

Afin de vous aider à réaliser au mieux les développements et travaux réalisés pour les clients, cette partie de la documentation voit le jour. Le but est de pouvoir donner à tous les bonnes pratiques, à la fois internes (Chez Subteno IT), et externes, afin que la qualité de notre travail soit au rendez-vous pour tous les clients.

Cette partie sera longue, mais couvrera de nombreux points, et s'adaptera tout particulièrement aux personnes faisant partie du service qualité, afin de leur permettre d'avoir des ressources pour effectuer des revues de qualité :)

Prêts ? C'est parti !

1. Introduction

Tout d'abord, avant de commencer, nous allons lister ce qui sera traité, et ce qui ne sera pas traité au cours de cette documentation. Cette partie de la documentation vise tous les acteurs techniques de Subteno IT. Sera donc traité :

- Règles de développement
- L'arborescence d'un dépôt client
- Guide de test fonctionnel
- Guide de test technique
- Critères de validation qualité

Dans un second temps, nous aborderons certains cas spécifiques de travail, afin d'exemplariser le fonctionnement de l'entreprise vis à vis des spécificités rencontrées. Par exemple, nous expliciterons pourquoi nous avons choisi d'implémenter un dossier "**Extras**" dans les dépôts clients et réalisons des liens symboliques dans ce dossier pour les modules non construits par Subteno IT.

A l'attaque !

2. Règles de développement

TO OD

3. Arborescence d'un dépôt client

TO DO

4. Guide de test fonctionnel

TO DO

5. Guide de test technique

TO DO

6. Critères de validation qualité

Afin de permettre au service qualité de bénéficier d'une feuille de route de validation, nous avons listé ici toutes les exigences pouvant être prises en compte par le service qualité pour notifier d'une non validation de la tâche / du développement.

6.1 Validation : Backlog

Lorsque la tâche se situe dans le Backlog, avec un point rouge, le service qualité doit vérifier les points suivants :

Nom de la tâche

Le nom de la tâche doit être explicite, et présenter rapidement et simplement son contexte.

Client rattaché

Une tâche doit obligatoirement être assignée à un client, ou bien à Subteno IT. On ne travaille pas dans le vide chez Subteno IT, on le fait forcément pour quelqu'un, même si c'est pour nous-même ! :)

Date d'échéance

La date d'échéance doit obligatoirement être renseignée. Elle permet de planifier efficacement le travail sur toutes les tâches clientes par la suite.

Heures initialement prévues

Cet indicateur permet d'estimer le temps de développement d'une tâche. C'est un indicateur très important, couplé à la date d'échéance, et il doit donc forcément être renseigné à la création de la tâche.

Description

Les tâches doivent **obligatoirement** être décrites pour pouvoir être comprises par tous. Pour cela, l'utilisation du template de description de tâche est permise (même conseillé !), afin de décrire au mieux chaque tâche créée.

Si tout va bien, le point rouge passe à vert, prêt à être planifié.

6.2 Validation : Doing

Lorsque la tâche se situe dans le Doing, avec un point vert, le service qualité doit vérifier les points suivants :

Pipelines validées

Le service qualité doit vérifier que les pipelines lancées lors de la réalisation de la Merge Request sont bien passées et ne présentent aucun défaut. En cas de défaut, le développement ne peut pas être validé.

Respect du standard Odoo (Le code produit n'est-il pas un code dupliqué ? Le standard est-il modifié ?)

Le respect du standard Odoo implique :

- De ne pas avoir modifié le code du standard. Ce type de développement ne peut pas être mergé.

- De ne pas avoir “réinventé la roue” : On essaie d'utiliser au maximum les méthodes apportées par le standard Odoo.
- De ne pas avoir fait du “Code Spaghetti”. On doit pouvoir s'y retrouver simplement.
- De ne pas avoir, au travers de son développement, cassé un fonctionnement standard.
- De ne pas avoir développé en français.

Réalisation des traductions

Tout développement réalisé doit être traduit dans la langue de l'utilisateur final. Un défaut de traduction ne pouvant pas être livré en production. Ces traductions doivent être réalisées **dans le code**, et non en base.

Droits d'accès

En cas de développement nécessitant la mise en place de droits d'accès spécifiques, il faut vérifier leur mise en place correcte dans l'ERP.

Il est aussi nécessaire de vérifier qu'un développement ne risque pas de poser problème en cas de droit d'accès différents entre 2 utilisateurs distincts (Une fois le développement réalisé, il faut le tester avec un utilisateur ayant le plus bas niveau d'accès et vérifier que le fonctionnement est toujours le même).

Normes des commits (Squashés, signés, bien nommés, en français)

Il faut vérifier l'état de l'historique de commit de la merge-request qui a été produite. Si son état n'est pas valide, alors la merge-request sera bloquée.

Lien Odoo

Une merge-request doit être reliée à une issue Gitlab, elle-même reliée à une tâche Odoo. Si aucun lien n'est constaté, le développement doit être rattaché à une tâche avant d'être validé, pour assurer sa traçabilité.