Process Release & Deploy

# Présentation

## Problématiques

Il a été constaté plusieurs problématiques en ce qui concerne la préparation et la livraison des montées de version client que la solution présente propose de solutionner :

* Communication interne : Ce qui empêche la réactivité de certains services en cas de problème remonté par le client ou l’anticipation de problèmes pouvant être prévus lorsqu’une communication interservices est opérée
* Communication externe : Pas de visibilité sur les nouveautés ajoutées au produit à chaque montée de version (Release note)
* Planification : Possibilité par d’autres service de pouvoir consulter le journal de livraison afin d’éviter de planifier des actions multi-services à des moments d’indisponibilités des équipes concernées
* Accès aux informations : La possibilité pour les autres services d’accéder au planning de chaque équipe permet simplement de pouvoir planifier des actions parallèles en ayant la possibilité de le faire rapidement sans avoir à chercher les informations (Planning des différents intervenants / mails / discussion orale / etc…)

# Solution proposée

## Intégration

La solution proposée devra être intégrée à l’ERP existant chez NCI, le TAF.

## Minimum fonctionnel

Il existe actuellement dans la centrale de gestion des projets, une fonctionnalité non utilisée nommée « Journal des livraison ». Cette fonctionnalité ne sera plus utilisée et pourra être retirée de ce module.

Un module dédié aux gestions des livraisons devra alors être créé.

Dans l’entête du module, seront ajoutés les filtres suivants :

* Filtre service
* Filtre produit
* Filtre client
* Filtre de dates de livraison prévue

La vue devra contenir les informations suivantes :

* Client
* Produit
* Date prévisionnelle
* Date de release
* Date de communication au client
* Tag SQL
* Tag de version
* Statut
* Calendrier d’évènements

Fonctionnellement, il devra être possible de :

* Pouvoir créer / modifier, ou supprimer un enregistrement du journal de livraisons
* Pouvoir spécifier une livraison dans le passé et dans l’avenir, avec une durée théorique
* Pouvoir spécifier la date de communication auprès du client afin de savoir s’il a été prévenu
* Ajouter une observation libre dans la saisie d’un nouvel enregistrement
* Ajouter un complément à la release note dans la saisie d’un nouvel enregistrement
* Pouvoir renseigner un tag de version et un tag SQL
* Proposer choisir de créer deux « DE » (Demandes d’Evolution) via une case à cocher :
	+ Une pour préparation de la livraison
	+ Une pour la livraison
	+ Ouvrir la possibilité d’affecter une personne de l’équipe à chacune de ces tâches
* Pouvoir gérer 3 statuts sur un enregistrement du journal de livraisons
	+ A la création, si la date de livraison est dans l’avenir, marquer l’enregistrement de la livraison comme « Non déployé »
	+ Pouvoir basculer le statut « En cours » pour spécifier qu’un déploiement est en train de se faire
	+ Pouvoir basculer le statut « Déployé » pour marquer un déploiement comme finalisé
	+ A ma création, si la date de livraison est passée ou égale à la date du jour, marquer automatiquement l’enregistrement comme « Déployé »
* Créer un champ au niveau « équipe » permettant de renseigner l’adresse mail de l’équipe
* En termes de flux de communication, à la création d’une livraison prévisionnelle, un mail sera envoyé automatiquement aux équipes concernées par le client. (Exemple de Noyon : Infrastructure, Web, Transport, Gescom, Logistique, Hotline)
* En parallèle, un rendez-vous devra être envoyé à chacune des personnes concernées sur son compte Exchange
* Toujours en termes de communication, lorsqu’un enregistrement est pointé comme livré (Automatiquement ou manuellement) envoyer un mail automatiquement aux équipes concernées par le client. (Exemple de Noyon : Infrastructure, Web, Transport, Gescom, Logistique, Hotline)
* Dans les deux cas précédents, pouvoir inscrire adresses e-mail manuellement avant validation
* Au niveau algorithmique, pour éviter de spammer les équipes des clients multi-produits, il faudra vérifier si un mail n’a pas déjà été envoyé pour avertir les différentes équipes pour la date saisie lorsqu’un enregistrement est ajouté
* Il devra être possible de spécifier des équipes actives en complément de l’équipe concernées par la livraison en cours
* Chaque personne concernée activement par une livraison ou faisant partie de l’équipe concernée par la livraison, recevra un rendez-vous Exchange avec une importance haute.

## Maquette

Les maquettes ci-dessous ne sont pas contractuelles, les chartes graphiques et bonnes pratiques appliquées au TAF seront appliquées dans le cadre de ce développement.

Voici la maquette de ceux à quoi devra ressembler le minimum fonctionnel de ce nouveau module :



Voici le même écran avec des infobulles permettant de fournir des informations complémentaires aux utilisateurs :



Lors de l’ajout d’un enregistrement dans le journal (Bouton + entre le journal, et le calendrier), le formulaire suivant apparaitra :



## Release note

Afin de pouvoir automatiser la construction de release note par le biais d’une extraction, il va falloir créer une nouvelle zone prévue à cet effet dans les DE :



A la fermeture d’une DE, pouvoir également modifier le commentaire client :



(Attention, bien spécifier que ce sera l’observation apparente sur la note de version)

Il devra donc être communiqué aux différents développeurs, le fait que ce commentaire soit indispensable afin de compiler une « release note » sur la base d’informations sûres.

La même fonctionnalité sera accessible dans la gestion des statistiques du service client, afin de pouvoir faire ressortir dans des notes de versions, des corrections de bogues (Bug fixes) :



Pour le cas des demandes d’assistance, il sera également possible de spécifier s’il s’agit d’un point spécifique au client ou s’il doit apparaitre sur toutes les notes de versions.

## Extraction de la release note

Il existe actuellement un module d’extraction présent dans module Kanban, dans lequel il faudra ajouter la notion de client. Il ne sera plus utile. L’extraction des releases notes devra être faite depuis l’outil de gestion des déploiements.

Cette extraction, sera composée de 3 onglets, et permettra de faire ressortir au format csv, les informations suivantes :

* Un onglet « Features » avec :
	+ Produit
	+ N° de la DE
	+ Note de version
	+ Nature du développement (Spécifique ou Commun)
	+ Date de la demande Client ou de Création
	+ Date de fin de réalisation

Il faudra par défaut trier par produit, date de la demande, Nature et N° de DE.

Seules les DE archivées (Donc contrôlées et validées par le chef de produit) ressortiront. Lorsque le client sera sélectionné, seules les DE communes et les DE le concernant seront affichées.

Les DE de type « Tâches » n’apparaitront pas sur l’extraction.

* Un onglet « Bug fixes » avec :
	+ Produit
	+ N° de la DA
	+ Note de version
	+ Nature du développement (Spécifique ou Commun)
	+ Date de la dernière action de résolution

Il faudra par défaut trier par produit, date d’action, Nature, N° de DA.

Toutes les DA ayant subi au moins une action de résolution pendant la période précédente à la release en cours ressortiront. Lorsque le client sera sélectionné, seules les DA avec note de version commune et les DA avec note de version concernant le client seront affichées.

* Un onglet « Todo » avec :
	+ Produit
	+ N° de la DE
	+ Commentaire de création
	+ Date de la création

Il faudra par défaut trier par produit, date de création, N° de DE.

## Avantages de ces nouveaux process

* Centraliser un planning de livraison
* Simplification / semi-automatisation de l’élaboration des notes de version
* Mise en place d’une convention de versioning des livraisons
* Ouverture à l’intégration automatique des montées de version dans le TAF

# Evolutions possibles

En interconnectant des outils avec des webservices développés dans l’outil de gestion des livraisons, il sera possible de :

* Automatiser le changement de statut avec un outil de déploiement
* Automatiser le renseignement des notes de version avec un outil de versioning (Git)
* Automatiser l’extraction et la mise en forme des notes de version avec des outils tiers (Markdown)