiciels, notamment au regard du retour d'expérience issu des déclarations d'incidents faites par le biais de la matériovigilance, de la réactovigilance ou de la pharmacovigilance,
- d'apporter des recommandations sur l'application des normes aux fabricants de Logiciels.

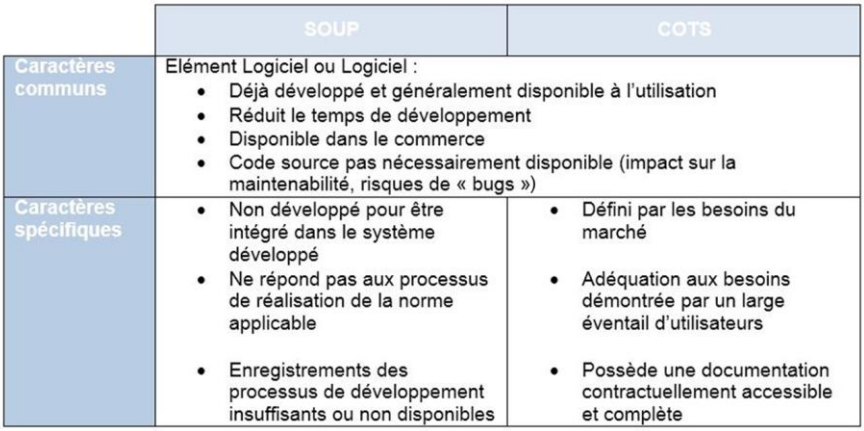
2.2. DOCUMENTS DE REFERENCE

Repère	Nom	Désignation
[CEI 61508]	CEI 61508 (2011)	Sécurité fonctionnelle des systèmes électriques/électroniques/électroniques programmables relatifs à la sécurité
[ISO 14971]	NF EN ISO 14971 (2007)	Application de la gestion des risques aux dispositifs médicaux
[TR 80002-1]	Technical Report related to 14971 CEI/TR 80002-1:2009	Guide d'application de la norme ISO 14971 aux Logiciels des dispositifs médicaux (Partie 1).
[NF EN 60601-1]	NF EN 60601-1 (2014)	Appareils électromédicaux Partie 1 : Exigences générales pour la sécurité de base et les performances essentielles
[ISO 13485]	ISO 13485 (2012)	Dispositifs médicaux — Systèmes de management de la qualité — Exigences à des fins réglementaires
[93/42/CEE]	93/42/CEE (1993)	Directive du conseil relative aux dispositifs médicaux
[NF EN 62304]	NF EN 62304 (2006)	Logiciels de dispositifs médicaux – Processus du cycle de vie du Logiciel
[NF EN 62366]	NF EN 62366 (2008)	Dispositifs médicaux – Application de l'ingénierie de l'aptitude à l'utilisation aux dispositifs médicaux
[EN 50128]	EN 50128 (2011)	Applications ferroviaires – Systèmes de signalisation, de télécommunication et de traitement – Logiciels pour systèmes de commande et de protection ferroviaire
[ISO 26262]	ISO 26262 (2011)	Véhicule routier – Sécurité fonctionnelle
[CEI 60880]	CEI 60880 (2006)	Centrales nucléaires de puissance – Instrumentation et contrôle-commande importants pour la sûreté – Aspects Logiciels des systèmes programmés réalisant des fonctions de catégorie A
[EN 50159]	EN 50159 (2011)	Applications ferroviaires - Systèmes de signalisation, de télécommunication et de traitement - Partie 2 : Communication de sécurité sur des systèmes de transmission ouverts
[FDA_Cybersecurity]	FDA Cybersecurity 02 octobre 2014	Content of Premarket Submissions for Management of Cybersecurity in Medical Devices

2.3. LISTE DES ABREVIATIONS

Acronyme	Définition
ADAS	Advanced Driver Assistance Systems - système d'aide à la conduite
AEEL	Analyse des Effets des Erreurs du Logiciel
AMDEC	Analyse des Modes de Défaillance, de leurs Effets et de leur Criticité
ANSM	Agence Nationale de Sécurité du Médicament et des produits de santé
APR	Analyse Préliminaire de Risques
ASIL	Automotive Safety Integrity Level - Niveau d'intégrité de sécurité du Automobile
COTS	Commercial Off-the-Shelf - Composant Logiciel ou Matériel disponible dans le commerce
DM	Dispositif Médical
EL	Elément Logiciel [NF EN 62304]: toute partie identifiable d'un programme informatique [ISO/CEI 90003:2004, définition 3.14, modifiée]
HW	Hardware (partie liée au matériel)
IHM	Interface Homme Machine
IMA	logiciel d'IMAgerie médicale
LABM	Logiciel d'Automates de Biologie Médicale
LAP	Logiciel d'Aide à la Prescription
MMR	Mesures de Maîtrise des Risques
REX	Retour d'EXpérience
RT	Logiciel de RadioThérapie
SGBD	Système de Gestion de Base de Données
SILBM	Système Informatique de Laboratoire de Biologie Médicale
SSIL	Software Safety Integrity Level - Niveau d'intégrité de sécurité du Logiciel
SOUP	Software Of Unknown Provenance - logiciel de provenance inconnue
SW	Software (partie liée au Logiciel)
TU	Tests Unitaires
TI	Tests d'Intégration
UL	Unité Logicielle [NF EN 62304]: Elément Logiciel qui n'est pas subdivisé en d'autres éléments.

2.4. TERMINOLOGIE

Termes	Définitions / Précisions
Activités (phases de développement)	Ensemble d'une ou de plusieurs tâches corrélées ou interactives - Source [NF EN 62304]
Catégorie	Correspond à la « Catégorie » d'appartenance du DM : cette information définit la destination d'utilisation du Logiciel (ex. : Logiciel de radiothérapie, Logiciel d'imagerie médicale,…).
Classe de sécurité du Logiciel	La classe de sécurité du Logiciel doit être attribuée par le fabricant en fonction des effets possibles sur le patient, l'opérateur ou d'autres personnes résultant d'un phénomène dangereux auquel le système Logiciel peut contribuer. Classe A : Aucune blessure non grave n'est possible Classe B : Une blessure non grave est possible Classe C : La mort ou une blessure grave est possible (Source [NF EN 62304])
Comparaison entre COTS et SOUP	
Incident	Incident déclaré, relevé sur un dispositif en cours d'exploitation chez l'exploitant/utilisateur ou lors de tests chez le fabricant.
Logiciel générique	Logiciel pouvant être utilisé pour une grande variété d'installations simplement en fournissant des données et/ou algorithmes de configuration propres à l'application - Source [EN 50128]
Logiciel préexistant	Logiciel développé avant l'application dont il est ici question, incluant les Logiciels SOUP/COTS (standards disponibles dans le commerce) et libres - Source [EN 50128]
Processus	Ensemble d'activités corrélées ou interactives qui transforme des éléments d'entrée en éléments de sortie - Source [NF EN 62304]
Risque	Combinaison de la probabilité d'un dommage et de sa gravité [ISO/CEI Guide 51:1999, définition 3.2]

Termes	Définitions / Précisions
Sécurité / Safety	Assurer que le Logiciel a été conçu pour prévenir les risques aléatoires et involontaires. D'après [ISO/CEI Guide 51:1999, définition 3.1] : Absence de risque inacceptable
Situations dangereuses	Situation dans laquelle des personnes, des biens ou l'environnement sont exposés à un ou à plusieurs phénomènes dangereux [ISO/CEI Guide 51:1999, définition 3.6] NOTE : Voir l'Annexe E de la norme [ISO 14971] pour une explication de la relation entre «phénomène dangereux» et «situation dangereuse».
Sûreté / Security	Assurer que le Logiciel a été conçu pour résister à toutes les attaques malveillantes. D'après [ISO/CEI 12207:1995, définition 3.25] : Protection des informations et des données de sorte que des personnes ou des systèmes non autorisés ne puissent les lire ou les modifier et que l'accès à ces informations et données ne soit pas refusé à des personnes ou des systèmes autorisés
Tâches	Partie unique d'un travail qui doit être effectué - Source [NF EN 62304]

3.PRESENTATION GENERALE DE L'ETUDE

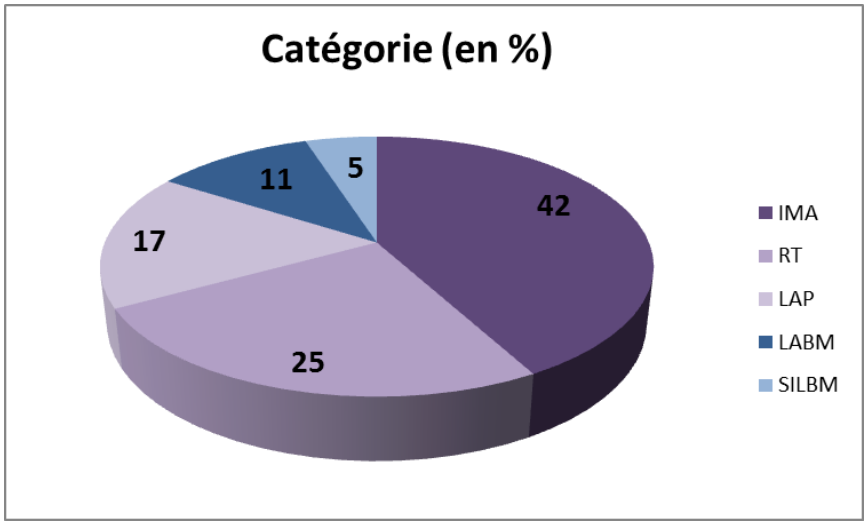
3.1. DEMARCHE

L'étude réalisée est composée des 3 grandes phases suivantes :
- Phase 1 : Réalisation d'une analyse d'incidents fournis par l'ANSM (état des lieux)
- Phase 2 : Proposition de recommandations pour les fabricants
- Phase 3 : Axes d'améliorations réglementaires et normatives

3.1.1. PHASE 1 : ETAT DES LIEUX

Le suivi mis en place par les autorités dans le cadre des vigilances (dont matériovigilance, réactovigilance, pharmacovigilance) a permis de remonter différents incidents qui ont été signalés et regroupés par l'ANSM.
L'ensemble des incidents concerne des Logiciels en exploitation. Certains incidents sont remontés directement par les utilisateurs, d'autres par les fabricants. Les fabricants peuvent avoir identifié des incidents après signalement des exploitants ou après constat d'une anomalie lors de tests internes, réalisés après la mise en exploitation du Logiciel.

Sur la base d'un échantillon de 156 incidents proposés par l'ANSM, SERMA INGENIERIE en a effectué une analyse détaillée. Cette analyse a porté sur les catégories suivantes :



Les incidents ainsi analysés ont été classés en deux groupes :

- Incidents liés au non-respect des normes.
- Incidents liés à la non-complétude des normes.

Ces informations ont ensuite été utilisées pour la réalisation des phases 2 et 3.

3.1.2. PHASE 2 : PROPOSITION DE RECOMMANDATIONS POUR LES FABRICANTS

La première catégorie d'incidents identifiés lors de la phase 1 a permis d'élaborer des recommandations à destination des fabricants de dispositifs médicaux afin de faciliter la compréhension et la mise en œuvre des normes [NF EN 62304], [ISO 14971] et [NF EN 62366].

Une synthèse de ces recommandations est présente au chapitre 4.1. Le détail de ces recommandations a été regroupé sous forme de fiches intégrées à ce rapport au chapitre 5.2.

3.1.3. PHASE 3 : AXES D'AMELIORATIONS REGLEMENTAIRES ET NORMATIVES

Des propositions d'axes d'amélioration pour les normes [NF EN 62304], [ISO 14971] et [NF EN 62366] ont été définies à partir de l'analyse de 3 normes issues de domaines d'application différents. En complément, la seconde catégorie d'incidents identifiés lors de la phase 1 (non-complétude des normes) a permis de conforter ces propositions.

Une analyse détaillée de 8 normes issues de différents domaines et portant sur le développement de Logiciels critiques a été réalisée.

Les référentiels examinés et les domaines associés ont été les suivants :

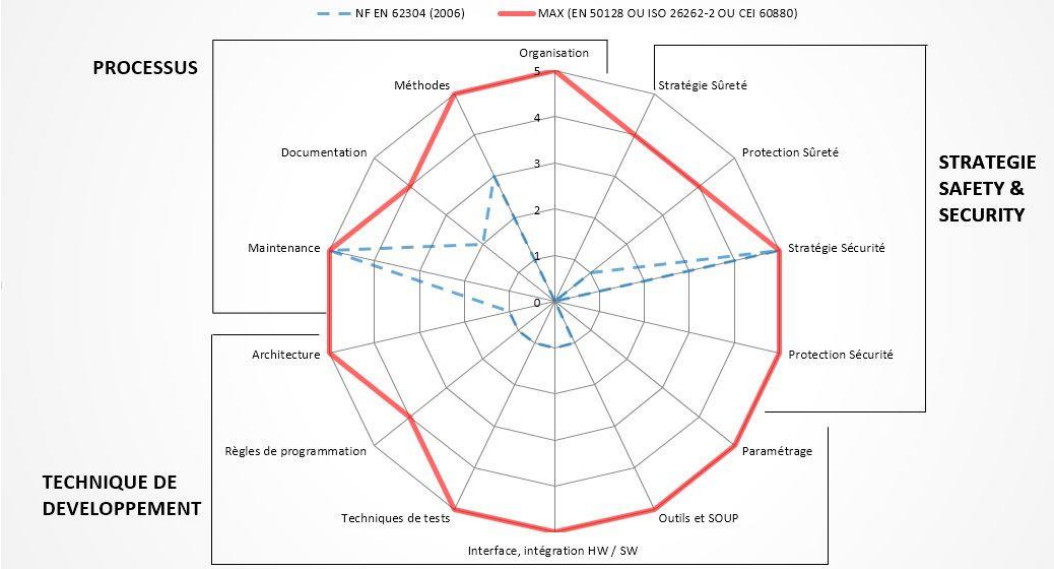
Norme	Date	Domaine
EN 50128	**2011**	**Ferroviaire**
ISO 26262-6	**2011**	**Automobile**
CEI 60880	**2006**	**Nucléaire**
DO-178C	**2012**	**Aéronautique**
CEI 62138	**2009**	**Nucléaire**
CEI 61511	**2003**	**Process**
CEI 62061	**2005**	**Machines**
ECSS-Q-ST-80-C	**2009**	**Spatial**

Une étude multicritères a ainsi pu mettre en évidence les apports potentiels de chaque référentiel par rapport à la norme [NF EN 62304] actuelle.

Les critères analysés ont été classés suivant 3 catégories telles que présentées ci-après :

Catégories	Critères
Processus	**Organisation**
	Méthodes
	Documentation
	Maintenance
Techniques de développement	**Architecture**
	Règles de programmation
	Techniques de tests
	Interface, intégration HW / SW
	Outils & SOUP
	Paramétrage
Stratégie Safety & Security	**Protection Sécurité**
	Protection Sûreté
	Stratégie Sécurité
	Stratégie Sûreté

L'analyse réalisée a mis en évidence l'apport des normes [EN 50128], [ISO 26262] et [CEI60880] dans cette démarche. Une présentation synthétique de ces normes est fournie en annexe 1 de ce rapport. Leur apport est symbolisé sur le graphe ci-après :



Une description des critères et des notes utilisés sur ce diagramme en radar est donnée en annexe 2 de ce rapport.

La ***norme [EN 50128]*** (ferroviaire) est la norme qui apporte le plus de détail concernant les aspects Processus (Organisation, Méthodes, Documentation, Paramétrage …) ainsi qu'un bon niveau concernant les Techniques de développement.

La ***norme [ISO 26262-6]*** (automobile) permet de compléter les aspects Techniques de développement notamment concernant l'Architecture et l'interface Matériel / Logiciel.

La ***norme [CEI 60880]*** (nucléaire) a été retenue car elle intègre la notion de Sûreté / Security. Des mesures sont demandées afin de prendre en compte des possibles attaques intentionnelles extérieures et d'assurer la protection des informations. Dans le contexte médical, ce point n'est pas négligeable vis-à-vis notamment de la problématique du secret médical et de la malveillance entraînant un risque pour le patient. Afin de compléter ou préciser certains points, la prise en compte de prescriptions présentes dans le [FDA_Cybersecurity] a également été étudiée.

En résumé, la stratégie retenue pour l'analyse peut être synthétisée de la façon suivante :

1

Comparaison avec [EN 50128] et [ISO 26262-6] :

- Comparaison de toutes les clauses [EN 50128] par rapport à [NF EN 62304]
- Comparaison de toutes les clauses [ISO 26262-6] par rapport à [NF EN 62304]

2

Ouverture du périmètre initial [NF EN 62304] avec [CEI 60880] :

- Identification de notions non couvertes par les 2 normes (ex : Security)
- Comparaison de la couverture de ces notions avec [NF EN 62304]

De la même manière que pour la phase 2, une synthèse de ces améliorations est présente au chapitre 4.2. Le détail de ces améliorations a été regroupé sous forme de fiches intégrées à ce rapport aux chapitres 5.3, 5.4 et 5.5.

4.SYNTHESE DE L'ETUDE

L'analyse réalisée a abouti à la formulation de recommandations et d'améliorations faisant l'objet de la présente synthèse. Pour chacune de ces recommandations / améliorations, une fiche détaillée d'analyse a été réalisée et est fournie au chapitre suivant.

4.1. SYNTHESE DES RECOMMANDATIONS AUX FABRICANTS DE LOGICIELS

Les recommandations proposées dans ce rapport ont été réalisées sur la base des incidents analysés qui ne sont pas à l'origine d'une proposition d'amélioration normative. Elles constituent des conseils aux fabricants de Logiciels de dispositifs médicaux pour la mise en application de certaines exigences des normes [NF EN 62304] et [ISO 14971].

Les synthèses des recommandations sont les suivantes :

- Base de données - [NF EN 62304] :
  Lorsque le Logiciel possède une Base de Données, la spécification des exigences du Logiciel doit identifier des principes sur les données (pérennité, intégrité, unicité et confidentialité). L'utilisation d'un SGBD reconnu et de règles de bonne pratique est recommandée.

- Vérification spécification / conception architecturale - [NF EN 62304] :
  Il est recommandé d'effectuer et de formaliser les relectures des documents produits, en suivant plusieurs axes de relecture (complétude, pertinence, cohérence, traçabilité, …).

- Analyse d'impact d'une modification - [NF EN 62304] :
  Toute modification du Logiciel doit faire l'objet d'une analyse d'impact pour éviter les défauts induits par cette modification. Il est recommandé de traiter les points suivants : Impact sur MMR existantes, ajout de MMR supplémentaires, Eléments Logiciels (EL) impactés, tests de la modification et de non régression.

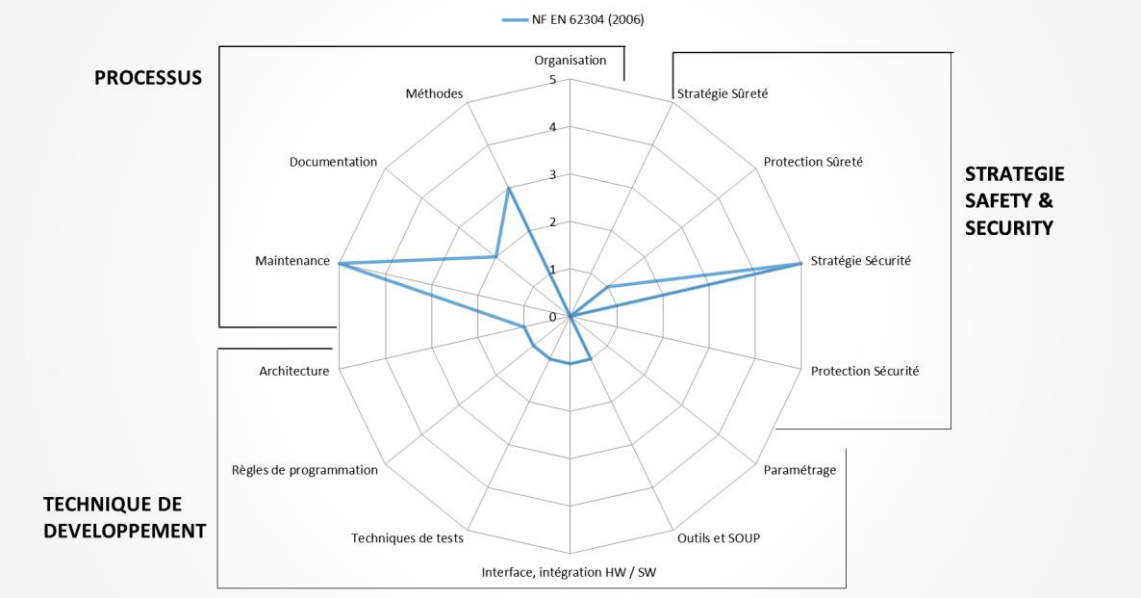
- Gestion et suivi des risques - [NF EN 62304] / [ISO 14971] :
  Des analyses de risque sont à réaliser au niveau du DM ainsi qu'au niveau du Logiciel (AMDEC du Logiciel). La réalisation d'un document de suivi des Mesures de Maîtrise de Risque est recommandée afin de statuer sur l'implémentation et le test de ces MMR.
  L'utilisation du guide d'application de la norme ISO 14971 [TR 80002-1] est recommandée.

4.2. SYNTHESE DES PROPOSITIONS D'AMELIORATIONS NORMATIVES

Les améliorations proposées dans ce rapport ont pour objectif de couvrir le périmètre le plus large possible en terme de typologie de Logiciel (contrainte temps-réel ou non). Les activités de développement associées à ces différentes typologies pourront faire l'objet d'une adaptation lors de l'élaboration d'une évolution de la norme actuelle.
Par exemple, sur les Logiciels n'ayant pas de contrainte temps-réel, les activités concernant la conception détaillée et les tests unitaires peuvent ne pas être obligatoire en fonction des risques identifiés à travers les activités de gestion de risque définies dans la norme [ISO 14971].

4.2.1. AMELIORATION [NF EN 62304] ET [ISO 14971]

Pour rappel, le positionnement de la norme [NF EN 62304] au regard de l'état de l'art a conduit au constat ci-après :



Les améliorations proposées permettent de combler les manques identifiés dans la norme [NF EN 62304]:

4.2.1.1. Catégorie Processus - [NF EN 62304]

Pour la catégorie Processus, ces améliorations portent sur les aspects suivants :

- Structure organisationnelle :
  La structure organisationnelle doit être définie en fonction de la classe du Logiciel en précisant le rôle et les responsabilités des différents intervenants ainsi que l'indépendance entre les équipes. La compétence des intervenants est à démontrer (connaissances techniques, expérience et formation appropriée).

- Evaluation/Certification :
  Un processus d'évaluation doit être ajouté dans la norme. L'évaluateur/certificateur doit réaliser un Plan d'Evaluation Logiciel (description de la stratégie d'évaluation) et un Rapport d'Evaluation Logiciel statuant sur la conformité à la norme, sur les Mesures de Maîtrises de Risques et sur les anomalies identifiées.

- Traçabilité globale :
  Une traçabilité entre les différentes activités (intégrant les MMR) doit être mise en place, via une traçabilité descendante / montante (de la Spécification des Exigences du Logiciel jusqu'au code) et via une traçabilité horizontale afin de mettre en relation les tests du Logiciel avec les documents de développement.

- Documents à produire :
  Un tableau identifiant les documents à produire par activité de développement en fonction de la classe du Logiciel est à ajouter dans la norme. Pour chaque activité dans le corps de la norme, il faut identifier les points d'entrée et les points de sorties.

- Contraintes de performance et d'environnement :
  Les contraintes de dimensionnement, de performance, les caractéristiques physiques, l'environnement informatique dans lequel le Logiciel doit fonctionner sont à définir et à contrôler lors de l'installation.

- Installation :
  Des tests spécifiques doivent être réalisés et formalisés pour s'assurer que le Logiciel est installé correctement en prenant en considération l'environnement d'exécution, les performances requises et en démontrant un respect des fonctionnalités.

- Conception détaillée :
  La conception détaillée doit décrire les Unités Logicielles (UL) et ces interfaces (définition des entrées / sorties, des domaines de définitions, valeurs aux limites, algorithmes, interfaces d'appel).

4.2.1.2. Catégorie Technique de Développement - [NF EN 62304]

Pour la catégorie Technique de Développement, ces améliorations portent sur les aspects suivants :

- Techniques de développement générales :
  Des tableaux doivent être intégrés dans la norme afin de détailler les techniques à mettre en œuvre pour réaliser les activités du processus de développement. Les techniques à appliquer doivent être fonction de la classe du Logiciel.

- Contraintes architecturales :
  Une description détaillée de l'architecture de sécurité et de la stratégie de tolérances aux fautes est à mettre en place. Afin d'assurer la robustesse de l'architecture du Logiciel plusieurs approches sont possibles (redondance, ségrégation, programmation défensive, interfaces limitées, mécanismes de détection et de gestion des erreurs).

- Contraintes d'implémentations et vérifications UL :
  Des contraintes sont à lister dans la norme de façon non-exhaustive concernant les langages de programmation qui peuvent être utilisés ainsi que des règles de programmation.
  Pour s'assurer de la cohérence de l'activité de codage, une vérification doit être établie au travers d'une analyse statique de code (vérification des règles de programmation) et de la vérification de la cohérence entre la conception détaillée et le code.

- Techniques de test générales :
  Les contraintes concernant les types de tests, les couvertures de tests, les techniques de tests et relatives à l'environnement de tests sont à définir. Par ailleurs, le chapitre 5.7 de la norme [NF EN 62304] concernant les Essais du Logiciel (Tests de Validation Logiciel) doit être applicable pour les Logiciels de Classe A.

- Activités de Tests Unitaires :
  Pour les Logiciels de Classe C, l'activité de Tests Unitaires est à ajouter dans les exigences de la norme de façon explicite. Cette activité permet de démontrer que la conception détaillée a bien été implémentée dans le code et que le code ne contient pas de fonctionnalité non-désirée.

- Interfaces et intégration Hardware / Software :
  Spécifier les éléments de description des interfaces externes en cohérence avec les contraintes de sécurité dans son environnement matériel et en interface avec d'autres Logiciels. Pour chaque Elément Logiciel, spécifier les éléments de description des interfaces internes. Des tests d'intégration sont à mettre en place pour couvrir les interfaces externes et internes.

- Qualification des outils :
  Une partie spécifique concernant la qualification des outils est à ajouter dans la norme. Les outils sont à classer selon l'impact d'un mauvais fonctionnement. En fonction de cet impact des tâches sont à réaliser pour qualifier l'outil (réalisation d'un Manuel d'Utilisation, validation de l'outil, …). La démonstration d'une utilisation antérieure réussie de l'outil dans des environnements similaires est également possible pour qualifier un outil.

- Regroupement exigences SOUP :
  Un chapitre dédié contenant toutes les exigences applicables au SOUP doit être réalisé.
  Par ailleurs, la notion de Logiciel préexistant est plus adéquate que la notion de SOUP. Sa définition est « Logiciel développé avant l'application dont il est ici question, incluant les Logiciels COTS (standards disponibles dans le commerce) et libres. »

- Logiciels configurés par données d'application / Paramétrage :
  Un chapitre dédié aux Logiciels génériques configurés par des données d'application (paramètres) ainsi que leur vérification et validation doit être ajouté dans la norme. Les Logiciels configurés par des données d'application doivent avoir un niveau de confiance identique selon les différents jeux de données possibles.

4.2.1.3.Catégorie Safety & Security - [NF EN 62304] ET [ISO 14971]

Pour la catégorie Safety & Security, ces améliorations portent sur les aspects suivants :

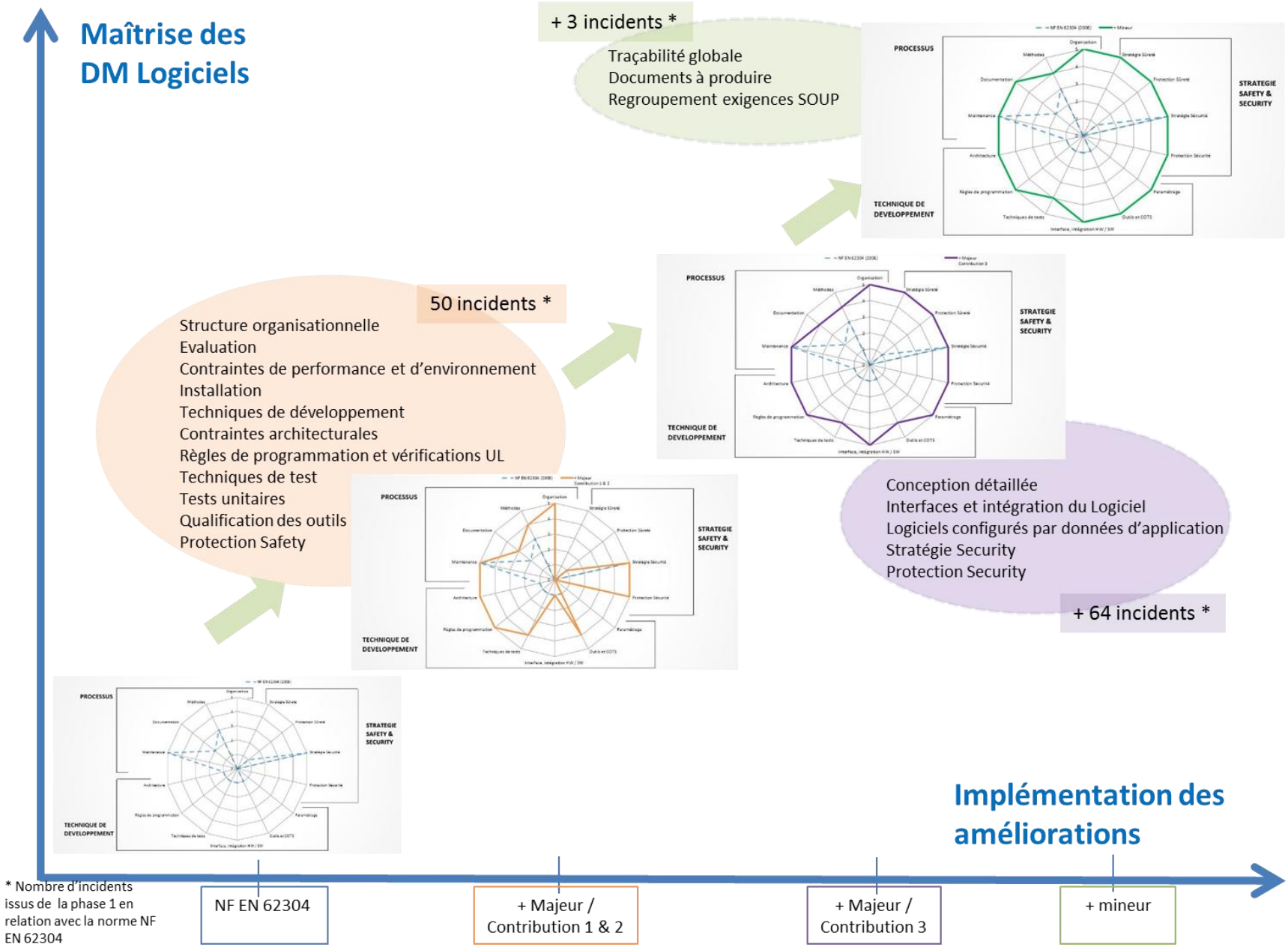
- Auto-surveillance :
  Le Logiciel doit réaliser de l'auto-surveillance afin de surveiller le matériel lors de l'exploitation, à intervalles de temps spécifiés, ainsi que le comportement du Logiciel.
  En cas de détection d'une défaillance, le Logiciel doit donner les réponses appropriées en temps voulu (mise en position de repli, fonctionnement dégradé …).

- Lignes directrices pour stratégie Security :
  Les lignes directrices concernant la stratégie de Sûreté (processus Security) sont à aborder dans la norme [NF EN 62304] mais un référentiel spécifique traitant ce sujet est nécessaire. Un plan de prévention des menaces et un rapport de Sûreté du Logiciel sont à réaliser. De plus, les exigences applicables tout au long du cycle de vie du Logiciel sont à intégrer dans les activités (exigences de contre-mesures, d'authentification, détection de failles de sécurité, …).

- Protection Security :
  Les contraintes de conception et de développement concernant les protections de Sûreté du Logiciel sont à définir (cryptage, contrôle d'intégrité, authentification de la source, absence de chemins cachés). De plus, l'environnement de développement du Logiciel (poste de travail) doit être sécurisé, de manière physique ou numérique (cloisonnement, pare-feu, antivirus, cryptage des données, …).

- Gestion des risques et traitement de la problématique Sûreté (Security)
  La gestion des risques relatifs à la problématique de Sûreté (Security) est à intégrer dans la norme [ISO 14971].
  Le traitement de la problématique de Sûreté (Security) dans sa globalité est à réaliser dans un référentiel spécifique.

Les améliorations proposées peuvent être implémentées de manière évolutive en fonction du degré d'importance et de la contribution pour le fabricant. Cette représentation graphique permet de visualiser l'apport de ses améliorations pour la maîtrise des DM Logiciels.


Maîtrise des DM Logiciels
+ 3 incidents *
Traçabilité globale
Documents à produire
Regroupement exigences SOUP
PROCESSUS
STRATEGIE SAFETY & SECURITY
TECHNIQUE DE DEVELOPPEMENT
50 incidents *
Structure organisationnelle
Evaluation
Contraintes de performance et d'environnement
Installation
Techniques de développement
Contraintes architecturales
Règles de programmation et vérifications UL
Techniques de test
Tests unitaires
Qualification des outils
Protection Safety
Conception détaillée
Interfaces et intégration du Logiciel
Logiciels configurés par données d'application
Stratégie Security
Protection Security
+ 64 incidents *
Implémentation des améliorations
* Nombre d'incidents issus de la phase 1 en relation avec la norme NF EN 62304
NF EN 62304
+ Majeur / Contribution 1 & 2
+ Majeur / Contribution 3
+ mineur

4.2.2. AMELIORATION APTITUDE A L'UTILISATION [NF EN 62366]

Les améliorations proposées portent sur les aspects suivants :

- Interface Utilisateur :
  Il faut introduire des exigences d'interface utilisateur dans la partie normative (§5.7), en se basant sur les informations fournies dans les chapitres D.2.6 (exemples d'exigences d'interface utilisateur), D.4.6.3 (spécifications qui peuvent être incluses pour une interface utilisateur) et du tableau G.7 (indications de conception en ce qui concerne les signaux d'alarmes).

- Notice d'utilisation :
  La fourniture d'une notice d'utilisation du Logiciel est à exiger. De plus, des exigences sur la formalisation d'une notice d'utilisation sont à ajouter au chapitre 6 (contraintes liées au paramétrage, périmètre d'utilisation, les limites et restrictions d'utilisation).

- Formation des Utilisateurs :
  Le chapitre 7 (Formation et supports de formation) de la norme doit être complété afin d'ajouter des contraintes sur le contenu de la formation, sur la vérification du support de formation et sur la preuve de la formation – preuve que le personnel médical a bien suivi une formation et qu'il est apte à utiliser le DM.

4.2.3. AMELIORATION VERSUS INNOVATION

Le monde de la santé connait depuis quelques années une véritable révolution technologique. Cette révolution porte des noms tels que :

- Hôpital numérique
- Télémédecine / téléchirurgie
- Objets connectés
- Nouvelles technologies (nano-implants, multi-cœur, virtualisation, …)

Outre les enjeux économiques voire éthiques, des enjeux réglementaires et normatifs se posent pour l'évaluation de ces dispositifs destinés à des applications médicales.

Les normes actuelles ne doivent pas empêcher l'évolution technologique mais doivent pouvoir essayer d'anticiper la prise en compte de nouveaux risques.
Cet aspect est relativement bien pris en compte dans la norme [ISO 14971] liée à la gestion des risques. En effet, celle-ci définit une démarche générique applicable à toute évolution technologique.

Les impacts sur les DM Logiciels issus du développement de ces nouvelles technologies peuvent ainsi être de différentes natures :

- Impact sur les fonctionnalités et/ou temps de réponse (contrainte Safety et Security) du DM Logiciel lié à l'utilisation des Logiciels multi-application
- Impact sur la maîtrise et/ou la pérennité (contrainte Safety) du DM Logiciel en lien avec l'utilisation accrue de SOUP/COTS
- Impact sur les fonctionnalités et sur les données / confidentialité (Contrainte Safety et Security) lié à une vulnérabilité

4.2.3.1. Utilisation des Logiciels multi-application

Le développement de Logiciels multi-application peut utiliser notamment des technologies basées sur la virtualisation ou l'utilisation de systèmes multi-cœur.
Ces technologies sont une opportunité pour la meilleure prise en compte des contraintes Safety et Security.
Ces solutions favorisent la mise en œuvre sur une même plateforme d'applications de niveaux de criticité différents tout en garantissant leur cohabitation. Ces technologies permettent par ailleurs de limiter les coûts de certification au juste nécessaire en fonction du niveau de criticité de chaque application.
Néanmoins, en fonction des possibilités offertes par ces technologies, certaines limites devront être à considérer telles que :

- La perte du déterminisme
- Les risques associés à la mémoire partagée
- L'exécution temporelle et les données partagées entre tâches

Par ailleurs, l'utilisation de ces technologies n'est pas encore acceptée dans tous les domaines. En effet, plusieurs questions se posent sur la capacité à évaluer d'un point de vue sécuritaire tous les comportements de ces systèmes.

Certaines exigences présentes dans la norme [NF EN 62304] et différentes pistes d'amélioration proposées dans ce rapport peuvent d'ores et déjà s'appliquer. Ces points couvrent notamment les aspects suivants :

- Logiciel modulaire
- Architecture : cloisonnement / mémoire séparée partagée (voir §5.3.2.2)
- Paramétrage (voir §5.3.2.7)
- Interfaces limitées / intégration (voir §5.3.2.5)
- Règles de programmation spécifique pour ce type d'application (voir §5.3.2.3)

4.2.3.2. Utilisation de SOUP/COTS

Dans le domaine médical, l'utilisation des SOUP/COTS est de plus en plus répandue.
Le chapitre 5.3.2.6 identifie une proposition d'amélioration à cet effet.

De plus, outre cette proposition d'amélioration, le choix initial du SOUP est primordial et doit prendre en considération les contraintes de sécurité (Safety) dans lequel le SOUP va être intégré.

En effet, le rapport de gestion des risques doit s'assurer que les SOUP utilisés dans le Logiciel sont appropriés pour le DM concerné et qu'ils respectent les exigences de sécurité fonctionnelles du DM.

4.2.3.3. Intégration des notions de Security

Dans le domaine médical, le besoin de sûreté des DM Logiciels est de plus en plus important du fait de la généralisation de la connectivité (filaire ou non-filaire) des appareils (Internet, réseaux locaux...), ceci pour des besoins comme la mise à jour du Logiciel, la remontée des données, l'interconnexion aux systèmes d'informations ou bien le contrôle à distance du dispositif.

Dans le cadre de Logiciels à contraintes temps-réel, le problème de la sûreté devient d'autant plus important à partir du moment où ces Logiciels contrôlent des dispositifs implantés

garantissant la thérapie ou le maintien en vie des patients (risque immédiat). La connectivité de ces dispositifs introduit des risques réels sur la sûreté des Logiciels qui les contrôlent.

Plus de détails concernant la notion de sûreté (Security) sont présents dans le chapitre 5.3.3.2.

Ainsi il convient de prendre des mesures, non seulement lors du développement du Logiciel mais aussi tout au long de son cycle de vie, pour garantir un niveau minimum de sûreté du Logiciel.

Ces mesures se trouvent dans les chapitres 5.3.3.2.2 et 5.3.3.2.3 sous forme de propositions d'amélioration de la norme [NF EN 62304] pour la prise en compte des aspects de sûreté dans le cycle de vie des Logiciels de Dispositifs Médicaux.

5.DETAIL DES RECOMMANDATIONS ET AXES D'AMELIORATIONS

5.1. CADRE DE L'ANALYSE

Les recommandations et axes d'améliorations sont présentés sous forme de fiches détaillées :

Type	**Recommandation / Complément / Création**	Degré d'importance	**Mineur / Majeur**
		Contribution	1 à 3
Norme [Référence de la Norme] - §			
Constat			
Recommandation ou Amélioration proposée			
Source	[EN 50128] ☐	[ISO 26262] ☐	[CEI 60880] ☐
	Incidents ☐	Autres ☐	
Justification			

- **Type**
  Type concernant la proposition d'amélioration de la norme, 3 types sont définis :
  Recommandation : Elle est destinée aux fabricants de DM pour mettre en application le paragraphe de la norme cité en référence
  Complément : La notion est mentionnée dans la norme en question mais un complément d'information est à ajouter.
  Création : La notion n'est pas mentionnée dans la norme en question.

- **Degré d'importance**
  Degré d'importance lié à la proposition de recommandation/amélioration, 2 degrés sont définis :
  Mineur : la recommandation/amélioration n'est pas considérée avec un degré d'importance fort dans le contexte médical.
  Majeur : la recommandation/amélioration est considérée avec un degré d'importance fort dans le contexte médical.

- **Contribution (Effort / charge) - *uniquement pour les axes d'amélioration***
  Contribution de la part des fabricants de DM pour mettre en place l'amélioration proposée, 3 niveaux sont définis :

1 : La contribution est jugée faible pour la mise en place de cette amélioration car :
- Activité déjà intégrée chez la majeure partie des fabricants de DM
- Charge faible pour réaliser les activités

2 : La contribution est jugée moyenne pour la mise en place de cette amélioration car :
- Activité déjà intégrée chez la majeure partie des fabricants de DM
- Charge forte pour réaliser les activités

OU
- Activité faiblement intégrée chez les fabricants,
- Charge moyenne pour réaliser les activités

3 : La contribution est jugée forte pour la mise en place de cette amélioration car :
- Activité faiblement intégrée chez les fabricants de DM,
- Définition du processus/méthodologie à mettre en place
- Charge forte pour réaliser les activités

- **Constat**

  Pour les recommandations : Constat concernant le non-respect du paragraphe de la norme en question
  Pour les axes d'amélioration : Constat concernant la non-complétude, l'absence de l'exigence ou un manque de la notion dans la norme

- **Recommandation / Amélioration proposée**

  Recommandation pour l'application de la norme en question en se basant sur le retour d'expérience de SERMA INGENIERIE.
  Proposition d'amélioration de la norme en question en se basant sur les normes retenues [EN 50128], [ISO 26262] et [CEI 60880] ainsi que sur le retour d'expérience de SERMA INGENIERIE.

- **Source -** ***pour les recommandations, seul le champ « Incident » est complété***

  Identification des sources à l'origine ou utilisées pour réaliser la proposition d'amélioration.
  Lorsque les normes sont citées, le chapitre (ou les chapitres associés) est identifié.
  Pour les incidents, le nombre d'incidents ayant pour cause potentielle primaire l'amélioration proposée est indiqué entre parenthèse le cas échéant.
  Si une source autre que les normes retenues et les incidents a été utilisée, elle sera citée dans la partie « Autres ».

- **Justification**

  Pour les recommandations : Justification permettant d'expliquer le degré d'importance et les incidents retenus.
  Pour les axes d'amélioration : Justification permettant d'expliquer la plus-value apportée par cette amélioration ainsi que les niveaux de degré d'importance et de contribution choisis. Cette justification pourra s'appuyer sur les sources (Normes retenues, incidents, …).

5.2. RECOMMANDATIONS POUR LES FABRICANTS

5.2.1. BASE DE DONNEES

Type	**Recommandation**	Degré d'importance	**Majeur**
Norme [NF EN 62304] § 5.2.2 g) Exigences en matière de base de données et de définitions des données			
Constat	Les Dispositifs Médicaux ont souvent besoin de gérer une grande quantité d'informations : données patient, données médicales, données techniques, … Certains sont même dédiés à cela. Ils gèrent des données à destination (données patient) ou en provenance (surveillance d'activité) d'autres dispositifs médicaux. Pour cela, ces DM s'appuient sur une base de données et un Système de Gestion de Base de Données (SGBD). La norme [NF EN 62304], au § 5.2.2 g), demande aux fabricants d'établir des exigences en matière de bases de données. Quelques recommandations peuvent être apportées sur ce sujet.		
Recommandation	Il existe des notions de base, reconnues, qui permettent de garantir une bonne conception d'une base de données. Les exigences du Logiciel peuvent faire apparaître tout ou partie de ces notions, parmi lesquelles figurent : - La pérennité des données : les données ne doivent pas être perdues. - L'intégrité des données : les valeurs doivent être dans les limites admises, dans un format correct et cohérentes avec ce qui existe déjà dans la base. - L'unicité des données : les données ne doivent pas être redondantes. - La confidentialité des données : aucune personne non autorisée ne doit avoir accès à des données confidentielles. Des règles de bonne pratique et l'utilisation d'un SGBD reconnu sont à définir afin de mettre en application les notions ci-dessus. Cela concerne : - La spécification du type de données, la structure des données, … - La définition des règles de cohérence (cohérence des dates, pas d'informations redondantes, …) - La définition d'un langage de commande afin de manipuler la Base de Données (traitement des requêtes).		
Source	Incidents	2	
Justification	Deux incidents analysés en PT1 sont dus à une mauvaise gestion de base de données. L'un provient d'une redondance de données dans la base, l'autre d'une absence de mise à jour de la base suite à une action utilisateur (ces bases de données sont internes au dispositif médical). Le degré d'importance de cette recommandation est majeur au vu de la criticité et de la confidentialité des données gérées par les dispositifs médicaux.		

5.2.2. VERIFICATION SPECIFICATION / CONCEPTION ARCHITECTURALE

Type	**Recommandation**	Degré d'importance	**Majeur**
Norme [NF EN 62304] § 5.2.6 : Vérification des exigences du Logiciel § 5.3.6 : Vérification de l'architecture du Logiciel			
Constat	Les exigences du § 5.2.6 et 5.3.6 de la norme [NF EN 62304] demandent de vérifier les activités de : - définition des exigences du Logiciel (Spécification), - conception architecturale. Plusieurs recommandations peuvent être apportées à ces exigences afin de guider les fabricants de Logiciels de Dispositifs Médicaux.		
Recommandation	Les documents de ces 2 activités doivent faire l'objet d'une relecture afin de valider leur adéquation avec le ou les documents en entrée de l'activité. Plusieurs axes de relecture sont possibles, aussi bien sur le fond que sur la forme des documents. Les vérifications concernent : - La complétude des exigences. - La cohérence (non-contradiction) des exigences. - La pertinence des exigences. - La traçabilité des exigences avec les exigences amont. - La testabilité des exigences. - La maintenabilité du document. - La lisibilité du document. Cette relecture est à formaliser dans un document (Exemple : fiche de relecture) permettant ainsi d'assurer le suivi des remarques. Le traitement des points est le suivant : - Remarques émises par les relecteurs - Prises en compte des remarques ou justifications apportées par le rédacteur du document (selon l'acceptation ou le refus des remarques) - Vérification de la prise en compte et des justifications pour clôture des remarques par les relecteurs Ce traitement doit se répéter jusqu'à ce que les remarques soient clôturées ou qu'un accord ait été trouvé pour traiter les remarques dans une version ultérieure.		
Source	Incidents	7	
Justification	Sept incidents analysés en PT1 sont dus à des problèmes de spécification ou de conception architecturale qui auraient pu être évités avec une bonne vérification : exigences manquantes, cas particuliers non pris en compte, … Le degré d'importance de cette recommandation est majeur. Une bonne vérification de la spécification et de l'architecture est indispensable car toute erreur à ce niveau peut se propager sur tout le cycle de développement du Logiciel et peut ne pas être détectée par les tests.		

5.2.3. ANALYSE D'IMPACT D'UNE MODIFICATION

Type	**Recommandation**	Degré d'importance	**Majeur**
Norme [NF EN 62304] § 5.7.3 : Contre-essais après modifications § 7.4.2 : Analyse de l'impact des modifications apportées au Logiciel sur les mesures existantes de maîtrise du risque § 9.7 : Vérification de la résolution des problèmes du Logiciel			
Constat	La modification d'un Logiciel peut avoir un impact sur des parties du Logiciel qui ne sont pas concernées directement par la modification. C'est pourquoi, il est indispensable de mesurer cet impact, de l'évaluer d'un point de vue « Risque » et d'effectuer les tests appropriés afin de valider le Logiciel modifié. La norme [NF EN 62304] émet plusieurs exigences en ce sens : - Le § 5.7.3 parle des modifications qui ont lieu pendant la phase d'essais du Logiciel. Le fabricant doit alors effectuer des « essais appropriés » afin de démontrer que des « effets secondaires non prévus » n'ont pas été introduits. - Au § 7.4.2, le fabricant doit déterminer si la modification du Logiciel peut « interférer » avec les mesures existantes de maîtrise du risque. - Le § 9.7 demande au fabricant de vérifier si des « problèmes supplémentaires » n'ont pas été introduits suite à la mise en œuvre d'une modification. Par ailleurs, le § 5.6.6 demande de réaliser, pendant la phase d'intégration du Logiciel, un « essai de régression approprié » pour démontrer que des défauts n'ont pas été introduits dans le Logiciel précédemment intégré.		
Recommandation	Quelle que soit la phase du cycle de vie du Logiciel (développement, exploitation, maintenance), toute modification du Logiciel doit faire l'objet d'une analyse d'impact qui doit permettre : - De s'assurer que des causes potentielles supplémentaires ne sont pas introduites. - D'identifier des Mesures de Maîtrises de Risques supplémentaires si nécessaire. - De s'assurer que les Mesures de Maîtrises de Risques existantes sont toujours traitées. - D'identifier tous les Eléments Logiciels (EL) potentiellement touchés par cette modification, en particulier les EL « critiques » (c'est-à-dire pouvant contribuer à une situation dangereuse). - D'identifier les essais qui doivent être « rejoués » (non-régression). - De réaliser de nouveaux tests spécifiques liés à cette modification. Le travail d'analyse d'impact est facilité par l'existence d'une conception architecturale et d'une conception détaillée riches qui permettent de connaître les interactions entre les Eléments Logiciels et les interactions entre les Unités Logicielles.		

Source	Incident	1
Justification	Un incident analysé en PT1 est dû à un défaut introduit lors de la correction d'un autre défaut. L'impact de la modification liée au premier défaut n'a pas été correctement déterminé. Le degré d'importance de cette recommandation est majeur au vu de la criticité des Logiciels de dispositifs médicaux et de la complexité croissante des Logiciels qui introduit des effets de bord, parfois difficilement identifiables, lors de modifications.	

5.2.4. GESTION ET SUIVI DES RISQUES

Type	**Recommandation**	Degré d'importance	**Majeur**
Norme [NF EN 62304] § 7 : Processus de gestion des risques du Logiciel Norme [ISO 14971]			
Constat	Certaines anomalies en relation avec la spécification ou la conception architecturale peuvent être le résultat d'une analyse de risque qui n'a pas su : - Identifier correctement les situations dangereuses - Évaluer les risques associés - Mettre en place des Mesures de Maîtrise de Risque (MMR) - Vérifier la mise en œuvre de ces MMR Des recommandations peuvent être apportées au § 7 de la norme [NF EN 62304] et la norme [ISO 14971].		
Recommandation	<u>Réalisation d'analyses de risques :</u> Plusieurs méthodes d'analyse dysfonctionnelle existent afin d'identifier les risques liés aux DM et à ses Logiciels. Certaines de ces méthodes sont présentées dans l'annexe G de la norme [ISO 14971] et sont à réaliser par le fabricant. Cela concerne par exemple : - L'APR : Le fabricant, au démarrage du processus de développement, peut effectuer une Analyse Préliminaire de Risque (APR) afin d'identifier les phénomènes dangereux, les situations dangereuses et les événements pouvant entraîner des dommages (pour plus de détail voir le G.2 de la norme [ISO 14971] - L'AMDEC : Suite au découpage en bloc fonctionnel du DM, le fabricant peut effectuer une Analyse des Modes de Défaillance, de leurs Effets et de leur Criticité (AMDEC). Cette analyse, basée sur la défaillance des		

	blocs fonctionnels, permet d'identifier l'atteinte ou non des situations dangereuses et d'en déduire des Mesures de Maîtrises de Risques appropriées pour sécuriser la défaillance les différents blocs du DM. Nous recommandons d'appliquer la méthodologie d'AMDEC au Logiciel et de réaliser une AMDEC du Logiciel. Cette analyse permet de s'assurer de : - La non-atteinte des situations dangereuses en cas de défaillance du Logiciel. - La suffisance des protections du Logiciel. Lors de cette analyse, si les situations dangereuses sont atteintes et que les protections sont jugées insuffisantes, cette analyse permet d'identifier de nouvelles protections du Logiciel sous la forme de MMR. Pour le Logiciel (identifié dans l'AMDEC) l'analyse peut être effectuée au niveau des fonctionnalités (spécification du Logiciel), au niveau des Eléments Logiciels (Conception architecturale) ou de plus bas niveau. Pour information, cette technique d'AMDEC du Logiciel est notamment utilisée dans le domaine automobile et dans le domaine ferroviaire sous le nom d'AEEL. <u>Suivi des Mesures de Maîtrise de Risque :</u> Pour le traitement des MMR, la mise en place par le fabricant d'un document de suivi des MMR liées aux situations dangereuses est recommandée. Ce document est initié suite aux analyses de risque en amont du développement Logiciel. Il est mis à jour tout au long du cycle de développement du Logiciel et s'enrichit à chaque activité. Les MMR sont ainsi suivies et tracées de la spécification jusqu'aux tests de validation. Par ailleurs, il est recommandé d'utiliser le guide d'application de la norme ISO 14971 [TR 80002-1].	
Source	Incident	1
Justification	Un incident est dû à une mauvaise identification des éléments Logiciels pouvant contribuer à une situation dangereuse. Le degré d'importance de cette recommandation est majeur. Il est primordial dans le domaine médical de savoir évaluer les risques inhérents à un Logiciel, de mettre en place et de vérifier les Mesures de Maîtrise des Risques.	

5.2.5. SYNTHESE DES RECOMMANDATIONS

Le tableau ci-dessous présente une synthèse de ces recommandations :

Recommandation	Degré d'importance	Nombre incidents (1)
[NF EN 62304]		
Base de données	Majeur	2
Vérification spécification / conception architecturale	Majeur	7
Analyse d'impact d'une modification	Majeur	1
[NF EN 62304] et [ISO 14971]		
Gestion et suivi des risques	Majeur	1

(1) : Le « Nombre d'incidents » correspond aux incidents identifiés issus des déclarations de matériovigilance analysées lors de la première partie de l'étude étant en relation avec les recommandations.

5.3. AXES D'AMELIORATION POUR LA NORME [NF EN 62304]

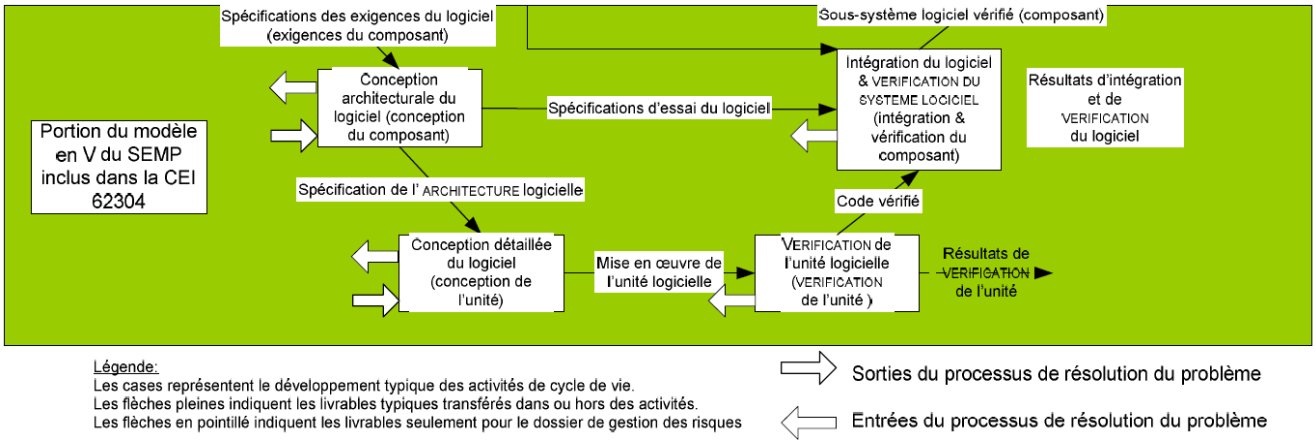
5.3.1. AXE D'AMELIORATION LIE AUX PROCESSUS

5.3.1.1. Introduction aux Processus du cycle de vie

La norme [NF EN 62304] identifie plusieurs Processus dans le cycle de vie du Logiciel :

- Le Processus de développement du Logiciel,
- Le Processus de maintenance du Logiciel,
- Le Processus de gestion des risques liés au Logiciel,
- Le Processus de gestion de la configuration du Logiciel,
- Le Processus de résolution des problèmes du Logiciel.

Afin de mieux appréhender les améliorations proposées, il convient d'introduire la notion d'activités (également appelé phase de développement) pour la partie Processus de Développement Logiciel (§5 de la norme) en intégrant la partie liée au Logiciel de la figure C.2 de la norme [NF EN 62304]. Cette figure est présente dans la norme dans une partie informative :



Afin de synthétiser cette figure, les activités de développement du Logiciel identifiées dans la norme [NF EN 62304] au §5, outre les phases de planification du Logiciel, sont :

- Analyses des exigences du Logiciel (Spécification des Exigences du Logiciel)
- Conception architecturale du Logiciel (Eléments Logiciels)
- Conception détaillée (Unités Logicielles)
- Mise en œuvre des Unités Logicielles (Codage)
- Vérification des Unités Logicielles
- Intégration et Essai d'Intégration du Logiciel
- Essais du système Logiciel (Tests de Validation du Logiciel)

La norme [NF EN 62304] référence la norme [ISO 14971] concernant le Processus de Gestion des Risques (pour plus de détail voir le chapitre 5.3.3).

En introduction de la norme [NF EN 62304], il est indiqué que le processus de maintenance est aussi important que le processus de développement (pour plus de détail voir le chapitre 5.3.1.5).

Les Processus de gestion de la configuration du Logiciel et de résolution des problèmes Logiciel font l'objet d'un chapitre spécifique dans la norme [NF EN 62304] et ont été jugés, dans le cadre de cette étude, suffisants.

5.3.1.2.Précisions sur le cycle de vie

En introduction de la norme [NF EN 62304], aucun modèle de cycle de vie spécifique n'est imposé. Néanmoins, la norme indique que le cycle de vie de développement du Logiciel doit être défini (§5.1.1 de la norme) et précise que différentes stratégies de développement (cascade, incrémentielle, …) peuvent être utilisées (annexe B.1.1 de la norme).
Cette approche nous semble cohérente et adaptée aux différents types de Logiciel que peuvent rencontrer les fabricants de DM Logiciel.
Ces informations ont été jugées suffisantes et ne sont pas renseignées dans les propositions d'amélioration.

5.3.1.3.Organisation

5.3.1.3.1.Structure organisationnelle

Type	**Création**	Degré d'importance	**Majeur**
		Contribution	2
Norme [NF EN 62304] (création d'un chapitre dans le §4 : Exigences générales)			
Constat	Comme indiqué en introduction de la norme [NF EN 62304], aucune prescription concernant la structure organisationnelle n'est abordée dans la norme. Les notions d'organisation, rôles, responsabilités, compétences et indépendance des équipes ou intervenants ne sont pas présentes.		
Amélioration proposée	La structure organisationnelle doit être définie en fonction de la classe du Logiciel en précisant le rôle et les responsabilités des différents intervenants. A titre d'exemple et en s'appuyant sur la norme [EN 50128], les intervenants peuvent être : - Chef de Projet - Gestionnaires des Exigences / Concepteur / Réalisateur (Développement) - Chargé d'Intégration / Chargé de Validation (Tests) - Vérificateur - Chargé des activités liées aux Risques L'annexe B de la norme [EN 50128] décrit les responsabilités et principales compétences des intervenants. Exemple d'une des responsabilités d'un Concepteur : doit transformer les exigences spécifiées relatives au Logiciel en solutions acceptables		

	L'indépendance entre les équipes doit être définie en fonction de la classe du Logiciel (schéma de la structure organisationnelle en fonction des classes). Pour la classe C, une indépendance entre l'équipe de développement et l'équipe de tests est nécessaire. De même que pour l'équipe en charge de la gestion des Risques. La compétence des intervenants est à définir dans la norme. Le fabricant devra fournir la preuve documentée de la compétence du personnel (connaissances techniques, expérience et formation appropriée). Il est nécessaire pour chaque intervenant de démontrer un maintien et un développement continus des compétences. Cette partie est abordée dans le §6.2 de la norme [ISO 13485].		
Source	[EN 50128] ☒ § 5.1, 5.2, Annexe B	[ISO 26262] ☒ Partie 2	[CEI 60880] ☐
	Incidents ☐	Autres ☐	
Justification	La définition de l'organisation, des rôles et responsabilités a pour objectif d'assurer que l'ensemble du personnel responsable du Logiciel est organisé, capable d'assumer ses responsabilités et a les compétences pour s'acquitter de ces responsabilités. Pour le dernier point, le fabricant devra démontrer la capacité d'exécuter les tâches correspondantes de manière correcte, efficace et cohérente à un niveau de qualité élevé (Source [EN 50128]). De ce fait, le degré d'importance est majeur. La contribution pour le fabricant est moyenne car l'organisation indépendante nécessite plusieurs intervenants.		

5.3.1.3.2.Evaluation / Certification

Type	**Création**	Degré d'importance	**Majeur**
		Contribution	2
Norme [NF EN 62304] (création d'un chapitre dédié)			
Constat	Aucune exigence n'est applicable à l'évaluateur/certificateur concernant le processus du cycle de vie du Logiciel dans la norme [NF EN 62304].		
Amélioration proposée	Un processus d'évaluation doit être ajouté dans la norme. L'évaluateur/certificateur doit réaliser un Plan d'Evaluation Logiciel qui définit notamment : - La stratégie et méthodologie d'évaluation (audit des activités et des intervenants, évaluation documentaire, évaluation des tests, …) - Les documents à évaluer - Les exigences relatives au contenu du Rapport d'Evaluation Logiciel		

<table>
<tr><td></td><td colspan="3">L’évaluateur/certificateur doit également réaliser un Rapport d’Evaluation Logiciel, qui doit statuer sur la base du rapport de gestion des risques :
- sur la conformité à la norme [NF EN 62304] en fonction de la classe du Logiciel
- sur les mesures de maîtrise des risques (MMR)
- sur les documents produits
- sur les anomalies identifiées
Il doit notamment citer la version évaluée, les restrictions et contraintes d’utilisation et est valable pour une version spécifique.

L’activité doit être menée par une entité indépendante et est fonction de la classe du Logiciel.

L’évaluateur/certificateur doit être impliqué à un stade précoce du projet.</td></tr>
<tr><td rowspan="2">Source</td><td>[EN 50128] §6.4 ☒</td><td>[ISO 26262] ☐</td><td>[CEI 60880] ☐</td></tr>
<tr><td>Incidents ☐</td><td colspan="2">Autres ☐</td></tr>
<tr><td>Justification</td><td colspan="3">Cette activité a pour objectif d’évaluer que les processus du cycle de développement ainsi que leurs sorties sont tels que le Logiciel est conforme à la classe de sécurité du Logiciel et qu’il est adapté à son application prévue.
La description du processus d’évaluation permettra de donner une ligne directrice aux évaluateurs/certificateurs et d’indiquer aux fabricants les points qui seront évalués.
La contribution pour le fabricant est moyenne car cela impose des contraintes aux évaluateurs/certificateurs et peut engendrer des démonstrations supplémentaires à mettre en place par le fabricant mais cela donnera un cadre concernant les activités à mener par l’évaluateur/certificateur.</td></tr>
</table>

5.3.1.4.Méthodes

5.3.1.4.1.Traçabilité globale

Type	**Complément**	Degré d'importance	**Majeur**
		Contribution	1
Norme [NF EN 62304] § 5.1.1 – Plan de développement du Logiciel			
Constat	Seule la traçabilité entre les exigences du Système, les exigences du Logiciel, les essais du Logiciel (Test de Validation) et les MMR mise en œuvre dans le Logiciel est à traiter.		
Amélioration proposée	Une traçabilité entre les différentes activités (intégrant les MMR) devrait être mise en place, via : - Une traçabilité descendante et montante de la Spécification des Exigences du Logiciel jusqu'au code : ✓ Spécification des Exigences <-> Conception Architecturale ✓ Conception Architecturale <-> Conception Détaillée ✓ Conception Détaillée <-> Code - Une traçabilité horizontale afin mettre en relation les tests du Logiciel avec les documents de développement : ✓ Conception Détaillée <-> Vérification de UL ✓ Conception Architecturale <-> Intégration du Logiciel ✓ Spécification des Exigences <-> Essais du Logiciel Les traçabilités doivent être vérifiées.		
Source	[EN 50128] ☒ §5.3.2.7, 6.5.4.14, 6.5.4.15	[ISO 26262-6] ☒ §7.4.2, 8.4.5	[CEI 60880] ☐
	Incidents ☒ (3)	Autres ☐	
Justification	La traçabilité a pour but de démontrer de bout-en-bout le suivi de toutes les exigences du Logiciel (y compris les exigences liées au MMR) et donc une maîtrise du développement et des activités de tests. La traçabilité a également comme intérêt de faciliter la maintenabilité. La contribution pour le fabricant est jugée faible car cela ne nécessite pas un investissement particulier et est aujourd'hui réalisé chez la majeure partie des fabricants. Par ailleurs, l'analyse des incidents a identifié 3 cas en lien avec un manque de traçabilité.		

5.3.1.5.Documentation

5.3.1.5.1.Documents à produire

Type	**Création**	Degré d'importance	**Mineur**
		Contribution	1
Norme [NF EN 62304] (création d'une annexe)			
Constat	La norme [NF EN 62304] n'identifie pas de façon claire et synthétique les documents à fournir dans le cadre d'un développement Logiciel. Par ailleurs, les points d'entrées documentaires et les livrables à produire pour chaque activité de développement (Ex : conception architecturale) ne sont pas identifiés dans la norme.		
Amélioration proposée	Un tableau identifiant les documents à produire par activités de développement en fonction de la classe du Logiciel est à ajouter. Ces tableaux sont présents dans les normes [EN 50128] et [ISO 26262-6]. De plus, pour chaque activité dans le corps de la norme, identifier dans la norme les points d'entrée et les points de sorties.		
Source	[EN 50128] ☒ Annexe A.1	[ISO 26262-6] ☒ Annexe A	[CEI 60880] ☐
	Incidents ☐	Autres ☐	
Justification	Les informations concernant les documents à produire pour chaque activité en fonction de la classe du Logiciel permettent de clarifier les documents à produire de la part du fabricant. La contribution pour le fabricant est faible car cela n'ajoute pas de charge supplémentaire.		

5.3.1.5.2.Contraintes de performance et d'environnement

<table>
<tr><td rowspan="2">Type</td><td rowspan="2">Complément</td><td>Degré d'importance</td><td>Majeur</td></tr>
<tr><td>Contribution</td><td>2</td></tr>
<tr><td colspan="4">Norme [NF EN 62304]
§ 5.2.2 – a) : Exigences en termes de fonctionnalité et de capacité</td></tr>
<tr><td>Constat</td><td colspan="3">L'exigence du §5.2.2 –a) de la norme [NF EN 62304], indique que des exigences en termes de fonctionnalité et de capacité doivent être définies.
La note 1 identifie des informations à ce sujet mais ceux-ci ne sont que des « exemples ».</td></tr>
<tr><td>Amélioration proposée</td><td colspan="3">La note 1 devrait être intégrée dans l'exigence du §5.2.2 – a) afin d'identifier clairement les contraintes à renseigner par le fabricant.
Cela concerne :
- Les contraintes de dimensionnement et de performance : temps de réponse, synchronisation, …
- Les caractéristiques physiques : langage, système d'exploitation, …
- L'environnement informatique dans lequel le Logiciel doit fonctionner : contraintes de place mémoire, infrastructure du réseau, CPU, taille disque dur, …

Ces contraintes doivent être couvertes par des vérifications/tests chez le fabricant et pour certaines, contrôlées lors de l'installation (voir amélioration liée à l'installation).
Des techniques de tests sont listées dans les normes [EN 50128] (Tableau A.18) et [ISO26262-6] (Tableau 13 et 16).</td></tr>
<tr><td rowspan="2">Source</td><td>[EN 50128] ☒
§6.2.4.7, §7.2.4.2, §7.2.4.19, Tableau A.18</td><td>[ISO 26262] ☒
§11, Tableau 13, 16</td><td>[CEI 60880] ☒
Annexe B2</td></tr>
<tr><td>Incidents ☒ (4)</td><td colspan="2">Autres ☐</td></tr>
<tr><td>Justification</td><td colspan="3">La définition de ces contraintes a pour rôle d'éviter l'apparition d'incidents pouvant avoir comme cause primaire :
- Un non-respect des performances (mémoire, CPU, taille disque dur, …).
- Une non-maîtrise de l'environnement - Logiciel non fonctionnel lié à une cause externe au Logiciel (surcharge réseau, problème d'interopérabilité de Logiciels, processeur sous-dimensionné, …)
4 incidents ont pour cause ces problématiques.
La contribution pour le fabricant est jugée moyenne car les contraintes sont à formaliser puis à valider chez le fabricant et lors de l'installation.</td></tr>
</table>

5.3.1.5.3.Installation

<table>
<tr><td rowspan="2">Type</td><td rowspan="2">Complément</td><td>Degré d'importance</td><td>Majeur</td></tr>
<tr><td>Contribution</td><td>1</td></tr>
<tr><td colspan="4">Norme [NF EN 62304]
§ 5.2.2 – h) : Les exigences d'installation et d'acceptation du Logiciel de DISPOSITIF MÉDICAL livré au(x) site(s) d'exploitation et de maintenance</td></tr>
<tr><td>Constat</td><td colspan="3">La norme [NF EN 62304] contient peu d'exigences liées à l'installation et à sa vérification.
Seule l'exigence au § 5.2.2 – h) indique que le fabricant doit inclure les exigences d'installation et d'acceptation du Logiciel de DM livré au(x) site(s) d'exploitation et de maintenance.</td></tr>
<tr><td>Amélioration proposée</td><td colspan="3">La norme [NF EN 62304] doit intégrer des exigences afin de contrôler l'installation sur site.
Des tests spécifiques doivent être réalisés pour s'assurer que le Logiciel est installé correctement en prenant en considération l'environnement d'exécution, les performances requises et en démontrant un respect des fonctionnalités.
Ces tests doivent être formalisés.</td></tr>
<tr><td rowspan="2">Source</td><td>[EN 50128] ☐</td><td>[ISO 26262] ☒
Tableau 16</td><td>[CEI 60880] ☒
§ 12</td></tr>
<tr><td>Incidents ☒ (4)</td><td colspan="2">Autres ☐</td></tr>
<tr><td>Justification</td><td colspan="3">La formalisation des activités d'installation permettra de s'assurer de la bonne installation d'un point de vue fonctionnel, de performance et par rapport à l'environnement d'exécution.

4 incidents proviennent d'une mauvaise installation.

La contribution pour le fabricant est jugée faible car, en dehors de la formalisation, ses activités sont déjà réalisées chez la majeure partie des fabricants.</td></tr>
</table>

5.3.1.5.4.Conception détaillée

<table>
<tr><td rowspan="2">Type</td><td rowspan="2">Complément</td><td>Degré d'importance</td><td>Majeur</td></tr>
<tr><td>Contribution</td><td>3</td></tr>
<tr><td colspan="4">Norme [NF EN 62304]
§ 5.4 : Conception détaillée du Logiciel</td></tr>
<tr><td>Constat</td><td colspan="3">Les exigences du §5.4 de la norme [NF EN 62304] demande l'élaboration et la documentation de chaque Unité Logicielle (UL) avec une documentation des interfaces entre les UL.
Néanmoins, aucune contrainte spécifique sur la description des UL et de ces interfaces n'est présente dans la partie normative.
Le §B.5.4 en annexe informative B apporte des informations complémentaires.</td></tr>
<tr><td>Amélioration proposée</td><td colspan="3">Les informations complémentaires du §B.5.4 doivent être intégrées au § 5.4 dans la partie normative, notamment la partie suivante :
« La conception détaillée spécifie des algorithmes, des représentations des données, des interfaces entre les différentes UNITÉS LOGICIELLES et les interfaces entre les UNITÉS LOGICIELLES et les structures de données. »
Afin d'être complet, les informations permettant de décrire une UL sont les suivantes :
- Définition des données en Entrées/Sorties des UL (paramètres, variables globales, retour,…)
- Définition et description des domaines de définitions des E/S (classes d'équivalences des données)
- Définition des valeurs aux limites des données et le comportement en cas de dépassement
- Description détaillée des algorithmes
- Interface d'appel entre les UL (appel de fonction)

Par ailleurs, la norme [EN 50128] intègre le fait que la taille et la complexité de chaque composant (UL) doivent être équilibrées.</td></tr>
<tr><td rowspan="2">Source</td><td>[EN 50128] ☒
§7.4.4.1 à 7.4.4.5</td><td>[ISO 26262-6] ☒
§8.4.4</td><td>[CEI 60880] ☐</td></tr>
<tr><td>Incidents ☒ (12)</td><td colspan="2">Autres ☐</td></tr>
<tr><td>Justification</td><td colspan="3">La mise en place de ces exigences permettra de répondre à la partie suivante du §B.5.4 de la norme :
« La conception détaillée renseigne les détails nécessaires à la construction du Logiciel. Il convient qu'elle soit suffisamment complète pour que le programmateur n'ait pas à prendre des décisions de conception circonstancielles. »</td></tr>
</table>

	Par ailleurs, 12 incidents ont pour cause primaire potentielle le manque de description de la conception détaillée. Il faut ajouter à cela que la conception détaillée est également le point d'entrée pour l'activité de vérification des UL et qu'un manque d'information ne permettra pas d'identifier des bugs sur des chemins d'exécution particuliers. La contribution pour le fabricant est forte car un effort de description des UL est à réaliser puis dans un second temps à tester lors de la vérification des Unités Logicielles.

5.3.1.6. Maintenance

Aucune amélioration spécifique n'est proposée.

5.3.2. AXE D'AMELIORATION LIE AUX TECHNIQUES DE DEVELOPPEMENT

La norme [NF EN 62304] contient peu de contraintes liées aux techniques de développement, concernant les :

- Contraintes architecturales
- Contraintes d'implémentation
- Techniques de tests
- Contraintes d'interfaces et d'intégration HW/SW
- Contraintes liées aux données d'application (paramétrage)

Ces contraintes font l'objet des différentes propositions d'amélioration présentes dans ce chapitre qui sont à adapter selon la classe de sécurité du Logiciel.

Néanmoins, la norme [NF EN 62304] utilise les notions d'Elément Logiciel dans l'activité de conception architecturale et d'Unité Logicielle dans l'activité de conception détaillée.
Ces notions contribuent à réaliser un développement Logiciel modulaire et structuré. Cela est un point positif dans le développement de Logiciel critique.

5.3.2.1. Techniques de développement générales

Type	**Création**	Degré d'importance	**Majeur**
		Contribution	2
Norme [NF EN 62304] (création d'une annexe)			
Constat	La norme [NF EN 62304] n'indique aucune prescription sur les techniques de développement à réaliser.		
Amélioration proposée	La norme [EN 50128] possède en annexe A (normative), des tableaux (A1 à A23) qui indique de manière détaillée les techniques à mettre en œuvre pour réaliser les activités de développement en fonction du niveau de criticité du Logiciel. Pour chaque technique un lien vers une explication (objectif et description) est présent. La norme [ISO26262-6] possède également cette même approche présente dans les tableaux 1 à 16. Ce type de tableau doit être intégré dans la norme [NF EN 62304] en adaptant leur contenu au domaine médical. Les techniques à appliquer doivent être fonction de la classe du Logiciel, en instaurant plusieurs niveaux : - Obligatoire - Hautement Recommandé - Recommandé - Aucune recommandation pour ou contre son utilisation - Non Recommandé. Voici à titre d'exemple quelques techniques Obligatoires ou Hautement Recommandées extraites de la norme [EN 50128] pour l'activité d'Architecture du Logiciel (liste non exhaustive) : - Programmation défensive		

	- Détection de défauts & diagnostic - Programmation par assertion - Programmation diversifiée - Interface entièrement définie - Méthodologie structurée Voici à titre d'exemple quelques techniques extraites de la norme [EN 50128] pour l'activité de Conception et mise en œuvre du Logiciel (liste non exhaustive) : - Méthodologie structurée - Approche modulaire - Règles de conception et de codage - Programmation structurée - Sous-ensemble de langage		
Source	[EN 50128] ☒ Tableaux A.1 à A.23 (Annexe A)	[ISO 26262-6] ☒ Tableaux 1 à 16	[CEI 60880] ☐
	Incidents ☐	Autres ☐	
Justification	L'utilisation de ce type de tableaux permettrait d'illustrer les moyens d'assurer la conformité aux exigences de la norme. Par ailleurs, une annexe (normative) dédiée intégrant ces tableaux serait un atout pour les fabricants pour avoir des directives précises sur la réalisation du Logiciel. De ce fait, le degré d'importance est jugé majeur. La contribution pour le fabricant est moyenne de par l'application des techniques définies.		

5.3.2.2.Contraintes architecturales

Type	**Complément**	Degré d'importance	**Majeur**
		Contribution	2
Norme [NF EN 62304] § 5.3 : Conception architecturale du Logiciel			
Constat	Très peu de contraintes architecturales sont présentes dans la norme [NF EN 62304]. Au §5.3.5, une contrainte est présente pour un Logiciel Classe C : Les séparations entre Eléments Logiciels essentiels pour la maîtrise du risque doivent être identifiées ainsi que la méthode permettant de s'assurer que la séparation est efficace (Classe C – §5.3.5).		

<table>
<tr><td>Amélioration proposée</td><td colspan="3">Des contraintes architecturales pour les développements des Logiciels Classe C doivent être ajoutées.
De nombreuses notions sont présentes dans la norme [ISO26262] et sont à intégrer dans la norme [NF EN 62304] :

Une description détaillée de l’architecture de sécurité et de la stratégie de tolérances aux fautes (liée à la criticité, au matériel, aux contraintes d’architecture Système, aux exigences de Sécurité, ...) est à mettre en place ([ISO 26262–5] Annexe D).
Afin d’assurer la robustesse de l’architecture du Logiciel plusieurs approches sont possibles :
- Redondance. La mise en place de cette technique permet d’améliorer la sécurité et/ou la disponibilité du Logiciel.
- Ségrégation, cloisonnement et cohabitation des constituants Logiciels de niveaux de criticité différents (§5 et §6 de la norme [ISO 26262-9] pour la décomposition en ASIL d’un système et les critères pour la coexistence d’éléments d’ASIL différent)
- Programmation diversifiée
- Interfaces limitées entre les Logiciels de sécurité (Safety) et les Logiciels non relatifs à la sécurité
- Interfaces limitées entre les Eléments Logiciels de sécurité (Safety) et Eléments Logiciels non relatifs à la sécurité
- Mécanisme de détection des erreurs :
✓ Contrôle du séquencement des tâches
✓ Contrôle temporel (ex : compteur de cycle) : [ISO 26262-6] - D.2.2
✓ Contrôle de vraisemblance/limite des données (ex : assertion, comparaison) : [ISO 26262-6] - D.2.4
✓ Contrôle d’intégrité des données (ex : pointeurs nuls, CRC, checksum) : [ISO 26262-6] - D.2.4
✓ Contrôle de la mémoire (RAM, MMU, MPU) : [ISO 26262-6] - D.2.3
✓ Autotests (Watchdog, comparaison entre voies, HW, …) et protocole de communication
- Mécanisme de gestion des erreurs :
✓ Fonctions de mise dans un état sûr du système et de modes dégradés/position repli
✓ Fonctions de suivi et de diagnostic des erreurs détectées en exploitation :
• Gestionnaire d’alarmes,
• Gestionnaire d’historiques,
• Tampons de fautes.
•</td></tr>
<tr><td rowspan="2">Source</td><td>[EN 50128] ☒
Tableau A.3, Annexe D</td><td>[ISO 26262] ☒
Partie 5 : Annexe D
Partie 6 : §D.2.2, D.2.3, D.2.4
Partie 9 : §5, 6</td><td>[CEI 60880] ☒
Annexes B2, B3</td></tr>
<tr><td>Incidents ☒ (5)</td><td colspan="2">Autres ☐</td></tr>
</table>

Justification	Les contraintes architecturales concernant les stratégies de tolérance aux fautes et protections permettent de garantir la sécurité des Logiciels ayant une contrainte temps réel (déterminisme, position de repli, valeur de sortie cohérente, …). Ce point est primordial pour l'exécution du Logiciel dans un état sûr. Par ailleurs, 5 incidents ont pour cause primaire un manque de robustesse dans l'architecture du Logiciel. La contribution pour le fabricant est moyenne de par : - la mise en place des architectures robustes et des protections dans le Logiciel - la description des choix architecturaux dans la documentation

5.3.2.3. Contraintes d'implémentation et vérifications UL

Type	**Complément**	Degré d'importance	**Majeur**
		Contribution	2
Norme [NF EN 62304] § 5.5 : Mise en œuvre et vérification des UNITÉS LOGICIELLES			
Constat	Des critères d'acceptation de l'Unité Logicielle sont décrits au §5.5. en fonction de la classe du Logiciel. Ces critères manquent de précisions. Une note au §5.5.3 donne des exemples : – le code du Logiciel met-il correctement en œuvre les exigences, y compris les MMR ? – le code du Logiciel est-il en contradiction avec les interfaces décrites dans les documents de conception détaillée de l'unité LOGICIELLE ? – le code du Logiciel est-il conforme aux procédures de programmation ou normes de codage établies ?		
Amélioration proposée	Cette note doit être mise en exigence car la cohérence du code avec les exigences (dont MMR), la conception détaillée et les règles de programmation est nécessaire pour le développement de DM. De la même manière, une exigence doit être créée afin d'indiquer que le code doit être établi en suivant des règles de conception et de programmation définies par le fabricant. Les normes [EN 50128] et [ISO 26262-6] identifient cette contrainte et listent de façon non-exhaustive des règles de conception et de programmation à respecter ainsi que les langages de programmation qui peuvent être utilisées en fonction du niveau d'intégrité de sécurité du Logiciel (SSIL). La norme [EN 50128] interdit par exemple l'utilisation de branchements inconditionnels.		

	La norme [ISO 26262-6] met en avant dans un exemple le référentiel MISRA-C pour le langage C. Par ailleurs, pour s'assurer que le code respecte bien les règles de programmation définies, les normes [EN 50128] et [ISO 26262-6] demandent la réalisation d'une Analyse statique de code (cela peut être réalisé via un outil vérifiant l'application des règles de programmation).		
Source	[EN 50128] ☒ §7.3.4.25, §7.3.4.26, Tableaux A.12, A.19	[ISO 26262-6] ☒ §8.4.4, 8.4.5	[CEI 60880] ☐
	Incidents ☒ (20)	Autres ☐	
Justification	Tous les langages de programmation et toutes les instructions de ces langages ne peuvent pas être utilisés dans un développement critique car cela peut mettre à mal la fiabilité, la maintenabilité, la disponibilité et la sécurité du Logiciel. 20 incidents ont pour cause principale potentielle une mauvaise mise en œuvre des Unités Logicielles. Les origines peuvent être : - un non-respect ou une non-définition des règles de conception et de programmation - la non-réalisation d'analyse statique de code - la non-réalisation d'activités de cohérence du code avec les exigences (dont MMR) et/ou avec la conception détaillée. De ce fait, le degré d'importance est jugé majeur avec une contribution pour le fabricant jugée moyenne.		

5.3.2.4. Technique de Tests

5.3.2.4.1. Techniques de Tests générales

Type	**Création**	Degré d'importance	**Majeur**
		Contribution	2
Norme [NF EN 62304] (création d'une annexe)			
Constat	Dans la partie normative de la norme [NF EN 62304], les informations relatives aux techniques de tests sont absentes. Seul le §B.5.4 en annexe informative B identifie les notions de « boîte noire », « boîte blanche ». Comme cette partie est présente en annexe informative, le type de proposition d'amélioration est « Création ».		
Amélioration proposée	Outre les niveaux de tests boîte noire, boîte blanche, …plusieurs contraintes et notions de tests sont à ajouter : - Les contraintes concernant les types de tests. Exemple :		

	✓ Test nominal ✓ Test de robustesse ✓ Tests de performance - Les contraintes concernant les couvertures de test. Exemple : ✓ Couverture fonctionnelle ✓ Couverture structurelle - Les contraintes concernant les techniques de tests. Exemple : ✓ Classe d'équivalence et de partition d'entrées ✓ Tests des valeurs aux limites Les contraintes relatives à l'environnement de tests sont également à ajouter (Ex : test sur hôte ou sur cible – dans son environnement d'exécution) Ces notions peuvent être utilisées pour toutes les phases de tests (Vérification des Unités Logicielles, Tests d'Intégration et Essais du Logiciel). Les normes [EN 50128] et [ISO26262] introduisent ces notions de manière détaillées. Elles sont applicables en fonction du niveau d'intégrité de sécurité du Logiciel (SSIL). Cela peut s'adapter à la classe de sécurité du Logiciel. Par ailleurs, le chapitre 5.7 de la norme [NF EN 62304] concernant les Essais du Logiciel (Tests de Validation Logiciel) doit être applicable pour les Logiciels de Classe A.
Source	[EN 50128] ☐ §7.4.4.9, Tableau A.14, Annexe D / [ISO 26262] ☒ §9, 10, 11 / [CEI 60880] ☐ Incidents ☐ / Autres ☐
Justification	L'introduction de ces notions permettra aux fabricants de mieux appréhender les activités de tests (Vérification des Unités Logicielles, Tests d'Intégration et Essais du Logiciel). Le degré d'importance est jugé majeur avec une contribution pour le fabricant jugée moyenne car cela impactera la définition des activités de tests puis son application lors des activités de tests.

5.3.2.4.2. Activités de Tests Unitaires

Type	**Complément**	Degré d'importance	**Majeur**
		Contribution	2
Norme [NF EN 62304] § 5.5.4 : Critères supplémentaires d'acceptation de l'Unité Logicielle			

<table>
<tr><td>Constat</td><td colspan="3">Les critères d'acceptation supplémentaires de l'Unité Logicielle décrits au §5.5.4 laissent entrevoir l'activité de Tests Unitaires pour la classe C du Logiciel mais le fait que cela ne soit pas cité de manière explicite apporte un flou important.</td></tr>
<tr><td>Amélioration proposée</td><td colspan="3">Pour les Logiciels de Classe C, l'activité de Tests Unitaires est à ajouter dans les exigences de la norme [NF EN 62304] comme cela est le cas dans les normes [EN 50128], [ISO26262-6], [CEI 60880].
Cette activité est également demandée dans la table 3 de la norme FDA « Guidance for the Content of Premarket Submissions for Software Contained in Medical Devices » de Mai 2005 à partir du niveau « Moderate » (Blessure mineure pour le patient ou l'opérateur).

Les normes [EN 50128] et [ISO26262] introduisent les notions de couvertures des instructions, branches, conditions. Elles sont applicables en fonction du niveau d'intégrité de sécurité du Logiciel (SSIL). Cela peut s'adapter à la classe de sécurité du Logiciel.</td></tr>
<tr><td rowspan="2">Source</td><td>[EN 50128] ☒
§7.5, tableau A.21</td><td>[ISO 26262] ☒
§9</td><td>[CEI 60880] ☒
Annexe B.4.g</td></tr>
<tr><td>Incidents ☒ (4)</td><td colspan="2">Autres ☐</td></tr>
<tr><td>Justification</td><td colspan="3">L'activité de Tests Unitaires a pour rôle de démontrer via une analyse dynamique (exécution du Logiciel) que les Unités Logicielles décrites dans la conception détaillée ont bien été implémentées dans le code source (résultats de tests) et que le code ne contient pas de fonctionnalité non-désirée.
En effet, l'activité permet de s'assurer que toutes les instructions, branches, conditions sont bien exécutées (absence de code mort et passage dans tous les chemins d'exécution du code).
Cela est utile pour éviter de ne pas identifier rapidement l'origine d'un incident en l'exploitation.
Par ailleurs, 4 incidents ont pour cause primaire potentielle un manque d'activités de tests unitaires.
Le degré d'importance est jugé majeur.
La contribution pour le fabricant est jugée moyenne car il s'agit d'une activité ayant une charge importante mais cette activité est déjà réalisée chez la majeure partie des fabricants.</td></tr>
</table>

5.3.2.5. Interface, intégration Hardware / Software

Type	**Complément**	Degré d'importance	**Majeur**
		Contribution	3
Norme [NF EN 62304] § 5.3.2 : Elaboration d'une architecture pour les interfaces d'Eléments Logiciels			
Constat	L'exigence du §5.3.2 de la norme [NF EN 62304] demande l'élaboration et la documentation des interfaces internes (entre Eléments Logiciels) et des interfaces externes aux Eléments Logiciels (tant Logiciels que matériels) dans la conception architecturale. Néanmoins, aucune contrainte spécifique sur la définition des interfaces n'est présente dans la partie normative. Le §B.5.3 en annexe informative B apporte des informations supplémentaires sans indiquer quels sont les critères de définition des interfaces.		
Amélioration proposée	Interface externe : Au niveau de la conception architecturale ou dans un document spécifique de type Spécification des interfaces HW/SW (pour des besoins de réutilisation), spécifier les éléments de description des interfaces externes en cohérence avec les contraintes de sécurité : - Dans son environnement matériel - En interface avec d'autres Logiciels Les informations suivantes sont à apporter : - Moyen utilisé pour transférer les données : interfaces physique, protocoles, … - Contenu échangé : Interface logique, format du message, (voir la partie interface interne pour la description des interfaces). - Utilisation des ressources matérielles : les mémoires, registres, timers, interruptions, ports d'entrée / sortie, contraintes de temps (temps critique). Interface interne : Au niveau de la conception architecturale, pour chaque Elément Logiciel, spécifier les éléments de description des interfaces internes : - Définition données en Entrées/Sorties des EL - Définition et description des domaines de définitions des E/S (classes d'équivalences des données) - Définition des valeurs aux limites des données et le comportement en cas de dépassement - Pour les données d'entrée et de sortie à temps critique : ✓ contrainte de temps et exigences pour un fonctionnement correct, ✓ gestion des exceptions - L'existence de mécanismes de synchronisation entre les fonctions (voir point précédent) Des tests d'intégration sont à mettre en place pour couvrir les interfaces externes et internes (§5.6.2 [NF EN 62304].		

Source	[EN 50128] ☒ §7.3.4.19	[ISO 26262] ☒ Partie 4 : §7.4.6	[CEI 60880] ☐
	Incidents ☒ (47)	Autres ☐	
Justification	Les normes [EN 50128] et [ISO 26262] abordent cette problématique de manière détaillée. 47 incidents ont pour cause primaire potentielle le manque de description des interfaces. Cela explique le degré d'importance majeure. La contribution pour le fabricant est forte car un effort de description des interfaces est à réaliser puis dans un second temps à tester en intégration.		

5.3.2.6.Outils & SOUP

5.3.2.6.1.Qualification des outils

Type	**Création**	Degré d'importance	**Majeur**
		Contribution	2
Norme [NF EN 62304] (création d'un chapitre dédié)			
Constat	La norme [NF EN 62304], ne contient aucune prescription concernant la qualification des outils utilisés lors des activités (développement, validation, Logiciel générique configurable, …). Seul une mention indique que les outils doivent être contrôlés au §5.1.10.		
Amélioration proposée	Les 3 normes retenues pour l'étude contiennent une partie spécifique concernant la qualification des outils. Les outils sont classés selon l'impact d'un mauvais fonctionnement. 3 classes d'outils se dégagent : - Classe d'outil 1 : un mauvais fonctionnement n'a aucun impact sur le code exécutable (y compris les données). Exemple : éditeur de texte, outil d'aide à la conception, … - Classe d'outil 2 : outil utilisé pour le test ou la vérification. Un mauvais fonctionnement ne peut pas impacter directement le Logiciel exécutable mais pourrait ne pas révéler un défaut. Exemple : outil de tests, d'analyse statique, … - Classe d'outil 3 : génère des sorties susceptibles de contribuer, directement ou indirectement, au code exécutable (y compris les données). Exemple : compilateur Les tâches à mener pour la qualification des outils sont également fonction de la classe de sécurité du Logiciel.		

<table>
<tr><td></td><td colspan="3">Pour la classe d'outil 1, le choix de l'outil doit être justifié.
Pour la classe d'outil 2, en plus de la classe d'outil 1 :
- L'outil doit avoir un manuel utilisateur ou une spécification définissant ces fonctionnalités et ces contraintes d'utilisation.
- Une évaluation du niveau de confiance de l'outil doit être réalisée (étude des risques potentiels, mesures de préventions, validation de l'outil …) ou certificat.
- L'outil doit être géré en configuration
- Chaque nouvelle version d'outil doit être justifiée

Pour la classe d'outil 3, en plus de la classe d'outils 1 et 2 :
- La preuve de la conformité à la spécification de l'outil doit être apportée.
OU
- Une preuve de la mise en place de mesure permettant de se prémunir des défaillances de l'outil doit être apportée. (ex: génération redondante de code…)
OU
- Utiliser un compilateur certifié/qualifié

La démonstration d'une utilisation antérieure réussie de l'outil dans des environnements similaires est envisageable pour qualifier l'outil.

Un dossier de qualification des outils doit être réalisé contenant les tâches ou démonstrations cités au-dessus. Les versions des outils doivent être identifiées.</td></tr>
<tr><td rowspan="2">Source</td><td>[EN 50128] ☒
§ 6.7</td><td>[ISO 26262] ☒
Partie 8 : §11</td><td>[CEI 60880] ☒
§ 14</td></tr>
<tr><td>Incidents ☐</td><td colspan="2">Autres ☐</td></tr>
<tr><td>Justification</td><td colspan="3">L'objectif est de fournir la preuve que les défaillances potentielles des outils ne nuisent pas à la sécurité des données produites par l'ensemble des outils sans que cela ne soit détecté ([EN 50128] - § 6.7.1).
Le degré d'importance est jugé majeur.
La contribution pour le fabricant est moyenne car les tâches à réaliser pour qualifier les outils sont importantes mais réutilisables sur de nombreux projets (en cas d'utilisation de la version de l'outil qualifié).</td></tr>
</table>

5.3.2.6.2.Regroupement exigences SOUP

<table>
<tr><td rowspan="2">Type</td><td rowspan="2">Complément</td><td>Degré d'importance</td><td>Mineur</td></tr>
<tr><td>Contribution</td><td>1</td></tr>
<tr><td colspan="4">Norme [NF EN 62304] (création d'un chapitre dans le §5 : Processus de développement du Logiciel)</td></tr>
<tr><td>Constat</td><td colspan="3">Les exigences applicables au SOUP (Logiciel de provenance inconnue) sont disséminées dans toute la norme [NF EN 62304]. Plus de 12 chapitres contiennent des exigences relatives au SOUP, il y a des exigences d'identification, de définition, d'intégration, d'activités de gestion de risques, de maintenance, …</td></tr>
<tr><td>Amélioration proposée</td><td colspan="3">Un chapitre dédié contenant toutes les exigences applicables au SOUP doit être réalisé.

Par ailleurs, la notion de SOUP utilisée dans le domaine médical est réductrice et n'apporte pas de garantie car :
- il n'est pas développé pour être intégré dans le système développé
- il ne répond pas aux processus de réalisation de la norme applicable
- les enregistrements des processus de développement sont insuffisants ou non disponibles.

La norme [EN 50128] utilise la notion de Logiciel préexistant et possède un chapitre dédié (7.3.4.7) définissant les restrictions d'utilisation des Logiciels préexistants.
La définition de Logiciel préexistant dans la norme [EN 50128] est « Logiciel développé avant l'application dont il est ici question, incluant les Logiciels COTS (standards disponibles dans le commerce) et libres. »
A la différence du SOUP, le COTS :
- est défini par les besoins du marché
- son adéquation aux besoins est démontrée par un large éventail d'utilisateurs
- possède une documentation accessible et complète.

Des chapitres spécifiques sont aussi présents dans les normes [ISO 26262] et CEI [60880]</td></tr>
<tr><td rowspan="2">Source</td><td>[EN 50128] ☒
§7.3.4.7</td><td>[ISO 26262] ☒
Partie 8 : §12</td><td>[CEI 60880] ☒
§15</td></tr>
<tr><td>Incidents ☐</td><td colspan="2">Autres ☐</td></tr>
<tr><td>Justification</td><td colspan="3">Cette amélioration a pour objectif de clarifier la notion de SOUP et de faciliter l'application de ces exigences par le fabricant.
Le degré d'importance est jugé mineur et la contribution pour le fabricant est faible.</td></tr>
</table>

5.3.2.7.Paramétrage - Logiciels configurés par données d'application

Type	**Création**	Degré d'importance	**Majeur**
		Contribution	3
Norme [NF EN 62304] (création d'un chapitre dédié)			
Constat	La norme [NF EN 62304] ne contient aucune exigence concernant les Logiciels configurés par des données d'application (paramètres) ainsi que leur vérification et validation.		
Amélioration proposée	Un chapitre dédié aux Logiciels génériques configurés par des données d'application doit être ajouté dans la norme. La norme [EN 50128] contient un chapitre spécifique sur cette notion dans lequel toutes les phases du cycle de développement sont abordées. Le chapitre à ajouter doit contenir les exigences suivantes : - Réalisation d'un Plan de Préparation de l'Application définissant : ✓ Le processus à mettre en place pour chaque application spécifique ou pour chaque classe d'applications spécifiques ✓ Les Techniques et mesures à mettre en place (Méthodes de spécification en tableaux, …) ✓ Une description des activités de vérification et de validation (compatibilité des données d'application avec le Logiciel générique) - Spécification et Conception de l'application : ✓ Les exigences sur les conditions d'installation des données d'application ✓ Les exigences relatives aux données d'application ✓ L'emplacement des données d'application ✓ Une description précise des données d'application (Valeurs, type, Domaine, description…) - Activités de vérification et de validation, via : ✓ Liste de contrôle ✓ Tests fonctionnels des jeux de données (Tests de classes d'équivalence et de partitions d'entrées, valeurs aux limites, …) ✓ Preuve du caractère correct des données et de leurs intégrations Le Logiciel générique doit : - Identifier les fonctions configurables par les données d'application - Etre séparé des données d'application - Réaliser un contrôle de l'intégrité et de la cohérence des données d'application, lorsque cela est possible Des procédures de contrôles de modification doivent assurer qu'en cas de modification apportée au Logiciel générique, ce dernier peut être installé s'il est compatible avec les données d'application d'origine ou si ces données ont fait		

	l'objet d'une mise à jour. Si un outil de préparation de l'application est utilisé, il doit faire l'objet d'une qualification (voir la fiche concernant la qualification des outils).		
Source	[EN 50128] ☒ § 8	[ISO 26262] ☒ Annexe C	[CEI 60880] ☒ §7.1.4, 8.2.3.3, 14.3.5
	Incidents ☒ (5)	Autres ☐	
Justification	De nombreux Logiciels de DM sont architecturés avec un Logiciel dit « générique » configuré par des données spécifiques (paramètres). Les Logiciels configurés doivent avoir un niveau de confiance identique selon les différents jeux de données possibles. Par ailleurs, 5 incidents ont pour cause primaire une mauvaise gestion des données d'application. De ce fait, le degré d'importance est majeur. La contribution pour le fabricant est forte car un processus est à définir et à appliquer pour chaque application spécifique ou pour chaque classe d'applications spécifiques.		

5.3.3. AXE D'AMELIORATION LIE AUX STRATEGIES SAFETY & SECURITY

La norme [NF EN 62304] référence la norme [ISO 14971] concernant le Processus de Gestion des Risques.
La norme [ISO 14971] décrit de manière détaillée le Processus de Gestion des Risques (voir chapitre « Axe d'amélioration [ISO 14971] »).

Concernant les protections liées à la sécurité (Safety), la norme [NF EN 62304] ne définit pas d'architecture, de techniques / règles de Sécurité et de stratégie de tolérances aux fautes pour rendre robuste et sécuriser le Logiciel. Ce point fait l'objet de propositions d'amélioration.

De plus, la norme [NF EN 62304] identifie très peu de contraintes liées à la stratégie et aux protections de Sûreté (Security). Ce point fait également l'objet de propositions d'amélioration.

5.3.3.1. Partie Sécurité (Safety)

5.3.3.1.1. Stratégie Sécurité (Safety)

Aucune amélioration spécifique n'est proposée.

5.3.3.1.2. Protection Sécurité (Safety)

Type	**Complément**	Degré d'importance	**Majeur**
		Contribution	2
Norme [NF EN 62304] § 5.2.3 : Intégration des mesures de maîtrise du risque dans les exigences du Logiciel			
Constat	L'exigence du §5.2.3 de la norme [NF EN 62304] indique que le fabricant doit inclure dans les exigences les MMR mises en œuvre pour tenir compte des défaillances matérielles et des éventuels défauts du Logiciel en fonction du DM. Aucune précision n'est apportée pour la réalisation de cette exigence.		
Amélioration proposée	Le Logiciel doit réaliser de l'auto-surveillance afin de surveiller le matériel lors de l'exploitation, à intervalles de temps spécifiés, ainsi que le comportement du Logiciel. Les points qui peuvent être contrôlés sont : - Les défaillances aléatoires des composants matériels - Les zones mémoires contenant du code et des données invariables pour détecter des changement non prévus - Les erreurs de comportement du Logiciel (écarts de traitements, corruption / de données, contrôle de cohérence des données, …) - La transmission des données, ….		

	En cas de détection d'une défaillance, le Logiciel doit donner les réponses appropriées en temps voulu (mise en position de repli, fonctionnement dégradé, …).		
Source	[EN 50128] ☐	[ISO 26262] ☐	[CEI 60880] ☒ §6.2, Annexe B.3
	Incidents ☒ (13)	Autres ☐	
Justification	Ce type de contrôle permet au plus tôt de détecter une défaillance lors de l'exploitation pouvant provenir : - Du matériel - Des équipements en interface du Logiciel - Du Logiciel Par ailleurs, 13 incidents ont pour cause primaire une mauvaise ou non mise en place d'auto-surveillance. Ceci explique le degré d'importance majeur. La contribution pour le fabricant est jugée moyenne car les auto-surveillances sont à implémenter et à tester.		

5.3.3.2.Partie Sûreté (Security)

5.3.3.2.1.Introduction de la notion de Sûreté (Security)

Avant de proposer des améliorations normatives concernant la Sûreté (Security) du Logiciel, il convient d'introduire la notion de Sûreté (Security).

Le terme de Sûreté désigne ici l'ensemble des processus, moyens ou techniques mis en place pour assurer les propriétés suivantes du Logiciel :

- Confidentialité : fait d'assurer qu'une information n'est seulement accessible qu'à la personne ou entité dont l'accès est autorisé
- Disponibilité : aptitude du Logiciel à fonctionner lorsqu'on le sollicite
- Intégrité : assurance que ni les données, ni le Logiciel n'ont été modifiés, altérés ou supprimés lors d'un traitement ou d'une communication

La réalisation de ses propriétés a pour but de rendre toute attaque malveillante difficile à réaliser et ainsi prévenir la compromission des données et l'utilisation détournée du Dispositif Médical.

Les propositions d'amélioration se distinguent en deux catégories :

- les exigences lors du processus de développement et lors de son cycle de vie du Logiciel
- les exigences préconisant les protections relatives à la Sûreté du Logiciel et à l'environnement de développement

5.3.3.2.2.Stratégie Sûreté (Security)

Type	**Création**	Degré d'importance	**Majeur**
		Contribution	3
Norme [NF EN 62304]			
Constat	La norme [NF EN 62304] ne contient aucune exigence concernant le processus de développement de Logiciels permettant d'assurer la sûreté face aux actes de malveillance numérique. Seule une exigence est présente au § 5.2.2 e) indiquant que le fabricant doit inclure les exigences en matière de sûreté avec des exemples en Note. Ceci explique que le type de proposition de cette amélioration soit « Création ».		
Amélioration proposée	Les lignes directrices concernant la stratégie de Sûreté (processus Security) sont à aborder dans la norme [NF EN 62304] mais un référentiel spécifique traitant ce sujet est nécessaire. Une documentation spécifique à la Sûreté doit être mise en place : - Réalisation d'un Plan de Prévention des Menaces. Ce plan doit : ✓ Evaluer les vulnérabilités liées à l'environnement de développement. ✓ Evaluer les vulnérabilités du DM et du Logiciel tout au long du cycle de vie. ✓ Evaluer les fonctions critiques du DM ou du Logiciel. ✓ Evaluer les menaces concernant les propriétés Confidentialité/Disponibilité/Intégrité en fonction des vulnérabilités et fonctions critiques évaluées. ✓ Enoncer les exigences de contre-mesures et de sûreté pour toutes les menaces évaluées. ✓ Etre en lien avec le Plan de Développement Logiciel et le Plan de Gestion des Risques. ✓ Servir à la réalisation du Rapport de Sûreté du Logiciel vérifiant la prise en compte des exigences de Sûreté. - Réalisation d'un Rapport de Sûreté du Logiciel. Ce rapport doit : ✓ Evaluer les activités liées à la Sûreté du Logiciel. ✓ Evaluer la prise en comptes des exigences de sûreté émises dans le Plan de Prévention des Menaces. ✓ Statuer et donner un avis sur la Sûreté du Logiciel. Le Plan de Prévention des Menaces et le rapport Sûreté du Logiciel doit être intégré dans les analyses de Gestion des Risques [ISO 14971]. Les exigences applicables tout au long du cycle de vie du Logiciel sont à intégrer dans les activités. Cela concerne : - Spécification et Conception du Logiciel : ✓ Les exigences de sûreté ainsi que les exigences de contre-mesures émises par le Plan de Prévention des Menaces doivent être incluses dans la conception et indiquées comme telles.		

<table>
<tr><td></td><td colspan="3">
✓ Les stratégies d'accès limité aux Logiciels par l'authentification des utilisateurs sont à définir.
✓ Une matrice de traçabilité doit permettre de vérifier que les exigences de contre-mesures et de sûreté sont bien prises en compte dans la conception.
- Vérification et de Validation du Logiciel :
✓ Une activité dédiée de vérification et de validation de la sûreté du Logiciel est à mener.
✓ L'efficacité des mesures de protections et des fonctions de sûreté doit être confirmée et/ou démontrée par des tests adaptés
✓ Une activité de détection de failles de sécurité peut être menée (tests de pénétration).

- Exploitation et Maintenance du Logiciel :
✓ Les utilisateurs doivent être sensibilisés aux principes de sûreté afin d'être capable d'identifier des comportements anormaux du DM.
✓ Identifier les acteurs impliqués dans la mise à jour du Logiciel ainsi que leurs rôles. Une authentification lors de la mise à jour est obligatoire.
✓ Décrire les actions à réaliser pour la mise à jour du Logiciel.

Par ailleurs, un audit peut être réalisé afin de vérifier la mise en œuvre des exigences de sûreté dans l'environnement de travail, le processus de développement ainsi que les autres étapes du cycle de vie du Logiciel.
</td></tr>
<tr><td rowspan="2">Source</td><td>[EN 50128] ☐</td><td>[ISO 26262] ☐</td><td>[CEI 60880] ☒
§5.6, §5.7, §12.2</td></tr>
<tr><td>Incidents ☐</td><td colspan="2">Autres ☒
[FDA_Cybersecurity]</td></tr>
<tr><td>Justification</td><td colspan="3">L'augmentation du risque en matière de sûreté informatique et la législation imposant la confidentialité des données médicales imposent la mise en place d'un processus de Sûreté du Logiciel.
La constitution d'une stratégie pour le maintien d'un certain niveau de sûreté tout au long du cycle de vie du Logiciel est alors indispensable.

Le degré d'importance est donc jugé majeur et la contribution pour le fabricant est jugée forte (processus et stratégie à définir puis à appliquer).</td></tr>
</table>

5.3.3.2.3.Protection Sûreté (Security)

Type	**Création**	Degré d'importance	**Majeur**
		Contribution	3
Norme [NF EN 62304]			
Constat	La norme [NF EN 62304] ne contient aucune exigence concernant les protections permettant d'assurer la Sûreté face aux actes de malveillance numérique.		
Amélioration proposée	Les lignes directrices concernant les protections Sûreté sont à aborder dans la norme [NF EN 62304] mais un référentiel spécifique traitant ce sujet est nécessaire (de même que le point précédent). Les exigences concernant les protections de Sûreté du Logiciel sont les suivantes : - Activité de Conception : ✓ Les rôles et les privilèges des acteurs/utilisateurs doivent être clairement définis. ✓ La conception de la Sûreté (Security) ne doit pas être basée sur le secret des algorithmes. Toujours considérer qu'un attaquant peut avoir accès au code. ✓ Minimisation des données utilisées et la complexité du Logiciel. ✓ Eviter d'utiliser des Logiciels/librairies tiers (SOUP). A défaut, leur utilisation doit être justifiée et une étude de leur sûreté doit être menée et prise en compte. ✓ Utiliser des mécanismes appropriés pour l'authentification des utilisateurs. Utiliser une combinaison des informations suivantes : • Login • Mot de passe robuste (nombre minimum de caractère, caractère spéciaux, changement périodique du Mot de passe, …) • Contrôle du nombre de tentative de saisie du Login / Mot de passe • Support physique (badge, carte à puce) • Empreinte (information biométrique) ✓ Définir les mécanismes assurant la confidentialité des données (cryptage) lors de leur stockage et de leur communication. ✓ Mettre en place des mécanismes garantissant la disponibilité des fonctions critiques même en cas de compromission de la sûreté (Security). ✓ Toutes les communications doivent être sécurisées. L'ensemble suivant de défenses connues sont recommandées [NF EN 50159] : • Numéro de séquence • Datation • Délai d'attente • Identificateurs de source et de destination • Message en retour		

<table>
<tr><td></td><td colspan="3">

Procédure d'identification
Code de sécurité
Techniques cryptographiques

✓ Utiliser des protocoles de communication reconnus comme sécurisé.
✓ Utiliser des méthodes de cryptage/signature le plus fort possible selon le contexte.
✓ Les privilèges attribués aux utilisateurs doivent être réduits au minimum nécessaire pour assurer les fonctions dédiées au rôle associé.

- Activité de Développement
✓ Les spécificités du langage utilisé sont à prendre en compte (ex : fonction non sécurisées en C – strcpy vs strncpy)
✓ Contrôler les entrées : contrôle de l'intégrité (checksum : md5, sha, …), contrôle de la syntaxe et de la sémantique des données (si possible), authentification de la source (signature).
✓ Des dispositions doivent être prises contre la présence de fonction ou chemins cachés dans le Logiciel.

De plus, l'environnement de développement du Logiciel doit être sécurisé, de manière physique ou numérique. Les postes de travail doivent être sécurisés :
- Utilisation de Logiciels de protection (Pare-feu, antivirus, …)
- Cryptage des données : elles ne doivent pas être lues en cas d'accès physique
- Cloisonnement du réseau pour dissuader toute attaque numérique provenant de l'extérieur
- Accès physique règlementé et sécurisé (badge, …)
</td></tr>
<tr><td rowspan="2">Source</td><td>[EN 50128] ☐</td><td>[ISO 26262] ☐</td><td>[CEI 60880] ☒
§5.6, §5.7, §12.2</td></tr>
<tr><td>Incidents ☐</td><td colspan="2">Autres ☒
[FDA_Cybersecurity]
[NF EN 50159]</td></tr>
<tr><td>Justification</td><td colspan="3">L'augmentation du risque en matière de sûreté informatique et la législation imposant la confidentialité des données médicales nécessitent la mise en place de protections de Sûreté du Logiciel.
Le degré d'importance est donc jugé majeur et la contribution pour le fabricant est jugée forte car il faut définir et implémenter les protections de Sûreté du Logiciel.</td></tr>
</table>

5.3.4. SYNTHESE DES AMELIORATIONS [NF EN 62304]

Ce tableau présente une synthèse des améliorations pour la norme [NF EN 62304].

Catégorie	Critère	Amélioration proposée	Degré d'importance / Contribution	Nombre incidents (1)
[NF EN 62304]				
Processus	Organisation	Structure organisationnelle	Majeur / 2	-
		Evaluation/Certification	Majeur / 2	-
	Méthodes	Traçabilité globale	Mineur / 1	3
	Documentation	Documents à produire	Mineur / 1	-
		Contraintes de performance et d'environnement	Majeur / 2	4
		Installation	Majeur / 1	4
		Conception détaillée	Majeur / 3	12
	Maintenance	-	-	-
Techniques de développement	Général	Techniques de développement	Majeur / 2	-
	Architecture	Contraintes architecturales	Majeur / 2	5
	Règles de programmation	Contraintes d'implémentations et vérifications UL	Majeur / 2	20
	Technique de tests	Techniques de test	Majeur / 2	-
		Tests Unitaires	Majeur / 2	4
	Interface, intégration HW/SW	Interfaces et intégration du Logiciel	Majeur / 3	47
	Outils et SOUP	Qualification des outils	Majeur / 2	-
		Regroupement exigences SOUP	Mineur / 1	-
	Paramétrage	Logiciels configurés par données d'application	Majeur / 3	5
Stratégie Safety & Security	Stratégie Safety	-	-	-
	Protection Safety	Protection Safety – Auto-surveillance	Majeur / 2	13
	Stratégie Security	Lignes directrices pour stratégie Security	Majeur / 3	-
	Protection Security	Protection Security	Majeur / 3	-

(1) : Le « Nombre d'incidents » correspond aux incidents identifiés issus des déclarations de matériovigilance analysées lors de la première partie de l'étude étant en relation avec les recommandations.

5.4. AXES D'AMELIORATION POUR LA NORME [NF EN 62366]

La norme [NF EN 62366] concerne l'application de l'ingénierie de l'aptitude à l'utilisation aux Dispositifs médicaux.
Comme indiqué dans l'introduction de la norme, le processus d'ingénierie de l'aptitude à l'utilisation est destiné à obtenir une aptitude à l'utilisation raisonnable dans le but de minimiser les erreurs d'utilisation et à minimiser les risques associés à l'utilisation.
Dans cette norme, l'aptitude à l'utilisation se limite aux caractéristiques de l'interface utilisateur.

Sur la base de l'analyse des incidents réalisés dans la phase 1, 3 propositions d'amélioration sont présentes dans ce rapport.

5.4.1. INTERFACE UTILISATEUR

Type	**Complément**	Degré d'importance	**Mineur**
		Contribution	1
Norme [NF EN 62366] §5.7 : Conception et mise en application de l'interface utilisateur			
Constat	Les exigences sur l'interface utilisateur (IHM) sont manquantes dans la partie normative de la norme [NF EN 62366]. En effet, le chapitre 5.7 de cette norme concernant la conception et mise en application de l'interface utilisateur est très sommaire : « Le FABRICANT doit concevoir et mettre en application L'INTERFACE UTILISATEUR telle que décrite dans la SPECIFICATION DE L'APTITUDE A L'UTILISATION en utilisant, selon ce qui est approprié, les méthodes et les techniques D'INGENIERIE DE L'APTITUDE A L'UTILISATION. »		
Amélioration proposée	Introduire des exigences d'interface utilisateur dans la partie normative (§5.7), en se basant sur les informations fournies dans : - Tableau D.3 du §D.2.6. Ce tableau identifie des exemples d'exigences d'interface utilisateur concernant : ✓ Les généralités liées à l'édition dans l'IHM ✓ Les listes déroulantes (différenciation de la valeur sélectionnée, choix sans défilement, …) ✓ Les menus (coin supérieure gauche réservé à l'indicateur d'arrêt de l'alarme, …) ✓ L'affichage (luminance, contraste) ✓ Les dispositifs de commande (format des panneaux de commande, espacement entre les boutons, …) - Le §D.4.6.3 « Interface Utilisateur Logicielle ». Ce chapitre est présent dans l'annexe informative D. Il identifie les spécifications qui peuvent être incluses pour une interface utilisateur. A titre d'exemple, ces spécifications peuvent inclure : ✓ les dispositions d'écrans et de fenêtres y compris les dénominations, les polices, l'utilisation des couleurs et des		

<table>
<tr><td></td><td colspan="3">graphiques
✓ les flux de dialogue, y compris les événements sonores
✓ une description de l'interaction UTILISATEUR attendue avec les affichages et les commandes
- Tableau G.7 « Réaction à des signaux d'alarme et désactivation ». Ce tableau donne des indications de conception en ce qui concerne les signaux d'alarmes :
✓ Détectabilité : signalement sonore et visible à distance
✓ Compréhension : clarté des messages, distinction entre message d'alarme et message informatif, gestion des priorités, compréhension de la priorité de l'alarme, …
✓ Acquittement : identification de la cause de l'alarme, action requise pour résoudre la cause de l'alarme, …</td></tr>
<tr><td rowspan="2">Source</td><td>[EN 50128] ☐</td><td>[ISO 26262] ☐</td><td>[CEI 60880] ☐</td></tr>
<tr><td>Incidents ☒ (3)</td><td colspan="2">Autres ☐</td></tr>
<tr><td>Justification</td><td colspan="3">Cette amélioration a pour objectif d'ajouter des exigences dans la partie normative de la norme [NF EN 62366] pour que les fabricants définissent les interfaces utilisateur de leur Logiciel.

3 incidents ont pour cause primaire une mauvaise ergonomie due à un manque de clarté dans la présentation des éléments visualisés.
A titre d'information, ces 3 incidents sont relatifs à la catégorie de Logiciels :
- LAP (Logiciel non DM mais pris en compte dans le cadre de l'étude)

Le degré d'importance est jugé mineur la contribution pour le fabricant est faible car cette activité est déjà réalisée chez la majeure partie des fabricants.</td></tr>
</table>

5.4.2. NOTICE D'UTILISATION

<table>
<tr><td rowspan="2">Type</td><td rowspan="2">Complément</td><td>Degré d'importance</td><td>Majeur</td></tr>
<tr><td>Contribution</td><td>1</td></tr>
<tr><td colspan="4">Norme [NF EN 62366]
§6 : Document d'accompagnement</td></tr>
<tr><td>Constat</td><td colspan="3">Le chapitre 6 de la norme [NF EN 62366] n'indique pas d'obligation sur la fourniture d'une notice d'utilisation du Logiciel (Document d'accompagnement) possédant une interface utilisateur même si elle identifie des contraintes lorsque la notice d'utilisation est fournie.</td></tr>
</table>

Amélioration proposée	La fourniture d'une notice d'utilisation du Logiciel est à exiger. De plus, des exigences sur la formalisation d'une notice d'utilisation sont à ajouter au §6 de la norme [NF EN 62366]. Cela concerne : - Les contraintes liées aux données d'application (paramétrage) - Le périmètre d'utilisation - Les limites et restrictions d'utilisation		
Source	[EN 50128] ☐	[ISO 26262] ☐	[CEI 60880] ☐
	Incidents ☒ (11)	Autres ☐	
Justification	Cette amélioration a pour objectif d'ajouter des exigences dans la norme [NF EN 62366] concernant la formalisation de la notice d'utilisation afin que l'utilisateur possède toutes les informations pour utiliser correctement le Logiciel s'il est formé à son utilisation (voir la proposition d'amélioration sur la formation). 11 incidents sont dus à une incomplétude dans la notice d'utilisation. A titre d'information, ces 11 incidents sont relatifs aux catégories de Logiciels : - 4 incidents LAP (Logiciel non DM mais pris en compte dans le cadre de l'étude) - 4 incidents RT - 2 incidents IMA - 1 incident LABM De ce fait, le degré d'importance est jugé majeur. La contribution pour le fabricant est faible car cette activité est déjà réalisée chez la majeure partie des fabricants.		

5.4.3. FORMATION DES UTILISATEURS

Type	**Complément**	Degré d'importance	**Majeur**
		Contribution	2
Norme [NF EN 62366] §7 : Formation et supports de formation			
Constat	Le chapitre 7 de la norme [NF EN 62366] demande la réalisation d'une formation au DM spécifique si elle est requise pour l'utilisation sûre et efficace d'une fonction principale de service par l'utilisateur visé. Néanmoins, plusieurs manques ont été identifiés.		

<table>
<tr><td>Amélioration proposée</td><td colspan="3">Le chapitre 7 de la norme [NF EN 62366] doit être complété afin d'ajouter :
- Des contraintes sur la formation :
✓ Compétence du formateur
✓ Programmes détaillés de la formation
✓ Compétences requises pour l'utilisateur

- Des contraintes sur le contenu de la formation en introduisant des exigences sur les parties que doivent contenir une formation. Par exemple :
✓ Les fonctionnalités du Logiciel et la manière de les utiliser
✓ Les effets des erreurs récurrentes (REX : retour d'expérience…)

- Des contraintes sur la vérification du support de formation : Une fois le support de formation établi, il est nécessaire de procéder à une vérification de son contenu pour s'assurer qu'il est bien conforme aux fonctionnalités proposées par le Logiciel et qu'il n'y a aucune ambigüité.

- Des contraintes sur la preuve de la formation - preuve que le personnel médical a bien suivi une formation et qu'il est apte à utiliser le DM.</td></tr>
<tr><td rowspan="2">Source</td><td>[EN 50128] ☐</td><td>[ISO 26262] ☐</td><td>[CEI 60880] ☒ §12.4</td></tr>
<tr><td>Incidents ☒ (12)</td><td colspan="2">Autres ☐</td></tr>
<tr><td>Justification</td><td colspan="3">Cette amélioration a pour objectif d'améliorer les connaissances de l'utilisateur concernant le Logiciel du DM et ainsi d'avoir plus de confiance quant à sa maîtrise.

12 incidents sont dus à une incomplétude ou une incohérence dans la formation de l'utilisateur au Logiciel du DM.
A titre d'information, ces 12 incidents sont relatifs aux catégories de Logiciels :
- 5 incidents LAP (Logiciel non DM mais pris en compte dans le cadre de l'étude)
- 2 incidents RT
- 3 incidents IMA
- 1 incident LABM
- 1 incident SILBM

De ce fait, le degré d'importance est jugé majeur.
La contribution pour le fabricant est moyenne car cette amélioration nécessite une formalisation du processus de formation (contenu et vérification du contenu).</td></tr>
</table>

5.4.4. SYNTHESE DES AMELIORATIONS [NF EN 62366]

Ce tableau présente une synthèse des améliorations pour la norme [NF EN 62366]:

Amélioration proposée	Degré d'importance / Contribution	Nombre incidents (1)
[NF EN 62366]		
Interface Utilisateur	Mineur / 1	3
Notice d'utilisation	Majeur / 1	11
Formation des Utilisateurs	Majeur / 2	12

(1) : Le « Nombre d'incidents » correspond aux incidents identifiés issus des déclarations de matériovigilance analysées lors de la première partie de l'étude étant en relation avec les recommandations.

5.5. AXES D'AMELIORATION POUR LA NORME [ISO 14971]

5.5.1. INTRODUCTION

La norme [ISO 14971] concerne l'application de la Gestion des Risques aux Dispositifs Médicaux liés au patient, à l'opérateur ou d'autres personnes/équipements.

Cette norme identifie un processus pour permettre au fabricant d'identifier les phénomènes dangereux et les situations dangereuses associés aux dispositifs médicaux, d'estimer et d'évaluer les risques, de maîtriser ces risques et de surveiller l'efficacité de cette maîtrise.
Les exigences de la présente Norme internationale s'appliquent à tous les stades du cycle de vie d'un DM.

Cette norme possède un guide d'utilisation [TR 80002-1] sous la forme d'un rapport technique. Ce guide contient des détails complémentaires relatifs aux spécificités du Logiciel et constitue ainsi qu'un guide pour une meilleure compréhension de la norme dans une perspective Logicielle (Cf. [TR 80002-1], Introduction).

D'une manière générale, la norme [ISO 14971] et son guide [TR 80002-1], qui s'avère être un complément très détaillé des spécificités propres aux Logiciels, permet de décrire le processus de gestion de risques d'une façon suffisante. Néanmoins, une proposition d'amélioration a été identifiée.

5.5.2. SURETE DU LOGICIEL (SECURITY)

Type	Création	Degré d'importance	Majeur
		Contribution	3
Norme [ISO 14971] (création d'un chapitre dédié)			
Constat	La norme [ISO 14971] ne contient aucune prescription concernant les stratégies et protections relatives à la Sûreté du Logiciel (Security).		
Amélioration proposée	Les stratégies et protections relatives à la Sûreté du Logiciel (Security) sont à définir. Voir les fiches détaillées suivantes : - Stratégie Sûreté (Security) - chapitre 5.3.3.2.2 - Protection Sûreté (Security) - chapitre 5.3.3.2.3 Les lignes directrices concernant la stratégie et les protections de Sûreté (Security) sont à aborder dans la norme [NF EN 62304]. La gestion des risques relative à la problématique de Sûreté (Security) est à intégrer dans la norme [ISO 14971]. Le traitement de la problématique de Sûreté (Security) dans sa globalité est à réaliser dans un référentiel spécifique.		
Source	[EN 50128] ☐	[ISO 26262] ☐	[CEI 60880] ☒ §5.6, §5.7, §12.2
	Incidents ☐	Autres ☒ [FDA_Cybersecurity] [NF EN 50159]	
Justification	Ce point est un axe majeur d'amélioration.		

5.5.3. SYNTHESE DES AMELIORATIONS [ISO 14971]

Ce tableau présente une synthèse des améliorations pour la norme [ISO 14971]:

Amélioration proposée	Degré d'importance / Contribution	Nombre incidents (1)
[ISO 14971]		
Stratégies et protections Sûreté	Majeur / 3	-

(1) : Le « Nombre d'incidents » correspond aux incidents identifiés issus des déclarations de matériovigilance analysées lors de la première partie de l'étude étant en relation avec les recommandations.

6. ANNEXE 1 - PRESENTATION DES NORMES RETENUES POUR LA PHASE 3

6.1. EN 50128

Objectif
La norme [EN 50128] fournit un ensemble d'exigences pour le développement, le déploiement et la maintenance de tout Logiciel de sécurité destiné aux applications ferroviaires de contrôle-commande et de protection.

Domaine
Ferroviaire.

Définition des niveaux de sécurité
La norme [EN 50128] définit cinq niveaux d'intégrité de la sécurité (SSIL 0 à SSIL 4) déterminés selon le niveau de risque de l'utilisation du Logiciel intégré dans le sous-système :
- SSIL 0 : non lié à la sécurité
- SSIL 1 : faible
- SSIL 2 : moyen
- SSIL 3 : élevé
- SSIL 4 : très élevé

Ces niveaux d'intégrité de la sécurité permettent de définir les méthodes à appliquer à chaque phase du cycle de vie du Logiciel.

Description
La norme [EN 50128] s'applique à tout Logiciel de sécurité (Safety) destiné aux applications ferroviaires de contrôle-commande et de protection (Système bord et systèmes sol).
Cette norme peut être utilisée pour des Logiciels embarqués dans les systèmes ferroviaires mais aussi dans des systèmes de contrôle externes (non embarqués).
Cette norme prévoit également le traitement des Logiciels dits « génériques » (utilisables pour différentes applications) et de leurs paramétrages associés ainsi que l'intégration de Logiciels COTS. Elle décrit :
- l'organisation et la gestion du développement Logiciel, la compétence du personnel, le cycle de vie et la documentation
- l'assurance qualité du Logiciel
- le processus de développement du Logiciel
- le processus de développement de données d'application
- le déploiement et la maintenance du Logiciel
- les méthodes à mettre en œuvre à chaque phase du cycle de vie du Logiciel en fonction de son niveau d'intégrité de la sécurité

La norme ferroviaire ne couvre pas uniquement les Logiciels ayant un très haut niveau de sécurité (SSIL4) mais également de très nombreux Logiciels dont le niveau de sécurité (SSIL1 / SSIL2) peut être comparé à ceux du domaine médical. La norme apporte de nombreuses informations détaillées utiles en ce sens.
Tout comme dans d'autres secteurs, cette norme n'aborde pas du tout les aspects relatifs à la Security. Néanmoins, des informations sont disponibles en particulier dans la norme [EN 50159] qui traite de la communication de sécurité sur des systèmes de transmission ouverts.

Quelques exemples d'application

- Freinage d'urgence du train,
- Logiciel de contrôle porte d'accès du train,
- Système dit « homme mort »,
- IHM conducteur avec information de vitesse,
- Logiciel de Signalisation ferroviaire,
- Logiciel de gestion bord-sol,
- Poste de commande centralisé (systèmes de contrôle externes),
- Logiciels génériques et paramétrages associés.

Avantages vis-à-vis de la norme [NF EN 62304] :

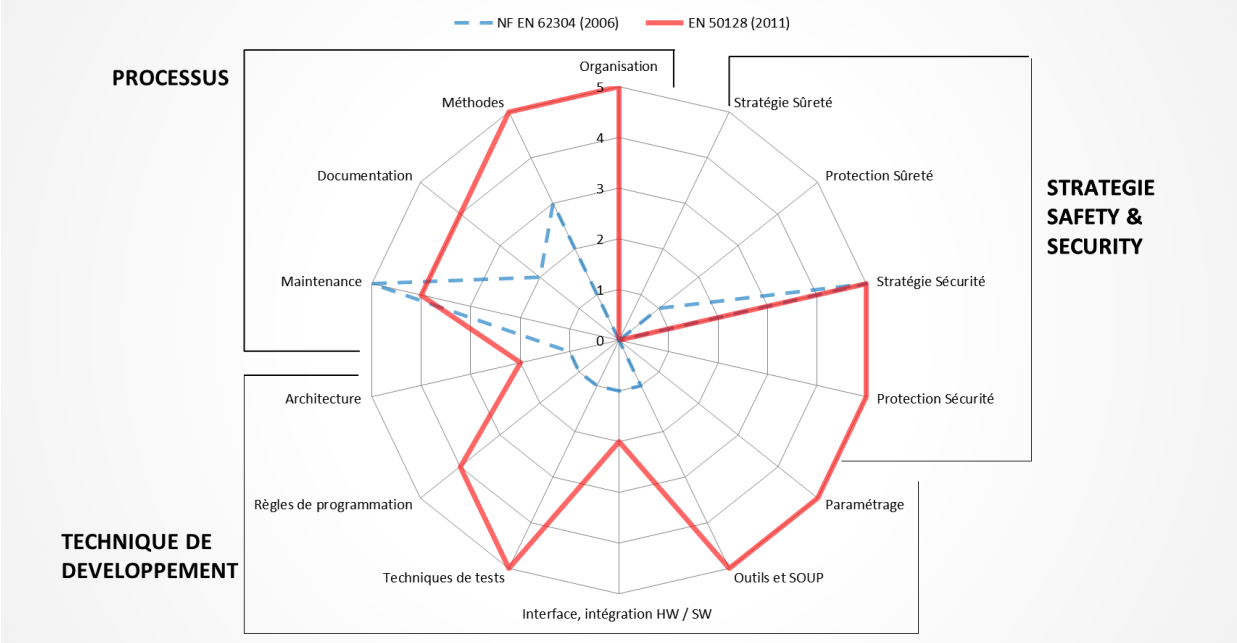
La norme [EN 50128] présente les avantages suivants :

- elle a été conçue sur la base d'une expérience industrielle forte pour répondre aux besoins du secteur en matière de sûreté de fonctionnement. Elle a fait l'objet d'une évolution majeure en 2011 et intègre de ce fait les nouvelles techniques de développement de Logiciel.
- elle définit des exigences concernant la structure organisationnelle, la relation entre organisations et la répartition des responsabilités impliquées dans les activités de développement, de déploiement et de maintenance.
- elle est très détaillée sur les processus à appliquer tout au long du cycle de vie du Logiciel.
- elle décrit les méthodes à appliquer selon le niveau d'intégrité de la sécurité pour chaque phase du cycle de vie du Logiciel.
- elle définit des règles de programmation et de protection Sécurité à mettre en place.
- elle aborde la partie tests dans son intégralité.
- elle définit des exigences concernant les outils utilisés dans le processus de développement qu'elle classe en trois catégories.
- elle définit en détail des exigences concernant les Logiciels paramétrables

Inconvénients vis-à-vis de la norme [NF EN 62304] :

Vis-à-vis de la norme [NF EN 62304], la norme [EN 50128] n'a pas de manque. Par contre, certains aspects ne sont pas abordés comme la Sûreté.

Diagramme en radar [NF EN 62304] vs [EN 50128] :



6.2. ISO 26262- PARTIE 6

Objectif
La norme [ISO 26262-6] définit les exigences pour le développement de Logiciels ayant à satisfaire à des contraintes de sécurité dans le domaine automobile. Afin d'être plus complet, ce chapitre s'appuiera également sur d'autres parties de la norme [ISO 26262].

Domaine
Automobile.

Définition des niveaux de sécurité
La norme [ISO 26262] définit quatre niveaux d'intégrité de la sécurité pour l'automobile (ASIL A à ASIL D). Cela exprime le niveau d'exigence de sécurité d'une fonction système qui doit être respecté lors de la conception et le développement du système, le niveau A étant le moins élevé et le niveau D le plus élevé. Si le risque est non influent sur la sécurité, on lui affecte le niveau QM (Quality Management).

Description
La norme [ISO 26262] - partie 6 s'applique à tout Logiciel intégré (embarqué) dans un véhicule et participant à une fonction de sécurité (Safety). Les véhicules intègrent de plus en plus d'électronique et de Logiciels utilisés pour assurer des fonctions de sécurité.

La norme [ISO 26262-6] définit des exigences pour :
- la spécification des exigences du Logiciel et des exigences de sécurité du Logiciel
- l'architecture du Logiciel
- l'implémentation du Logiciel
- les tests du Logiciel
- l'intégration du Logiciel
- la vérification du Logiciel

Avec, pour chaque phase du cycle de vie du Logiciel, des méthodes à appliquer en fonction du niveau d'intégrité de la sécurité du Logiciel.

La norme automobile est une des plus récentes et se focalise sur les aspects relatifs à la sécurité fonctionnelle (Safety). Elle couvre des Logiciels dont les niveaux de sécurité sont globalement similaires à ceux rencontrés dans le domaine médical. Elle définit également un processus adapté pour les Logiciels dits « configurables » et aborde des techniques de développement notamment d'architecture et d'interface Matériel / Logiciel.

La norme n'aborde pas ou peu les aspects relatifs à la Security.

Afin de traiter ce sujet dans le domaine automobile, des travaux ont été lancés afin de répondre au risque croissant d'attaques malveillantes. Des initiatives telles que le programme de recherche EVITA (E-Vehicle Safety Intrusion Protected Applications) a été lancé dès 2009 afin d'élaborer un référentiel pour la conception, la vérification et l'élaboration d'une architecture sécuritaire (Safety & Security) pour les réseaux embarqués dans l'automobile.

Note : les parties utiles à l'analyse de comparaison avec la [NF EN 62304] qui ne serait pas dans la norme [ISO 26262-6] mais présentes dans d'autres parties de la norme [ISO 26262] seront pris en considération dans l'activité de comparaison.

Quelques exemples d'application

- Système ADAS d'assistance au conducteur (participe à différentes fonctions telles que l'aide au parking automatique, la détection d'angles morts,…)
- Gestion du moteur
- Gestion des airbags
- Direction Assistée Electrique
- Système de gestion des batteries
- Ouverture / fermeture des ouvrants
- Frein parking
- Ouverture/commande à distance

Avantages vis-à-vis de la norme [NF EN 62304] :

La norme [ISO 26262-6] présente les avantages suivants :

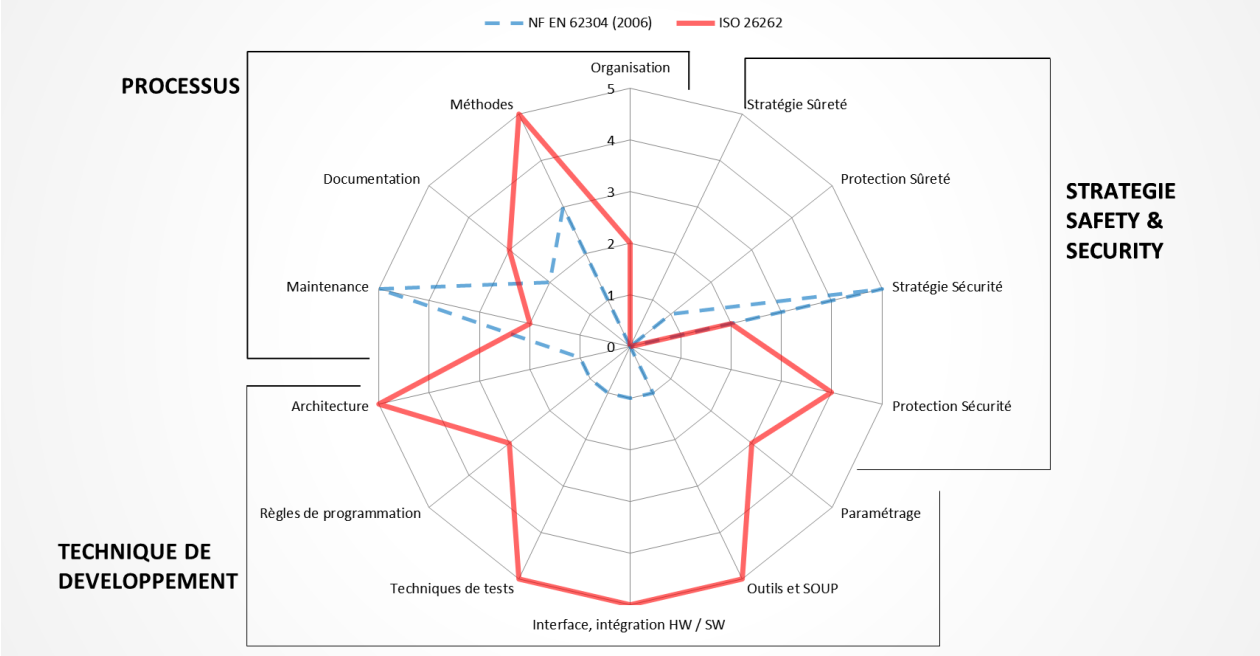
- elle fournit un cadre précis et structuré pour les phases du cycle de vie du Logiciel.
- elle donne une vision complète et concise en s'appuyant également sur les autres parties de la norme [ISO 26262].
- elle prend en compte l'environnement de développement du Logiciel et les contraintes matérielles.
- elle décrit les méthodes à appliquer selon le niveau d'intégrité de la sécurité pour chaque phase du cycle de vie du Logiciel.
- elle définit des règles de programmation et de protection Sécurité à mettre en place
- elle aborde la partie tests dans son intégralité
- elle définit des exigences concernant les outils utilisés dans le processus de développement.
- elle fournit des indications très détaillées sur les choix d'architectures et les informations à fournir concernant l'interface Matériel / Logiciel.

Inconvénients vis-à-vis de la norme [NF EN 62304] :

La norme [ISO 26262-6] présente les inconvénients suivants :

- elle donne peu de détails sur la maintenance du Logiciel.
- elle donne peu de détails sur la stratégie de Sécurité appliquée au Logiciel.

Diagramme en radar [NF EN 62304] vs [ISO 26262-6] :



6.3. CEI 60880

Objectif
La norme [CEI 60880] définit les principes et exigences relatives au Logiciel des systèmes de sûreté des centrales nucléaires.
Cette Norme est applicable aux Logiciels de hautes fiabilités des systèmes d'instrumentation et de contrôle-commande (I&C) programmés des centrales nucléaires de puissance, réalisant des fonctions de sureté de catégorie A.

Domaine
Nucléaire.

Définition des niveaux de sécurité
La norme [CEI 60880] concerne les Logiciels de classe I réalisant des fonctions de sureté de catégories A et/ou B et/ou C et/ou Non Classé.

Les catégories sont définies dans la norme [CEI 61226] et les classes sont définies dans la norme [CEI 61513].

La [CEI 61513] définit ainsi la classe des systèmes d'I&C importants pour la sûreté :
• les systèmes d'I&C de classe 1 sont principalement prévus pour réaliser des fonctions de catégorie A, mais peuvent aussi réaliser des fonctions de catégorie B et/ou C, ainsi que des fonctions non classées;
• les systèmes d'I&C de classe 2 sont principalement prévus pour réaliser des fonctions de catégorie B, mais peuvent aussi réaliser des fonctions de catégorie C, ainsi que des fonctions non classées;
• les systèmes d'I&C de classe 3 sont principalement prévus pour réaliser des fonctions de catégorie C, mais peuvent aussi réaliser des fonctions non classées.

Description
La norme [CEI 60880] fournit des prescriptions pour chaque étape du cycle de vie du Logiciel, de la spécification à l'exploitation. La vérification et le processus de modification du Logiciel sont également abordés. En outre, la norme consacre un chapitre aux défaillances de cause commune, aux outils et à leur qualification.

Avantages vis-à-vis de la norme [NF EN 62304] :
La criticité des Logiciels couverts par cette norme peut être considérée comme bien supérieure à celle des DM Logiciel. Néanmoins, cette norme apporte des notions intéressantes sur les points suivants :
- elle aborde le sujet de la sûreté qui est l'aptitude à prévenir les accès ou les modifications non autorisés (malveillance) des données et/ou des programmes.
- elle traite les moyens de défense contre les défaillances de cause commune.
- elle aborde la partie tests dans son intégralité
- elle définit des règles de programmation et de protection Sécurité à mettre en place
- elle introduit la problématique liée à la qualification des outils

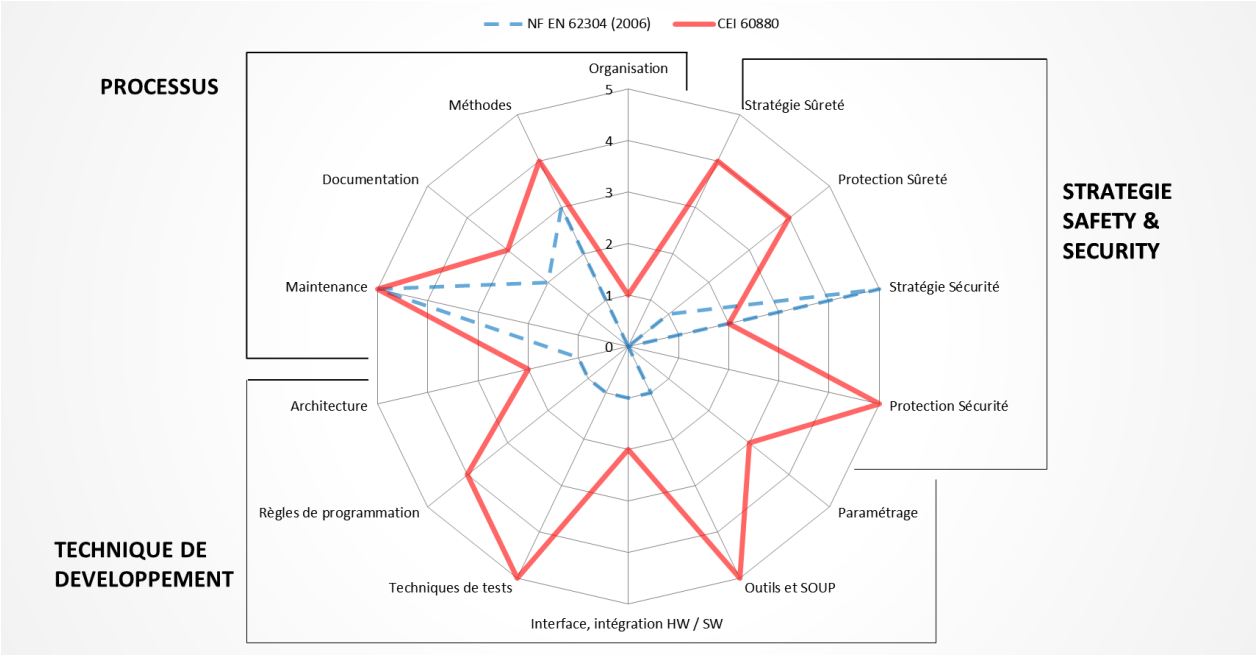
De plus, la norme aborde la problématique de la formation des opérateurs.

Inconvénients vis-à-vis de la norme [NF EN 62304] :

La norme [CEI 60880] présente les inconvénients suivants :

- elle n'aborde pas la nécessité d'une traçabilité complète
- la stratégie de Sécurité n'est pas assez développée
- elle utilise la formulation « il convient que », faisant des exigences une recommandation et non une obligation.

Diagramme en radar [NF EN 62304] vs [CEI 60880] :



7.ANNEXE 2 - Définition des critères et notes

7.1. CRITERES

Les critères utilisés pour la comparaison, classés en trois catégories, sont les suivants :

PROCESSUS :

- **Organisation** : Informations relatives à l'organisation du développement Logiciel à mettre en place, ainsi que sur les entités ou les personnes intervenant dans le cycle de vie du Logiciel.
  - Rôle : Informations relatives aux rôles des intervenants (Exemple : Chef de Projet, Concepteur, Chargé de vérification, Chargé de validation, …).
  - Responsabilité : Informations relatives aux responsabilités des intervenants dans le développement :
    - Exemple d'une des responsabilités d'un Concepteur : doit transformer les exigences spécifiées relatives au Logiciel en solutions acceptables
  - Compétence : Informations relatives aux compétences nécessaires pour chaque rôle
    - Exemple d'une des compétences d'un Chef de Projet : doit maîtriser la ou les normes définissant le processus de développement du Logiciel.
  - Indépendance : Informations relatives à l'indépendance entre les entités
    - Exemple : La même personne ne peut pas avoir les rôles de « Chargé de validation » et de « concepteur ».
- **Méthodes** : préconisation de méthodes à appliquer à chaque étape du cycle de vie du Logiciel en fonction du niveau de sécurité du Logiciel.
  - Cycle de vie : Informations relatives à un cycle de vie à adopter (Cycle en V, Cycle en cascade,…).
  - Description des activités de la partie descendante du cycle de vie
  - Présentation des activités de vérification : revue, fiche de relecture, vérification de traçabilités, équipe de vérification, …
  - Présentation des activités de validation : tests unitaires, tests d'intégration et tests de validation.
  - Traçabilité : Informations relatives à la traçabilité à mettre en œuvre au cours du développement Logiciel.
- **Documentation** : recommandations sur le format et le contenu de la documentation produite au cours du cycle de vie du Logiciel :
  - Document : Liste de documents à fournir pour chaque phase
  - Contenu : Informations relatives au contenu de chaque document (chapitre pré-défini, méthode de formalisation du contenu,…).
- **Maintenance** : Informations relatives à la maintenance du Logiciel.
  - Analyse d'impact : Informations relatives à une analyse d'impact d'une demande d'évolution ou de modification du Logiciel.
  - Non Régression : Informations relatives aux activités de non régression suite à la prise en compte de l'évolution /modification.

TECHNIQUE DE DEVELOPPEMENT

- **Architecture** : Informations relatives à des choix d'architecture sécurisé et de représentation :
  - Représentation via de la Modélisation : méthode formelle / semi-formelle, diagrammes de structure (classes, objet,..), diagrammes de comportement (activités, machines à états), …
  - Définition d'architecture sécurité. Exemple :
    - Détection / Traitement des erreurs
    - Redondance : Mise en parallèle de chaînes de traitement.

    - Diversification : Fonction ou Logiciel réalisé N fois de façon différentes.
- **Règles de programmation** (conception / codage) : contraintes sur les règles de conception et de programmation :
    - Obligation de réaliser un manuel de programmation : Informations relatives au manuel de programmation réunissant les règles de codage, de nommage et des bonnes pratiques
    - Présentation de langage de programmation adéquat : Liste de langage de programmation à utiliser ou non en fonction de la classe de sécurité du Logiciel
    - Analyse statique de code
- **Techniques de tests** : Informations relatives aux techniques de tests
    - Contraintes concernant les couvertures de test : Informations relatives aux couvertures que les tests doivent apporter (fonctionnelle, structurelle, robustesse, performance,...).
    - Contraintes concernant les techniques de tests : Informations relatives aux techniques de tests (classe d'équivalence, tests aux limites, …) et à leur environnement.
    - Constitution de jeux de données de tests : scénarios de tests à rejouer à chaque livraison, regroupés dans un catalogue ou une spécification de tests.
- **Interface, intégration Hardware / Software :** informations relatives à la définition de l'interface et de l'intégration entre le Matériel (Hardware) et le Logiciel (Software)
    - Définition des interfaces HW / SW en terme de contenant et contenu, respect des contraintes matérielles (horloge, datasheet, …).
    - Informations relatives aux processus d'assemblage d'éléments Logiciel et Matériel pour vérifier la compatibilité du Logiciel dans son environnement matériel.
- **Outils & SOUP** : Informations relatives aux outils utilisés dans le cadre du développement du Logiciel :
    - Choix des outils.
    - Qualification et validation : Informations relatives à la qualification des outils et à leur validation.
    - Contraintes concernant le choix et la maîtrise des COTS ou SOUP.
- **Paramétrage** : Informations relatives aux Logiciels paramétrables / configurables. Ces Logiciels s'adaptent aux exigences d'une application spécifique par le biais de paramètres, aussi appelés données d'application ou données de configuration.

STRATEGIE SAFETY & SECURITY

- **Protection d'implémentation Sécurité (SAFETY)** : Informations relatives aux techniques de protection du Logiciel et de tolérance aux fautes. Exemple :
    - Détection des défauts & diagnostic
    - Protection des données critiques
    - Codes de détection et de correction d'erreurs
    - Mise en position de repli / mode dégradé
- **Protection d'implémentation Sûreté (SECURITY)** : Informations relatives aux techniques de protection du Logiciel contre des actions non autorisées et intentionnelles (Malveillance). Exemple :
    - Autorisations d'accès des utilisateurs
    - Authentification (Mot de passe, Clé, empreintes,…)
    - Confidentialité
    - Intégrité des échanges
- **Stratégie Sécurité (SAFETY)** : Informations relatives aux analyses démontrant le niveau de sécurité du Logiciel (au sens Safety) contre des événements imprévisibles.
    - Réalisation de document démontrant la Sécurité
    - Analyse des risques : Informations relatives à l'analyse des risques potentiels pour la sécurité du Logiciel et à l'identification d'exigence de sécurité

  - Analyse dysfonctionnelle de type AEEL
  - Analyse des défaillances de mode commun
- **Stratégie Sûreté (SECURITY)** : Informations relatives aux analyses démontrant le niveau de sûreté du Logiciel (au sens Security) contre des actions malveillantes, non autorisées et intentionnelles.
  - Réalisation de document démontrant la Sûreté
  - Analyse de la sûreté : Informations relatives à l'analyse des menaces potentielles pour la sûreté du Logiciel

7.2. NOTE

A chacun des critères définis dans le chapitre précédent, une note (que l'on retrouve sur les axes des rosaces) allant de 0 à 5 a été attribuée :

- **0** : Absence totale du critère dans la norme
- **1** : Critère mentionné.
- **2** : Explication du critère succincte.
- **3** : Premier niveau de détail du critère.
- **4** : Niveau de détail intéressant mais incomplet ou tous les éléments du critère ne sont pas présents.
- **5** : Explication complète avec un niveau de détail élevé de tous les éléments du critère.

Si un élément d'un critère est manquant, la note ne peut être de 5, même si les explications et le niveau de détail est très élevé pour les autres éléments.