|  |
| --- |
|  |
| Titre du projet |
| Document de spécifications logicielles |
|  |
| **Nom de l'étudiant/des équipiers** |
| **Date (jj-mm-aaaa)** |

|  |
| --- |
|  |

# Suivi des changements

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Numéro de version | Date(jj-mm-aaaa) | Numéro de figure, table ou section | Type de changement | Description du changement | Originateur de la demande de changement |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

Table des matières

Suivi des changements 2

1. Introduction 4

1.1 But de ce document 4

1.2 Portée de ce document 4

1.3 Définitions, abréviations et acronymes utilisés dans ce document 4

1.4 Références 4

4. Spécifications non fonctionnelles 7

4.1 Aptitude fonctionnelle 7

4.2 Efficacité du rendement 7

4.3 Compatibilité 8

4.4 Facilité d’utilisation 8

4.5 Fiabilité 9

4.6 Sécurité 10

4.7 Maintenabilité 10

4.7 Portabilité 11

5. Contraintes 12

6. Documentation 13

# 1. Introduction

## 1.1 But de ce document

[Spécifiez le but de ce document. Typiquement, il s’agit de dresser la liste des fonctionnalités que devra comporter le logiciel ainsi que les fonctionnalités non fonctionnelles que le logiciel devra aussi rencontrer.]

## 1.2 Portée de ce document

[Définissez la portée de votre document. Ceci consiste à définir quel est la portion du logiciel à développer dont fait référence ce document. Il peut s’agit d’une partie du logiciel (un module ou un composant, par exemple) ou l’entièreté du logiciel.]

## 1.3 Définitions, abréviations et acronymes utilisés dans ce document

|  |  |
| --- | --- |
| Sigle | Description |
|  |  |

## 1.4 Références

[Cette section doit énumérer tous les documents qui sont cités en référence dans ce document. Spécifiez pour chaque document son titre, son ou ses auteur(s), la date ainsi que toute autre information pertinente.

Par exemple, il peut s’agir de normes qui ont été utilisées lors de la rédaction de ce document, de livres que vous avez consultés afin de mieux connaitre certaines technologies relatives à votre projet ou de sites web contenant de l’information sur UML.

Identifiez et numérotez les références en utilisant le préfixe REF.

Exemples :

Ref-1 : SEL (version 1.0), projet XYZ, Acme Inc, Département du développement logiciel, décembre 2011

Ref-2 : Document d’architecture (version 1.5), projet XYZ, Acme Inc, Département du développement logiciel, janvier 2012]

2. Description du projet
[Expliquez sommaire le projet à réaliser.

Si une description a déjà été fournie dans un autre document, vous pouvez lui faire référence en indiquant le titre du document, la date de rédaction, sa version ainsi que le chapitre de la description.]

3. Spécifications fonctionnelles [Listez l'ensemble des spécifications fonctionnelles. Les spécifications fonctionnelles représentent l'ensemble des fonctionnalités du système à développer. Leur objectif principal est de permettre à l'équipe de faire un consensus sur ce que le système doit faire. Typiquement, on doit faire réviser celles-ci avec les principaux intervenants du projet.

Il est aussi important de lui donner une priorité, soit qu’elle est essentielle au projet ou simplement souhaitable.

Finalement, si vous avez fait un document de vision, vous devez indiquer à quel besoin la spécification se réfère.]

|  |  |  |  |
| --- | --- | --- | --- |
| Numéro | Description | Priorité (Essentiel ou souhaitable) | Besoin |
| Exemple :SPEC-X | Le logiciel doit faire l’utilisation du système Microsoft Active Directory de l’entreprise pour authentifier les utilisateurs; afin d’avoir accès au système, les utilisateurs devront être membres d’un groupe prédéfini. | Essentiel | BES-X |

# 4. Spécifications non fonctionnelles

[La norme ISO/IEC 25000[[1]](#footnote-1) dresse une liste des spécifications non fonctionnelles qu’un système peut contenir. Chacune des définitions utilisées dans ce chapitre est tirée de cette norme. Pour chaque catégorie de spécifications non fonctionnelles, une liste de propriété possible est aussi spécifiée dans la norme.

Utilisez ces propriétés afin d’indiquer les spécifications non fonctionnelles auxquels votre application devra se soumettre.

Pour chaque spécification, vous devez aussi indiquer la propriété ISO 25000 de votre spécification et une mesure quantitative, c’est-à-dire comment mesurer votre spécification.]

## 4.1 Aptitude fonctionnelle

[La mesure dans laquelle le produit fournit des fonctions qui répondent aux besoins exprimés et implicites lorsque le produit est utilisé dans des conditions spécifiées.

* L'exhaustivité fonctionnelle : La mesure dans laquelle l'ensemble des fonctions couvre toutes les tâches spécifiées et les objectifs de l’utilisateur.
* L’exactitude fonctionnelle : La mesure dans laquelle le produit donne des résultats corrects avec le degré de précision nécessaire.
* La pertinence fonctionnelle : La mesure dans laquelle les fonctions sont adaptées pour des tâches déterminées et les objectifs de l'utilisateur.

Identifiez et numérotez ces spécifications en utilisant le préfixe SPEC-AF.]

|  |  |  |  |
| --- | --- | --- | --- |
| Numéro | Propriété ISO 25000 | Description | Mesure quantitative |
| Exemple : SPEC-AF-1 | L’exactitude fonctionnelle | Les services web vers l’application comptable ABC doivent être l’objet de test unitaire | Aucun test n’échoue. |

## 4.2 Efficacité du rendement

[La performance relative à la quantité de ressources utilisées dans des conditions déterminées.

* Comportement vis-à-vis du temps : Le temps de réponse et les délais de traitement et les taux de débit d'un système lors de l'exercice de ses fonctions dans des conditions déterminées par rapport à une référence établie (benchmark).
* Utilisation des ressources : Les quantités et les types de ressources utilisées lorsque le produit exerce sa fonction sous les conditions établies par rapport à une référence établie (benchmark).

Identifiez et numérotez ces spécifications en utilisant le préfixe SPEC-ER.]

|  |  |  |  |
| --- | --- | --- | --- |
| Numéro | Propriété ISO 25000 | Description | Mesure quantitative |
| Exemple : SPEC-ER-1 | Comportement vis-à-vis du temps | L’application doit répondre dans un délai de temps inférieur à 3 secondes | Aucun test de stress ne doit échouer. |

## 4.3 Compatibilité

[La mesure dans laquelle deux ou plusieurs systèmes ou composants peuvent échanger des informations et/ou à exercer les fonctions requises tout en partageant le même matériel ou l'environnement logiciel.

* Coexistence : La mesure dans laquelle le produit peut coexister avec d'autres produits indépendants dans un environnement commun partageant les ressources communes sans effets nuisibles.
* Interopérabilité : La mesure dans laquelle deux ou plusieurs systèmes ou composants peuvent échanger des informations et utiliser les informations qui ont été échangées.

Identifiez et numérotez ces spécifications en utilisant le préfixe SPEC-C.]

|  |  |  |  |
| --- | --- | --- | --- |
| Numéro | Propriété ISO 25000 | Description | Mesure quantitative |
| Exemple : SPEC-C-1 | Interopérabilité | L’application ABC doit pouvoir écrire dans la base de données de l’application XYZ. | Les tests unitaires et les tests de système de l’application XYZ ne doivent pas échouer lorsqu’exécutés avec les données produites par l’application ABC. |

## 4.4 Facilité d’utilisation

[La mesure dans laquelle un produit peut être utilisé par les utilisateurs spécifiés pour atteindre les objectifs spécifiés avec efficacité, efficience et satisfaction dans un contexte d'utilisation spécifié.

* Pertinence : La mesure dans laquelle les utilisateurs peuvent reconnaître si le produit est adapté à leurs besoins.
* Facilité d’apprentissage : La mesure dans laquelle un produit peut être utilisé par les utilisateurs spécifiés pour atteindre les objectifs d'apprentissage spécifique avec efficacité, efficience, sécurité et de satisfaction dans un contexte d'utilisation spécifiée.
* Opérabilité : La mesure dans laquelle le produit a des caractéristiques qui le rendent facile à utiliser et contrôler.
* Protection de l'utilisateur à faire une erreur : La mesure dans laquelle le système protège les utilisateurs à faire des erreurs.
* Interface utilisateur esthétique : La mesure dans laquelle l'interface utilisateur permet de plaire et de satisfaire l'interaction de l'utilisateur.
* Accessibilité : Le degré d'efficacité, d’efficience, de sécurité et de satisfaction quand les gens possédant le plus vaste éventail de capacités utilisent un produit.

Identifiez et numérotez ces spécifications en utilisant le préfixe SPEC-FA.]

|  |  |  |  |
| --- | --- | --- | --- |
| Numéro | Propriété ISO 25000 | Description | Mesure quantitative |
| Exemple : SPEC-FA-1 | Protection de l'utilisateur à faire une erreur | Les champs obligatoires doivent être marqués d’une mention à cet effet. | Les tests de validation de l’application devront tenir compte de ceci et ne doivent pas échouer. |

## 4.5 Fiabilité

[La mesure dans laquelle un système ou composant effectue les fonctions spécifiées dans des conditions spécifiées pour une période de temps spécifiée.

* Maturité : Le degré auquel un système répond aux besoins en matière de fiabilité dans des conditions normales d’opération.
* Disponibilité : La mesure dans laquelle un système ou composant est opérationnel et accessible lorsqu'il doit être utilisé,
* Tolérance aux pannes : La mesure dans laquelle un système ou composant fonctionne comme prévu, malgré la présence de défauts matériels ou de défauts logiciels.
* Rétablissement : La mesure dans laquelle le produit peut récupérer les données affectées et ré-établir l’état désiré si une interruption ou une panne se produit.

Identifiez et numérotez ces spécifications en utilisant le préfixe SPEC-F.]

|  |  |  |  |
| --- | --- | --- | --- |
| Numéro | Propriété ISO 25000 | Description | Mesure quantitative |
| Exemple : SPEC-F-1 | Disponibilité | Le système doit être disponible 99.9 % du temps | Les pannes au système doivent être documentées et sur une période de 6 mois, on sera en mesure de juger de la disponibilité. Si le système n’est pas disponible, 99.9 % des mesures devront être prises. |

## 4.6 Sécurité

[Le degré auquel les informations et données sont protégées de façon à ce que des personnes ou des systèmes non autorisés ne peuvent pas les lire ou les modifier et les personnes autorisées ou les systèmes ne se font pas refuser l'accès.

* Confidentialité : La mesure dans laquelle l'information et des données sont protégées contre la divulgation non autorisée d'informations ou de données qu'elle soit accidentelle ou délibérée.
* Intégrité : La mesure dans laquelle un système ou un composant empêche l'accès non autorisé ou la modification de programmes d'ordinateur ou de données.
* Non-réfutation : La mesure dans laquelle les actions ou les évènements peuvent être prouvés qu'ils ont eu lieu, de sorte que les évènements ou les actions ne peuvent pas être réfutés tard.
* Responsabilisation : La mesure dans laquelle les actions de l'entité peuvent être tracées de façon unique à l'entité.
* Authenticité : La mesure dans laquelle l'identité d'un objet ou d'une ressource peut être prouvée qu'elle est celle revendiquée.

Identifiez et numérotez ces spécifications en utilisant le préfixe SPEC-S.]

|  |  |  |  |
| --- | --- | --- | --- |
| Numéro | Propriété ISO 25000 | Description | Mesure quantitative |
| Exemple : SPEC-S-1 | Non-réfutation | Les actions des usagers doivent être gardées en mémoire afin d’être compatibles avec les normes de la FDA. | La présence d’un mécanisme pour garder en mémoire les actions des usagers dans le logiciel. |

## 4.7 Maintenabilité

[Le degré d'efficience et d'efficacité avec lesquels le produit peut être modifié.

* Modularité : La mesure dans laquelle un système ou un programme d'ordinateur est composé de composantes discrètes de sorte qu’un changement à l'une des composantes à un impact minime sur les autres composantes.
* Réutilisabilité : La mesure dans laquelle un actif peut être utilisé dans plus d'un système ou dans la construction d'autres actifs.
* Analysabilité : La facilité avec laquelle l'impact d'un projet de modification sur le reste du produit peut être évalué, ou le produit peut être diagnostiqué pour des déficiences ou pour les causes des échecs ou les parties qui doivent être modifiées peuvent être identifiées.
* Modifiabilité: La mesure dans laquelle un produit peut être modifié efficacement et d’une manière efficiente sans introduire des défauts ou dégradations de la performance.
* Testabilité : La facilité avec laquelle les critères d'essai peuvent être établis pour un système ou une composante et les essais peuvent être réalisés afin de déterminer si ces critères ont été satisfaits.

Identifiez et numérotez ces spécifications en utilisant le préfixe SPEC-M.]

|  |  |  |  |
| --- | --- | --- | --- |
| Numéro | Propriété ISO 25000 | Description | Mesure quantitative |
| Exemple : SPEC-M-1 | Modifiabilité | Les modifications au logiciel ne doivent pas entrainer de nouveaux problèmes. | Avant de mettre en production une mise à jour, celle-ci doit avoir passé l’ensemble des tests unitaires et de système. |

## 4.7 Portabilité

[La mesure dans laquelle un système ou composant ne peut être effectivement et efficacement transféré d'un matériel, logiciel ou d’un autre environnement opérationnel ou d’un environnement d’utilisation.

* Installabilité : La facilité avec laquelle le produit peut être installé avec succès et/ou désinstallé dans un environnement spécifié.
* Remplaçabilité : La mesure dans laquelle le produit peut être utilisé à la place d'un autre produit logiciel spécifié pour la même intention dans le même environnement.

Identifiez et numérotez ces spécifications en utilisant le préfixe SPEC-P.]

|  |  |  |  |
| --- | --- | --- | --- |
| Numéro | Propriété ISO 25000 | Description | Mesure quantitative |
| Exemple : SPEC-P-1 | Installabilité | Le logiciel doit être compatible avec Windows 7 et Windows XP. | Les tests d’installation doivent être réussis sur Windows 7 et sur Windows XP.s |

# 5. Contraintes

[Listez l’ensemble des contraintes dont vous devez tenir compte au niveau des spécifications du logiciel à construire. Les contraintes sont des facteurs externes au projet actuel, mais qui influence ce dernier.

Identifiez et numérotez les contraintes en utilisant le préfixe CON.]

|  |  |
| --- | --- |
| Numéro | Description |
| Exemple : CON-1 | Puisque l’entreprise possède un système financier de la firme ABC, le nouveau logiciel de facturation devra être en mesure de se connecter à ce dernier afin de pouvoir lire certaines informations comptables. |

# 6. Documentation

[Spécifiez les documents qui devront être produits dans le cadre de ce logiciel. Il ne s’agit pas ici des documents relatifs à l’activité de développement logiciel, mais bien les documents qui représentent des livrables au même titre que le logiciel lui-même. À titre d’exemple, il peut s’agir de manuel d’utilisation, de guide de maintenance ou de procédures de mise à jour de l’application.

Identifiez et numérotez les exigences non fonctionnelles en utilisant le préfixe DOC.]

|  |  |  |
| --- | --- | --- |
| Numéro | Titre du document | Description |
| Exemple :Doc-X | Manuel de l’utilisateur | Manuel qui permet à un nouvel utilisateur d’apprendre le fonctionnement du nouveau système. |

1. ISO/IEC 25000-2005, Ingénierie du logiciel — Exigences de qualité du produit logiciel et évaluation (SQuaRE) — Guide de SQuaRE, Organisation internationale de normalisation, Genève, Suisse [↑](#footnote-ref-1)