Retour au Dossier Organisationnel	
Sujets connexes	Cartographie fonctionnelle de LoGeAs Proc#145 : Procédure de Gestion des Versions [OBSOLÈTE] Procédure de versionning des codes sources (ancienne Procédure #30) [OBSOLÈTE] Procédures de gestion du dépôt GitLab [OBSOLÈTE] Procédure de Dépôt des Codes Sources (ancienne Procédure #09)

Gestion des tickets

Informations qualité

Suivi des modifications majeures	juin 2026 - Nicolas Marchand - Refonte complète 5 mars 2015 - Nicolas Marchand - Création du document 27 novembre 2015 - Nicolas Marchand - Evolution vocabulaire, non-gestion du temps dans projeQtOr 27 novembre 2015 - Valentin Barrere - Correction orthographique 02 aout 2017 - Nicolas Marchand - Portage sur DoKuWiKi & Evolutions ajout “A tester”
Suivi des approbations	ce fait sur la plateforme de cartographie fonctionnelle
Objet	L'objet de ce document est d'indiquer la procédure à suivre lors de la vie d'un ticket
Destinataires	**- Validation des modifications :** Gérant **- Approbation du document** : Equipe dev & Equipe Ass

Qualification des tickets

Types de tickets

Type	Quand l'utiliser ?	Info clé à fournir	Priorité	Traitement	En savoir plus
Bug	Le logiciel produit un résultat incorrect ou inattendu	Étapes de repro + résultat observé vs attendu	Bloquant → Majeur → Mineur	Investigation immédiate si bloquant	types_tickets_bug.pdf types_tickets_bug.odt
Évolution	La fonctionnalité demandée n'existe pas encore	Besoin métier + cas d'usage + critères d'acceptation	Selon impact métier et charge estimée	Planification sprint / roadmap	types_tickets_evolution.pdf types_tickets_evolution.odt
Amélioration	La fonction existe mais pourrait être mieux conçue	Usage concerné + problème concret + gain estimé	Faible à moyenne selon contexte	Backlog, trié par valeur	types_tickets_amelioration.pdf types_tickets_amelioration.odt

Type	Quand l'utiliser ?	Info clé à fournir	Priorité	Traitement	En savoir plus
Question / Assistance	L'utilisateur cherche à comprendre comment faire	Ce qu'il essaie de faire + ce qu'il a tenté	Non applicable	Réponse assistance, pas devs	types_tickets_question-assistance.pdf types_tickets_question-assistance.odt

Arbre de décision

1. Le logiciel produit un résultat **incorrect ou inattendu** ? → **Bug**
2. La fonctionnalité demandée **n'existe pas encore** dans le logiciel ? → **Évolution**
3. La fonctionnalité **existe mais est difficile** à utiliser ou trop lente ? → **Amélioration**
4. L'utilisateur **cherche à comprendre** comment utiliser le logiciel ? → **Question / Assistance**

Les niveaux de gravité

Niveaux de gravité des tickets

Bloquant

Le logiciel est inutilisable pour tout ou partie des utilisateurs. Aucun contournement possible. Le travail est à l'arrêt.

Quand utiliser ce niveau ?

Situations
Fonctionnalité principale totalement non fonctionnelle
Aucun contournement disponible
Impact sur tous les utilisateurs ou un métier entier
Perte de données possible ou avérée
Blocage d'un processus critique (clôture, AG, collecte…)
Faille de sécurité ou risque RGPD avéré

Critères de classification

Critère	Valide ?
Aucun contournement possible	✓ obligatoire
Impact sur l'ensemble ou la majorité des utilisateurs	✓ obligatoire
Travail totalement à l'arrêt	✓ obligatoire
Risque de perte ou corruption de données	à vérifier

Signaux typiques

Signal
Personne ne peut se connecter

Signal
Toutes les saisies sont perdues
Le module cotisations est inaccessible à tous
Les exports produisent des données erronées
Une corruption de données est détectée
Des données personnelles sont exposées (faille RGPD)

Délai de traitement

Indicateur	Valeur
Délai cible d'investigation	< 4 heures
Traitement	Immédiat — escalade si besoin
Contournement	Aucun disponible

Pièges fréquents

Piège
Qualifier “bloquant” uniquement parce que c'est urgent pour soi
Confondre avec majeur si un collègue peut prendre le relais
Un module lent n'est pas bloquant (c'est majeur)
Un seul utilisateur bloqué n'est souvent pas bloquant

Exemple

✗ À éviter	✓ Bon exemple
Objet : Problème connexion urgent *Ça marche pas depuis ce matin, c'est urgent, on a une réunion.*	**Objet :** [Bloquant] Impossible de se connecter — tous les agents — prod v2.4.1 *Depuis 8h00, aucun agent ne peut se connecter. Page connexion se recharge sans message d'erreur. 8 postes testés. Firefox 124 et Chrome 124. Prod v2.4.1. Aucun contournement.*

Confusion fréquente

<note warning> “Urgent” n'est pas un niveau de gravité — c'est une émotion. Le niveau Bloquant se définit par l'**absence de contournement** et l'**impact total** sur les utilisateurs.
À distinguer de : Majeur (contournement pénible disponible) </note>

Majeur

Une fonctionnalité importante est dégradée ou partiellement non fonctionnelle. Un contournement existe mais est pénible ou risqué.

Quand utiliser ce niveau ?

Situations
Fonctionnalité importante impactée mais pas critique

Situations
Contournement possible mais coûteux en temps
Impact sur plusieurs utilisateurs ou un rôle clé
Risque d'erreur ou de perte de données potentielle
Gêne significative sur le travail quotidien

Critères de classification

Critère	Valide ?
Fonctionnalité importante impactée	✓ obligatoire
Contournement pénible, risqué ou très lent	✓ obligatoire
Impact sur plusieurs utilisateurs ou rôle clé	✓ obligatoire
Travail possible mais fortement dégradé	✓ obligatoire

Signaux typiques

Signal
Export CSV incomplet ou mal formé
Calcul de cotisation incorrect pour certains cas
Filtre qui ne se réinitialise pas correctement
Lenteurs importantes après une mise à jour
Un seul utilisateur bloqué avec contournement possible
Bug sur données financières avec correction manuelle risquée

Délai de traitement

Indicateur	Valeur
Délai cible d'investigation	< 48 heures
Traitement	Priorité haute
Contournement	Existe mais pénible ou risqué

Pièges fréquents

Piège
Ne pas qualifier “majeur” uniquement parce que c'est gênant
Vérifier si le contournement est vraiment risqué (sinon mineur)
Un bug majeur sur une fonction peu utilisée peut rester en backlog
Confondre avec bloquant si 1 seul utilisateur est impacté

Exemple

✗ **À éviter**	✓ **Bon exemple**
Objet : Bug export *L'export marche pas bien depuis hier, c'est bloquant pour la compta.*	**Objet :** [Majeur] Colonne mode_paiement absente export CSV cotisations — régression v2.4.1 *Colonne “mode_paiement” disparue depuis v2.4.1. Contournement : saisie manuelle (~3h). 2 utilisateurs impactés. Clôture dans 4 jours. Chrome 124, prod.*

Confusion fréquente

<note warning> Si **aucun** contournement n'existe → c'est **Bloquant**. Si le contournement est simple et sans risque → c'est **Mineur**.
La règle clé : Contournement pénible ou risqué = **Majeur**. </note>

Mineur

Un comportement incorrect mais sans impact majeur sur le travail. Un contournement simple existe. Le bug est gênant mais pas bloquant.

Quand utiliser ce niveau ?

Situations
Comportement incorrect mais d'impact limité
Contournement simple et sans risque
Impact sur peu d'utilisateurs ou cas rares
Pas de risque de perte ou corruption de données
Pas d'urgence métier associée

Critères de classification

Critère	Valide ?
Bug réel mais impact fonctionnel limité	✓ obligatoire
Contournement simple, rapide et sans risque	✓ obligatoire
Pas de risque sur les données	✓ obligatoire
Pas d'urgence métier immédiate	✓ obligatoire

Signaux typiques

Signal
Libellé ou message d'interface incorrect
Tri par colonne qui ne fonctionne pas
Une date s'affiche dans le mauvais format
Un champ optionnel qui se vide sans raison
Une icône ou couleur qui ne correspond pas
Faute de frappe dans un libellé

Délai de traitement

Indicateur	Valeur
Délai cible d'investigation	Prochain sprint
Traitement	Planification normale
Contournement	Simple, sans risque

Pièges fréquents

Piège
Ne pas ignorer les mineurs : ils s'accumulent et dégradent l'expérience
Un bug “visuel” peut cacher un bug fonctionnel plus grave
Vérifier qu'il n'affecte pas des processus réglementaires
Sous-estimer l'impact sur les nouveaux utilisateurs

Exemple

✗ À éviter	✓ Bon exemple
Objet : Dates bizarres *Les dates s'affichent pas bien dans la liste des adhérents.*	**Objet :** [Mineur] Format date naissance MM/DD/YYYY au lieu de DD/MM/YYYY — régression v2.4.1 *Dates de naissance en format US dans la liste adhérents depuis v2.4.1. Données correctes en base. Contournement : ouvrir la fiche. Non reproductible sur les fiches individuelles.*

Confusion fréquente

<note warning> Si un bug visuel entraîne des erreurs de saisie ou des décisions incorrectes → reclasser en **Majeur**.
La règle clé : Contournement trivial + données correctes = **Mineur**. </note>

Amélioration / Évolution

Aucun comportement incorrect. La fonctionnalité fonctionne comme prévu mais pourrait être améliorée, ou une nouvelle capacité est souhaitée.

Quand utiliser ce niveau ?

Situations
Aucun bug : le logiciel fonctionne correctement
Demande de confort, d'ergonomie ou de performance
Fonctionnalité inexistante mais souhaitée
Automatisation d'une tâche manuelle répétitive
Pas d'urgence métier immédiate

Critères de classification

Critère	Valide ?
Aucun comportement incorrect	✓ obligatoire
Le logiciel fonctionne comme prévu	✓ obligatoire
Objectif : confort, efficacité ou nouvelle capacité	✓ obligatoire
Priorisé selon la valeur métier apportée	✓ obligatoire

Signaux typiques

Signal
Processus fonctionnel mais avec trop de clics
Fonctionnalité inexistante mais souhaitée
Automatisation d'une tâche manuelle répétitive
Interface lisible mais perfectible
Nouveau besoin métier émergent

Délai de traitement

Indicateur	Valeur
Délai cible	Selon priorité roadmap
Traitement	Backlog, priorisé par valeur
Contournement	Non applicable — pas de bug

Pièges fréquents

Piège
Ne pas mettre en “amélioration” ce qui est en réalité un bug
Quantifier le gain attendu — sans chiffre, difficile à prioriser
Distinguer amélioration (existe) et évolution (n'existe pas)
Ne pas confondre l'absence de bug avec l'absence de problème

Exemple

✗ À éviter	✓ Bon exemple
Objet : Interface à améliorer *Le logiciel est bien mais on pourrait améliorer quelques trucs sur la page d'accueil.*	**Objet :** [Amélioration] Saisie adhérent — réduction 12 clics → 4 clics *Création adhérent : 12 clics sur 4 écrans. Cible : 4 clics sur 1 écran. Constat sur 3 secrétaires : ~15 min/jour perdues. Champs obligatoires à regrouper sur un écran unique.*

Confusion fréquente

<note warning> Si le logiciel produit un résultat incorrect → c'est un **Bug** avec un niveau de gravité (Bloquant/Majeur/Mineur).
Amélioration/Évolution ne sont pas des niveaux de gravité — ce sont des types de tickets sans gravité associée. </note>

Synthèse — Les 4 niveaux de gravité

Tableau comparatif

Niveau	Situation	Contournement	Délai cible	Traitement
Bloquant	Travail totalement à l'arrêt, aucun contournement	Aucun	< 4 heures	Investigation immédiate, escalade
Majeur	Fonctionnalité dégradée, contournement pénible	Pénible ou risqué	< 48 heures	Priorité haute
Mineur	Bug limité, contournement simple	Simple et sans risque	Prochain sprint	Planification normale
Amélioration	Pas de bug, confort ou nouvelle capacité	Non applicable	Selon roadmap	Backlog, par valeur

La règle du contournement

Contournement disponible ?	Niveau
Aucun contournement possible	**Bloquant**
Contournement pénible, lent ou risqué	**Majeur**
Contournement simple et sans risque	**Mineur**
Pas de bug — logiciel fonctionnel	**Amélioration / Évolution**

Arbre de décision

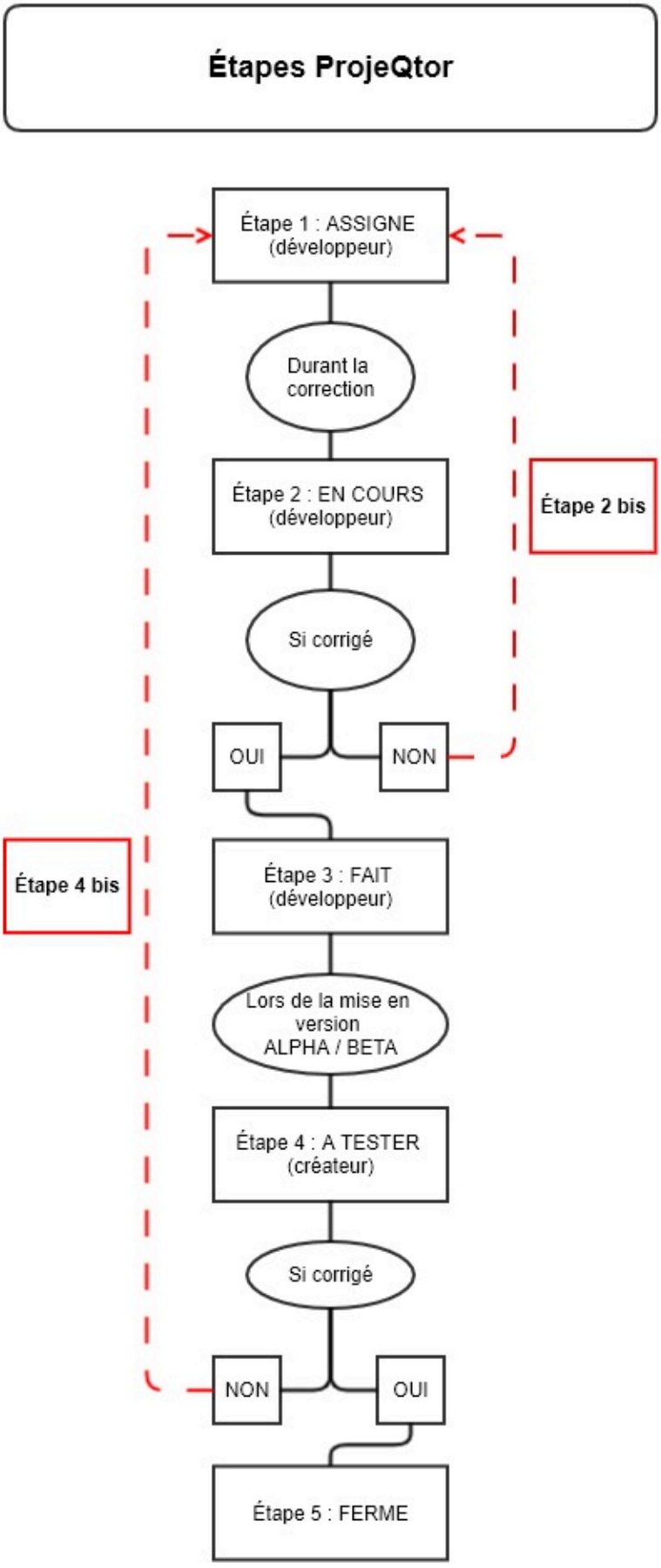
1. Est-ce que personne (ou presque) ne peut travailler normalement ? → **Bloquant**
2. Y a-t-il un contournement simple et sans risque ? → Non = **Majeur**
3. L'impact est limité et le contournement facile ? → **Mineur**
4. Le logiciel fonctionne correctement mais pourrait être mieux ? → **Amélioration / Évolution**

<note tip> En cas de doute entre deux niveaux → choisir le plus élevé et laisser l'assistance ajuster après investigation. </note>

Le piège le plus courant

<note warning> **“Urgent” n'est pas un niveau de gravité.** C'est une émotion. Le niveau se base sur l'impact métier réel et l'existence d'un contournement, pas sur la pression ressentie par l'utilisateur. </note>

Etapes de traitement


Étapes ProjeQtor
Étape 1 : ASSIGNE (développeur)
Durant la correction
Étape 2 : EN COURS (développeur)
Étape 2 bis
Si corrigé
OUI
NON
Étape 4 bis
Étape 3 : FAIT (développeur)
Lors de la mise en version ALPHA / BETA
Étape 4 : A TESTER (créateur)
Si corrigé
NON
OUI
Étape 5 : FERME

	Responsable	**Type de ticket**	**Urgence**
Niveau 1 (plus urgent)	Nicolas	Problème sur base Client	Bloquant
Niveau 2	Nicolas	Problème sur base Client	Urgent
Niveau 3	Nicolas	Problème sur base Client	Non Urgent
Niveau 4	Nicolas	Dysfonctionnement	Bloquant
Niveau 5	Nicolas	Dysfonctionnement	Urgent
Niveau 6	Nicolas	Dysfonctionnement	Non Urgent

Etape 1 : Référencement d'une demande

Procédure vis-à-vis du client

La première action à faire vis-à-vis du client est d'identifier sur quelle version du logiciel il travaille. S'il n 'est pas sur la dernière version disponible, il lui est demandé de bien vouloir le mettre à jour, puis de vérifier si le problème persiste.

AUCUNE assistance sur les fonctionnalités n'est faite sur les versions antérieures du logiciel à partir du moment où une mise à jour est publiée. Seules l'assistance à l'évolution de la version et/ou l'aide à la correction conséquente à un problème dû à une version antérieure seront prises en compte.

Chaque cas sera étudié et une réponse sera émise au client dans tous les cas. Celle-ci sera obligatoirement tracée au travers de la procédure d'assistance

Cas où une correction est nécessaire

Si lors d'une demande d'assistance, d'une formation ou de test, un problème nécessitant une action sur le code ou les fichiers annexes est repéré, la procédure est la suivante :

Cas 1 : Problème impliquant uniquement une correction du code

1. Lancer une session Projeqtor
    - Sélectionner le Projet et la sous-version correspondant à la prochaine version [PROC-Nflog-5 : gestion des versions]
    - Dans “Travail\Bugs”, créer un nouveau Ticket et le remplir comme indiqué plus loin en tenant compte des spécificités listées ci-dessous :
        - **Projet = Prochaine sous-version** (par défaut si le projet est sélectionné)
        - **Type de ticket = “Anomalie / Bug issue - origine client” ou “Anomalie / Bug issue - origine autre”**
        - **Référence externe = Numéro de ticket OTRS s'il existe.** Exemple :*Ticket#2015022310000059*
        - **Responsable = vide**
    - NB : Dans le cas où la demande est un doublon par rapport à une demande existante, on se contentera de compléter le ticket existant en ajoutant la référence à la demande du client.
2. Revenir sur le ticket OTRS

- Envoyer une réponse à l'utilisateur
- Faire une note en indiquant en titre “PROJEQTOR –BUGS#XX” et le texte de votre choix

Cas 2 : Problème impliquant uniquement une décision du groupe de travail EPUdF-Logeas

1. Lancer une session Projeqtor
   - Sélectionner le Projet et la sous-version correspondant à la prochaine version [certif:procedure:develop:gestionversion|PROC-Nflog-5 : gestion des versions]
   - Dans “Journaux des revues\Questions”, créer un nouveau Ticket et le remplir comme indiqué plus loin en tenant compte des spécificités listées ci-dessous :
     - **Projet = Prochaine sous-version** (par défaut si le projet est sélectionné)
     - **Type de ticket = “Assistance - demande d'explication EPUdF” ou “Assistance - demande de correction EPUdF“**
     - **Référence externe = Numéro de ticket OTRS s'il existe.** Exemple :*Ticket#2015022310000059*
     - **Responsable = “Jean-Marc DEGON & Michel HAFFNER”**
   - NB : Dans le cas où la demande est un doublon par rapport à une demande existante on se contentera de compléter le ticket existant en ajoutant la référence à la demande du client.
2. Revenir sur le ticket OTRS
   - Envoyer une réponse à l'utilisateur
   - Faire une note en indiquant en titre “PROJEQTOR – TICKET #XX” et le texte de votre choix (obligatoire, mais sans intérêt ..)

Cas 3 : Demande d'évolution du logiciel

1. Lancer une session Projeqtor
   - Sélectionner le Projet et la sous-version correspondant à la prochaine version [PROC-Nflog-5 : gestion des versions]
   - Dans “Travail\Evolutions”, créer un nouveau Ticket et le remplir comme indiqué plus loin en tenant compte des spécificités listées ci-dessous :
     - **Projet = Prochaine sous-version** (par défaut si le projet est sélectionné)
     - **Type de ticket = “Assistance - demande d'évolution ''**
     - **Référence externe = Numéro de ticket OTRS s'il existe.** Exemple :*Ticket#2015022310000059*
     - **Responsable = vide**
   - NB : Dans le cas où la demande est un doublon par rapport à une demande existante, on se contentera de compléter le ticket existant en ajoutant la référence à la demande du client.
2. Revenir sur le ticket OTRS
   - Envoyer une réponse à l'utilisateur
   - Faire une note en indiquant en titre “PROJEQTOR – TICKET #XX” et le texte de votre choix

Information générique sur la saisie d'un ticket

Description	Projet	Selon le cas
Description	Type de ticket	Selon le cas
Description	Nom	**Donner un titre compréhensible**

Description	Référence externe	Selon le cas
Description	Urgence	Non utilisé
Description	Date de création	Automatique
Description	Émetteur	Automatique, nom de l'utilisateur connecté NB : Chaque utilisateur doit avoir une session distincte
Description	Demandeur	Non utilisé
Description	Origine	Non utilisé
Description	Ticket en doublon	Utilisation postérieure
Description	Contexte	A remplir si connu
Description	Produit	Indiquez le produit
Description	Version d'origine	Indiquer la version du produit où a été constaté le problème, si connue
Description	Description	Indiquer le détail du problème
Traitement	Activité de planning	Ne pas utiliser
Traitement	Etat	Mettre “Assigné”
Traitement	Responsable	Assigner le ticket à la personne qui doit s'en occuper, ce qui envoie un mail à la personne
Traitement	Criticité	Selon le cas
Traitement	Priorité	Selon le cas
Traitement	Échéance initiale,; actuelle	Non utilisé
Traitement	Travail estimé, restant	Non utilisé
Traitement	Pris en charge, Fait, Clos	Automatique
Traitement	Version cible	Non utilisé
Avancement		Sans objet
Élément prédécesseur / Successeur		Généralement Sans Objet
Élément liés		Ne pas utiliser
Fichiers attachés		Ajouter les documents, bases ... qui permettent de mettre en lumière le problème et de tester la solution
Notes		Permet d'ajouter des éléments textuels

Etape 2 : Prise en charge par le dev d'une demande

Lancer une session Projeqtor

1. Sélectionner le ticket correspondant
2. Le basculer en “En cours”
3. Initialiser le **Traitement\Responsable** s'il ne l'est pas

Etape 2bis : Prise en charge par le dev d'une demande (Echec)

Lancer une session Projeqtor

1. Sélectionner le ticket correspondant : **“Travail\Bugs“**
    - Compléter le ticket dans la partie “**Traitement**”:
        - Initialiser le **Traitement\Responsable** s'il ne l'est pas

Activité de planning	Initialisé à la création du ticket
Etat	**Mettre “Assigné”**
Responsable	**Changer le responsable pour l'assigner à quelqu'un d'autre**
Criticité	Non utilisé
Priorité	Non utilisé
Échéance initiale,; actuelle	Non utilisé
Travail estimé, restant	Non utilisé
Pris en charge, Fait, Clos	Automatique
Version cible	Non utilisé
Résultat	vide

- Enregistrer
- Lier les documents, si il y a lieu (base de test, copie ...)
- Mettre une note si il y a lieu

Etape 3 : Prise en charge par le dev d'une demande (Succès)

Au niveau développement :

1. Effectuer la correction, la tester
2. Publier le code sur SVN, en indiquant dans le commentaire le tag du ticket (id=#20)

Au niveau ProjeQtOr

Lancer une session Projeqtor

1. Sélectionner le ticket correspondant : **“Travail\Bug“**
    - Compléter le ticket dans la partie “**Traitement**”:
        - Initialiser le Traitement\Responsable s'il ne l'est pas

Activité de planning	Initialisé à la création du ticket
Etat	**Mettre “FAIT”**
Responsable	**ne pas changer**
Criticité	Non utilisé
Priorité	Non utilisé
Échéance initiale,; actuelle	Non utilisé
Travail estimé, restant	Non utilisé

Pris en charge, Fait, Clos	Automatique
Version cible	Non utilisé
Résultat	vide

- Enregistrer
- Lier les documents, si il y a lieu (base de test, copie ...)
- Mettre une note si il y a lieu

Etape 4 : Mise en place d'une version de test (alpha ou beta)

1. Lancer une session Projeqtor
2. Dans les tickets sur le projet concerné, cliquer sur le bouton à droite “Mise à jour multiple”
3. Sélectionner le(s) ticket(s)/activité(s) correspondant à un état “FAIT”
4. Initialiser le Traitement\Responsable au créateur du ticket
5. Les basculer en “A TESTER”

From:
https://wiki-logeas.fr/certif/ - **dokuwiki-certif**

Permanent link:
https://wiki-logeas.fr/certif/doku.php?id=certif:procedure:develop:gestionprojeqtor&rev=1780413958

Last update: **2026/06/02 17:25**