
Projet Qualité Logicielle : Modèles de l'environnement

Équipe H4314 – 9 février 2014

Chef de projet

Jean-Marie COMETS

Membres de l'équipe

Franck MPEMBA BONI: responsable qualité

Pierre TURPIN

Samuel CARENSAC

Grégoire CATTAN

Van PHAN HAU

Iler VIRARAGAVANE

Table des matières

1	Système balance	2
1.1	Système Web	2
1.2	Système externe	2
1.3	Capteur Balance	3
2	Drone	3
2.1	tapisRoulant	3
2.2	trappe	3
2.3	systemeRonde	4
2.4	listeColis	4
2.5	systemeReceptacle	4
2.6	balance	4
2.7	timer	4
3	Réceptacle	5
3.1	PaveNum	5
3.2	capteurBadge	5
3.3	diode	5
3.4	panneauAffichage	6
3.5	supportDrone	6
3.6	porteLaterale	6
3.7	panneauSuperieur	6
3.8	capteurRFID	7
3.9	systemeDrone	7
3.10	systemeClient	7
4	Système Ronde	8
4.1	ecranSupervision	8
4.2	btnVisualiserDrones	8
4.3	btnVisualiserTournées	9
4.4	boiteMail	9
4.5	interfaceControleClient	9
4.6	dronePhysique	9
4.7	alarme	9
4.8	BD	10
5	Système Web	10
5.1	BtnAssignBalance	11
5.2	BtnConfigBalance	11
5.3	BtnScan	11
5.4	BtnVoirCommande	11
5.5	BtnValiderCommande	11
5.6	InterfaceCatalogue	11
5.7	SystèmeBalance	11
5.8	ListeCommande	12
5.9	PanneauInformation	12

1 Système balance

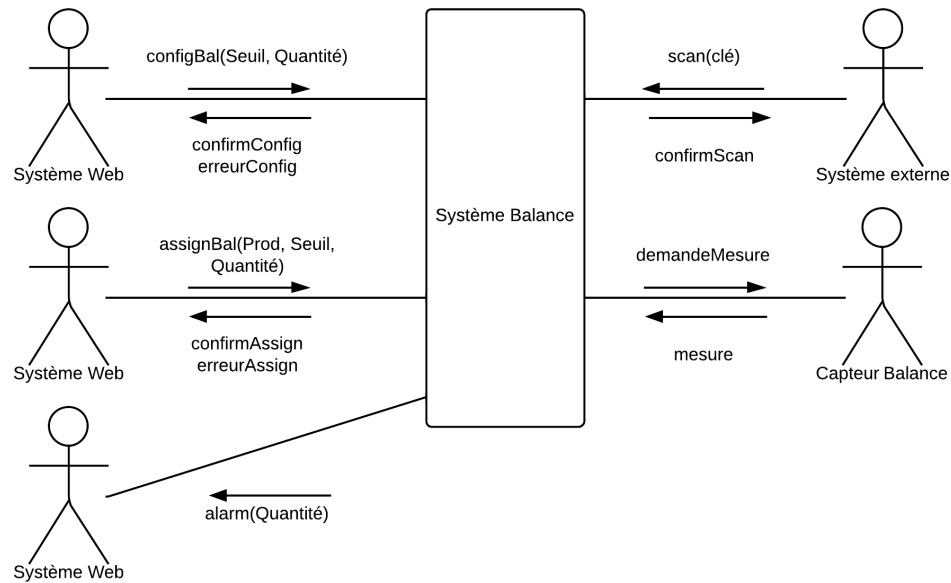


FIGURE 1 – Schéma du modèle de l'environnement "Système balance"

1.1 Système Web

Contrôleur principal côté client.

Acteur du CdU : Utilisateur

- *configBal(Seuil, Quantité)* : Demande de configuration du seuil de quantité minimale *Seuil* et de la quantité de produit à racheter *Quantité*.
CdU concerné : *aucun*
- *confirmConfig()* : Confirmation de la configuration réussite du système balance
CdU concerné : *aucun*
- *erreurConfig()* : Indication qu'une erreur à eu lieu lors de la configuration du système balance (en général cela se produit lorsque les paramètres ne sont pas valides)
CdU concerné : *aucun*
- *assignBal(Produit, Seuil, Quantité)* : Demande d'assignement d'un nouveau produit *Produit* avec une nouvelle configuration en accord avec le produit *Seuil* et *Quantité*.
CdU concerné : *aucun*
- *confirmAssign()* : Confirmation de l'assignement réussi du système balance
CdU concerné : *aucun*
- *erreurAssign()* : Indication qu'une erreur à eu lieu lors de l'assignement du nouveau produit (en général cela se produit lorsque les paramètres ne sont pas valides)
CdU concerné : *aucun*
- *alarm(Quantité)* : Envoie une alarme au système web pour notifier le manque d'un produit.
CdU concerné : *aucun*

1.2 Système externe

Contrôleur principal côté client.

Acteur du CdU : Système web

- *scan()* : Demande d'identification de la part d'un contrôleur externe (liaison entre les deux toujours pas établie). Répond par une confirmation si la clé du système balance correspond à celui du contrôleur.
CdU concerné : *aucun*
- *confirmScan()* : Confirme l'existence du système balance au système contrôleur.
CdU concerné : *aucun*

1.3 Capteur Balance

Capteur mesurant le poids.

Acteur du CdU : Système Balance

- *demandeMesure()* : Demande au capteur de fournir la valeur de la mesure.
CdU concerné : *aucun*
- *mesure()* : Le capteur renvoie la valeur de la mesure. La balance la traite en la transformant en grandeur physique (valeur utilisable).
CdU concerné : *aucun*

2 Drone

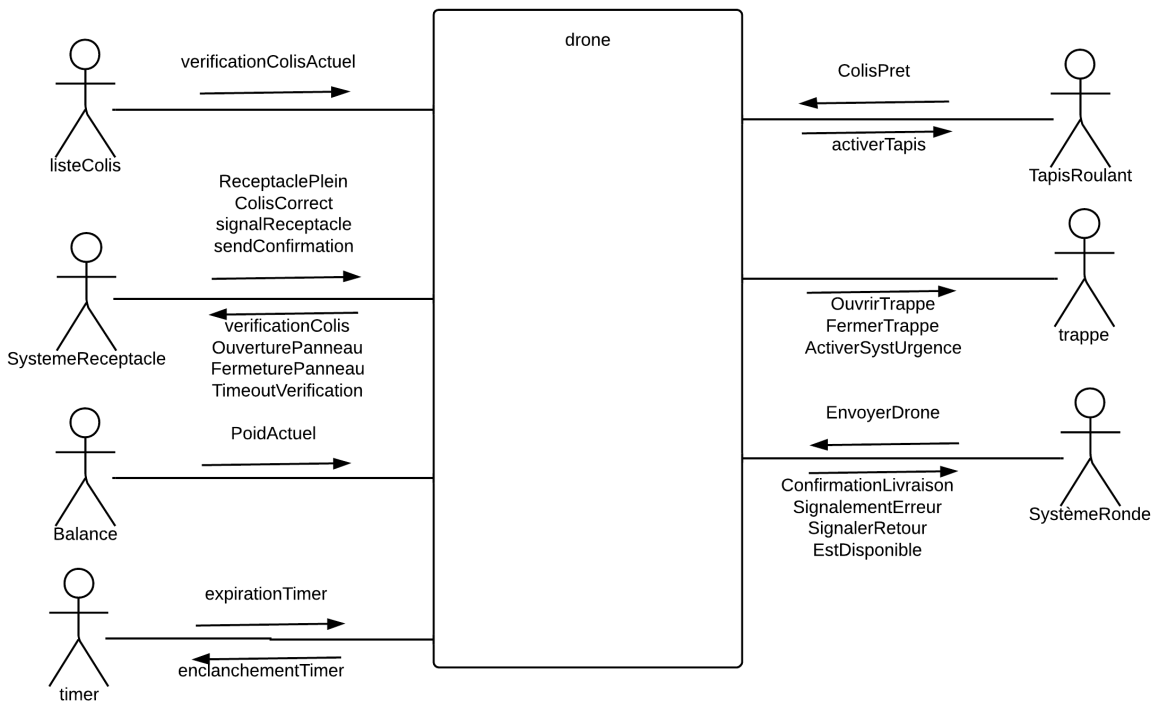


FIGURE 2 – Schéma du modèle de l'environnement "Drone"

2.1 tapisRoulant

Acteur du CdU :

- *colisPret()* :
CdU concerné : *aucun*
- *activerTapis(e)* :
CdU concerné : *aucun*

2.2 trappe

Acteur du CdU :

- *OuvrirTrappe()* :
CdU concerné : *aucun*
- *FermerTrappe()* :
CdU concerné : *aucun*
- *ActiverSystUrgence()* :
CdU concerné : *aucun*

2.3 systemeRonde

Acteur du CdU :

- *EnvoyerDrone(itineraire)* :
CdU concerné : *aucun*
- *ConfirmationLivraison()* :
CdU concerné : *aucun*
- *SignalementErreur(msg)* :
CdU concerné : *aucun*
- *SignalerRetour()* :
CdU concerné : *aucun*
- *EstDisponible(e)* :
CdU concerné : *aucun*

2.4 listeColis

Acteur du CdU :

- *verificationColisActuel(idColis)* :
CdU concerné : *aucun*

2.5 systemeReceptacle

Se référer à la section 3.

2.6 balance

Acteur du CdU :

- *PoidActuel(n)* :
CdU concerné : *aucun*

2.7 timer

Acteur du CdU :

- *expirationTimer(idTimer)* :
CdU concerné : *aucun*
- *enclenchementTimer(timer)* :
CdU concerné : *aucun*

3 Réceptacle

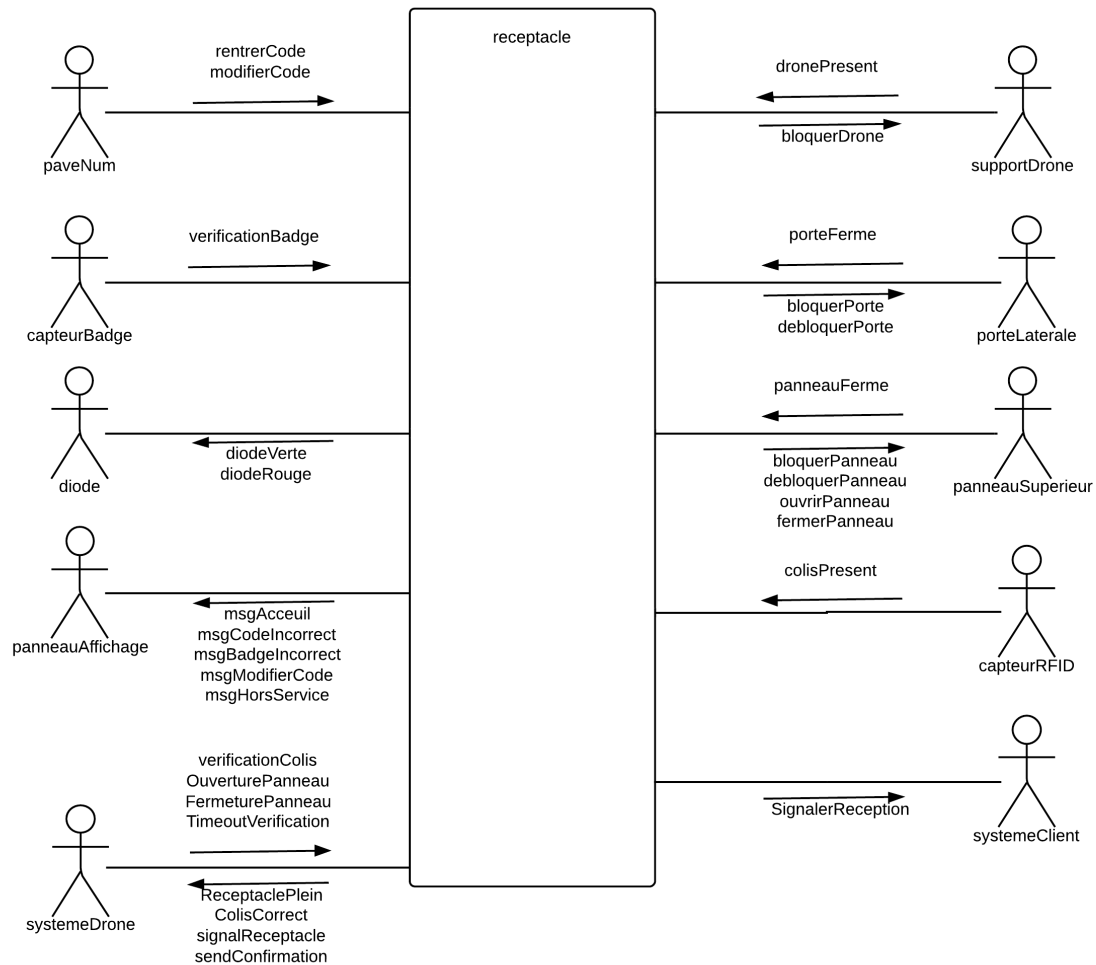


FIGURE 3 – Schéma du modèle de l'environnement "Réceptacle"

3.1 PaveNum

Pavé numérique type standard, composé de chiffres (valeur allant de 0 à 9), d'un bouton "valider", et d'un bouton "annuler".

Acteur du CdU :

- *rentrerCode*(*n*) : Le code *n* est vérifié.
CdU concerné : *aucun*
- *modifierCode*(*n*) : Le code *n* est fixé en tant que nouveau code.
CdU concerné : *aucun*

3.2 capteurBadge

Capteur de badge pour détecter la présence du badge de l'utilisateur (ou celui du technicien).

Acteur du CdU :

- *verificationBadge*(*b*) : La valeur du badge est vérifiée.
CdU concerné : *aucun*

3.3 diode

Diode pouvant prendre une couleur verte ou rouge pendant un temps donné.

Acteur du CdU :

- *diodeVerte*() : Demande à la diode de devenir verte pour 3 secondes.
CdU concerné : *aucun*

- *diodeRouge()* : Demande à la diode de devenir rouge pour 3 secondes.
CdU concerné : *aucun*

3.4 panneauAffichage

Panneau d'affichage pour l'affichage d'informations générales sur l'état du système.

Acteur du CdU :

- *msgAccueil()* : Demande d'affichage du message d'accueil.
CdU concerné : *aucun*
- *msgCodeIncorrect(e)* : Si *e* est vrai, demande d'affichage d'un message indiquant que le code rentré est incorrect. Sinon, demande du réceptacle d'effacer le message.
CdU concerné : *aucun*
- *msgBadgeIncorrect(e)* : Similaire à *msgCodeIncorrect(e)*.
CdU concerné : *aucun*
- *msgModifieurCode(e)* : Similaire à *msgCodeIncorrect(e)*. L'effacement est suivi d'une demande d'affichage d'un message demandant la fermeture.
CdU concerné : *aucun*
- *msgHorsService(e)* : Similaire à *msgCodeIncorrect(e)*.
CdU concerné : *aucun*

3.5 supportDrone

Acteur du CdU :

- *dronePresent()* : Le support indique que le drone est présent.
CdU concerné : *aucun*
- *bloquerDrone(e)* : Si *e* est vrai, une demande est envoyée au support pour bloquer le drone. Sinon, une demande est envoyée au support pour relâcher le drone.
CdU concerné : *aucun*

3.6 porteLaterale

Porte latérale du réceptacle, utilisée par le client pour récupérer les colis. Ne peut être ouverte que manuellement.

Acteur du CdU :

- *porteFerme()* : Signale au réceptacle que la porte vient d'être fermée.
CdU concerné : *aucun*
- *bloquerPorte()* : Demande du réceptacle de bloquer la porte latérale.
CdU concerné : *aucun*
- *dbloquerPorte()* : Demande du réceptacle de débloquer la porte latérale.
CdU concerné : *aucun*

3.7 panneauSuperieur

Panneau supérieur du réceptacle, utilisé par le drone pour déposer les colis. Peut être ouverte électroniquement ou manuellement.

Acteur du CdU :

- *panneauFerme()* : Signale au réceptacle que le panneau supérieur vient d'être fermé.
CdU concerné : *aucun*
- *bloquerPanneau()* : Demande du réceptacle de bloquer le panneau supérieur.
CdU concerné : *aucun*
- *debloquerPanneau()* : Demande du réceptacle de débloquer le panneau supérieur.
CdU concerné : *aucun*
- *ouvrirPanneau()* : Demande du réceptacle d'ouvrir du panneau.
CdU concerné : *aucun*
- *fermerPanneau()* : Demande du réceptacle de fermer le panneau supérieur.
CdU concerné : *aucun*

3.8 capteurRFID

Un capteur RFID permettant de vérifier la présence d'un colis à l'intérieur du réceptacle.

Acteur du CdU :

- *colisPresent()* : Le capteur indique s'il a détecté la présence d'un colis à l'intérieur du réceptacle.
CdU concerné : *aucun*

3.9 systemeDrone

Correspond au système du drone en liaison avec le réceptacle.

Acteur du CdU :

- *verificationColis(idDestinataire)* : Signal du drone demandant la vérification de la correspondance entre un colis et le réceptacle.
CdU concerné : *aucun*
- *ouverturePanneau()* : Signal du drone demandant l'ouverture du panneau supérieur du réceptacle.
CdU concerné : *aucun*
- *fermeturePanneau()* : Signal du drone demandant la fermeture du panneau supérieur du réceptacle.
CdU concerné : *aucun*
- *timeoutVerification()* : Signal du drone indiquant que celui-ci n'a pas reçu la confirmation de l'action demandée dans le temps voulu.
CdU concerné : *aucun*
- *receptaclePlein()* : Signal du réceptacle indiquant au drone que le réceptacle est plein.
CdU concerné : *aucun*
- *colisCorrect(e)* : Si *e* est vrai, signale que le colis correspond bien à ce réceptacle. Sinon, signale que le colis ne correspond pas à ce réceptacle.
CdU concerné : *aucun*
- *signalReceptacle()* : Signal du réceptacle pour indiquer au drone la position du réceptacle pour que celui-ci se pose.
CdU concerné : *aucun*
- *sendConfirmation(msg)* : Signal du receptacle pour indiquer qu'une action a bien été réalisée. Cette action est indiqué par le message *msg* (sous la forme d'une énumération).
CdU concerné : *aucun*

3.10 systemeClient

Acteur du CdU :

- *signalerReception()* : Signal du réceptacle au système client pour indiquer qu'il y a un colis à récupérer dans le réceptacle.
CdU concerné : *aucun*

4 Système Ronde

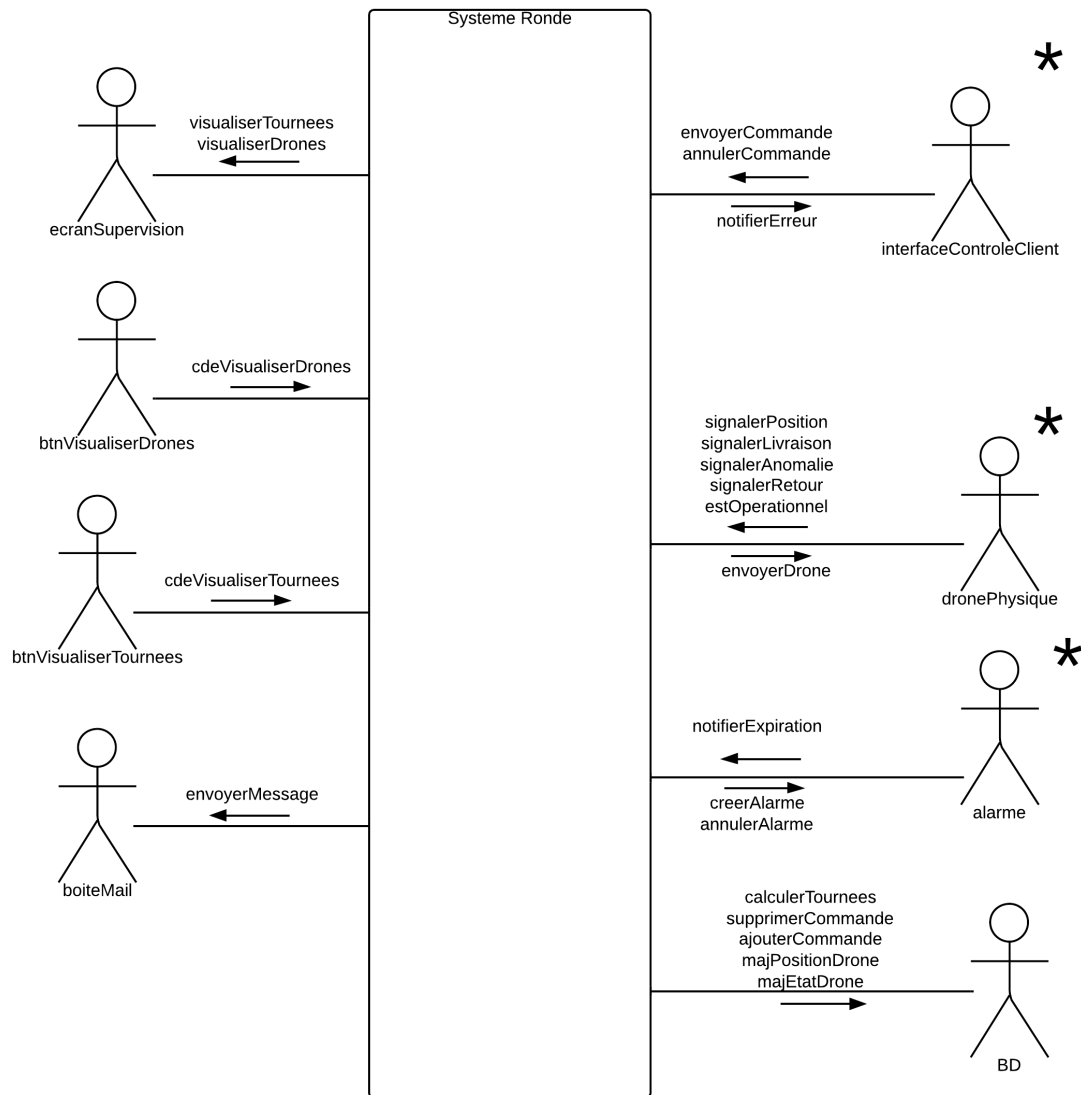


FIGURE 4 – Schéma du modèle de l'environnement "Système Ronde"

4.1 ecranSupervision

Un écran d'affichage où l'on peut visualiser les tournées, leur avancement, et la position des drones qui leur sont rattachées.

Acteur du CdU : Superviseur

- *visualiserDrone()* : Visualise la position des drones sur l'écran, ainsi que leur état (marche, alerte).
CdU concerné : *Superviser ronde*
- *visualiserTournée()* : Visualise les tournées et leur avancement sur l'écran
CdU concerné : *Superviser ronde*

4.2 btnVisualiserDrones

Un bouton de l'interface qui permet d'afficher les drones sur *ecranSupervision*.

Acteur du CdU : Superviseur

- *cdeVisualiserDrones()* : Le bouton a été actionné.
CdU concerné : *Superviser ronde*

4.3 btnVisualiserTornees

Un bouton de l'interface qui permet d'afficher les tournées sur *ecranSupervision*.

Acteur du CdU : Superviseur

- *cdeVisualiserTornees()* : Le bouton a été actionné.
CdU concerné : *Superviser ronde*

4.4 boiteMail

Une application qui gère l'affichage ainsi que le traitement des messages.

Acteur du CdU : Superviseur

- *envoyerMessage(idDrone, Message)* : Le système envoie le message *Message* venant du drone *idDrone* à la *boiteMail* (quitter entrepôt, en tournée, livraison impossible, alerte, ...).
CdU concerné : *Superviseur ronde*

4.5 interfaceControleClient

L'interface de l'application permettant de réceptionner les messages d'un client.

Acteur du CdU : Client

- *envoyerCommande(Commande)* : Envoi de la commande *Commande* au système ronde.
CdU concerné : *Livrer une tournée*
- *annulerCommande(Commande)* : Demande l'annulation de la commande *Commande*.
CdU concerné : *Livrer une tournée*
- *notifieErreur(Message)* : Préviens le client que sa demande ne peut être satisfaite (absence de produit demandé, annulation survenue trop tard, ...), en spécifiant le message *Message*.
CdU concerné : *Livrer une tournée*

4.6 dronePhysique

Un drone du système ronde

Acteur du CdU : Drone

- *signalerPosition(Position)* : Envoie les coordonnées GPS *Position* du drone au système.
CdU concerné : *Livrer une tournée*
- *signalerLivraison(Commande)* : Signale au système que la commande *Commande* a bien été livrée.
CdU concerné : *Déposer une commande*
- *signalerAnomalie(Anomalie)* : Signale au système une anomalie (panne, attaque, impossibilité de livrer une commande, ...). Une anomalie *Anomalie* peut être déclenchée manuellement si elle est constatée par un technicien du système.
CdU concerné : *Livrer une tournée*
- *signalerRetour()* : Le drone signale au système qu'il est bien revenu.
CdU concerné : *Livrer une tournée*
- *estOperationnel()* : Le drone signale au système qu'il est disponible (à l'entrepôt et en bon état).
CdU concerné : *Livrer une tournée*
- *envoyerDrone(Tournee)* : Le système demande au drone de commencer la tournée *Tournee*.
CdU concerné : *Livrer une tournée*

4.7 alarme

Une alarme associée à une tournée, qui prévient le système si la tournée risque de ne pas être livrée à temps.

Acteur du CdU : Alarme

- *notifierExpiration(Tournee)* : Signale au système qu'un élément de la tournée *Tournee* risque de ne pas être livré à temps si l'on attend plus longtemps.
CdU concerné : *Livrer une tournée*
- *creerAlarme(Tournee, Temps)* : Le système associe une alarme à la tournée *Tournee* et demande son expiration dans *Temps* heures.
CdU concerné : *Livrer une tournée*
- *annulerAlarme()* : Détruit l'alarme avant qu'elle n'expire.
CdU concerné : *Livrer une tournée*

4.8 BD

Base de données contenant toutes les commandes en cours de livraison (ou en attente) ainsi que leur tournée correspondante.

Acteur du CdU : Ronde

- *calculerTournée()* : Recalcule les tournées (et éventuellement crée les nouvelles).
CdU concerné : *Livrer une tournée*
- *ajouterCommande(Commande)* : Ajoute la commande *Commande* dans la *BD*.
CdU concerné : *Livrer une tournée*
- *supprimerComande(Commande)* : supprimer des commandes dans la *BD*.
CdU concerné : *Livrer une tournée*
- *majPositionDrone(idDrone, Position)* : Met à jour la position GPS à *Position*, pour le drone identifié par *idDrone*, dans la *BD*.
CdU concerné : *Livrer une tournée*
- *majEtatDrone(idDrone, Anomalie)* : Met à jour l'état, pour le drone identifié par *idDrone*, à l'état *Anomalie*.
CdU concerné : *Livrer une tournée*

5 Système Web

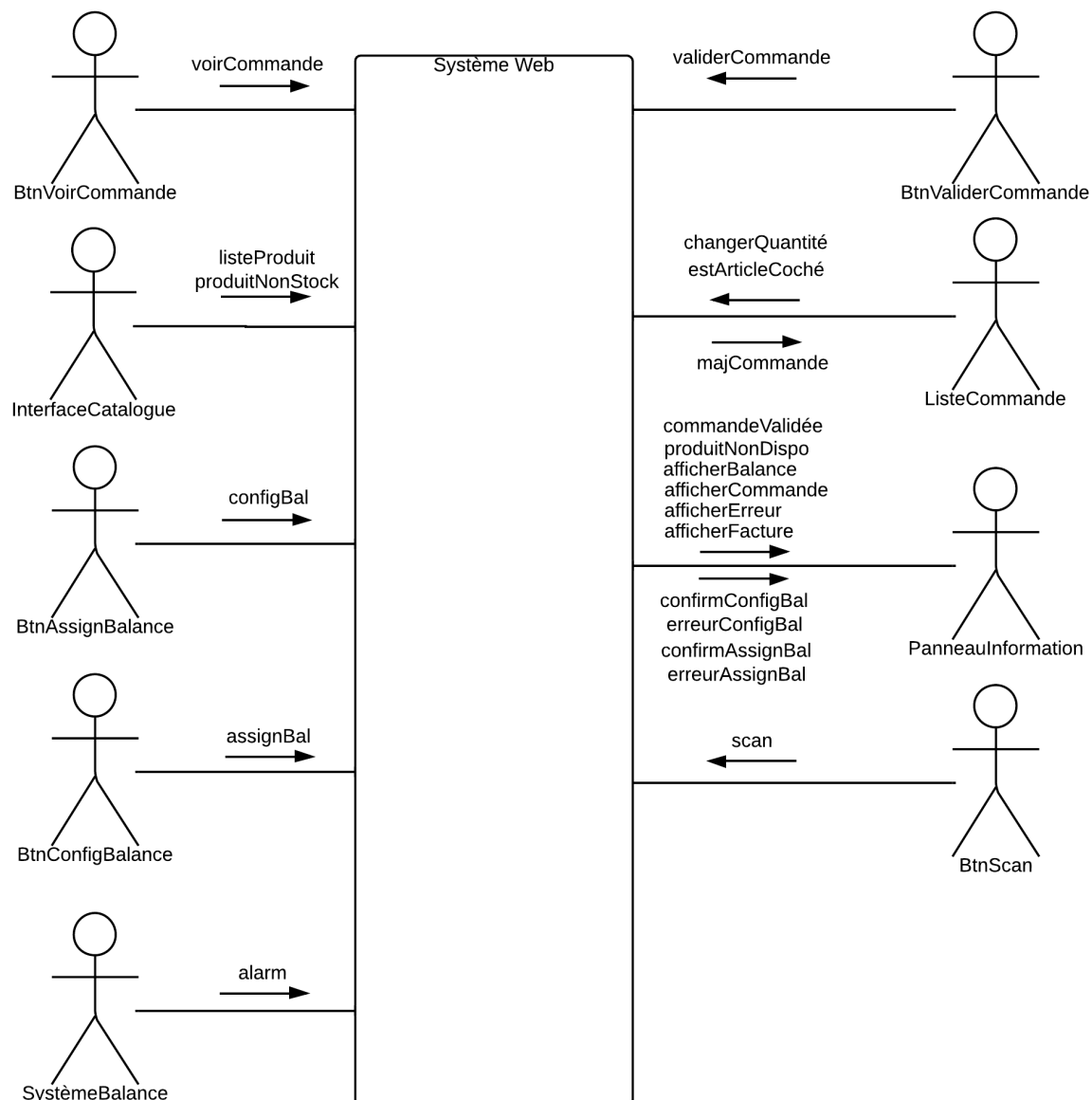


FIGURE 5 – Schéma du modèle de l'environnement "Système Web"

5.1 BtnAssignBalance

Un bouton pour assigner un produit à une balance existante.

Acteur du CdU : Utilisateur

- *assignBal(bal, art, seuil, qte)* : Le bouton a été activé pour la balance *bal* avec l'article *art* et comme configuration le couple (*seuil, qte*), respectivement le seuil et la quantité.
CdU concerné : *Configurer les balances*

5.2 BtnConfigBalance

Un bouton pour configurer une balance

Acteur du CdU : Utilisateur

- *configBal(bal, seuil, qte)* : Le bouton a été activé pour la balance *bal* avec la configuration (*seuil, qte*).
CdU concerné : *Configurer les balances*

5.3 BtnScan

Un bouton pour identifier les balances associées au client

Acteur du CdU : Utilisateur

- *scan()* : Demande d'identification de la part d'un contrôleur externe (liaison entre les deux toujours pas établie). Répond par une confirmation si la clé du système balance correspond à celui du contrôleur.
CdU concerné : *Configurer les balances*

5.4 BtnVoirCommande

Un bouton pour afficher la liste des commandes courante associée à l'utilisateur.

Acteur du CdU : Utilisateur

- *voirCommande()* : Le bouton à été actionné.
CdU concerné : *Valider une commande*

5.5 BtnValiderCommande

Un bouton pour valider la liste des commandes courante associée à l'utilisateur.

Acteur du CdU : Utilisateur

- *validerCommande()* : Le bouton à été actionné.
CdU concerné : *Valider une commande*

5.6 InterfaceCatalogue

Interface client pour ajouter un produit à la liste des commandes de l'utilisateur.

Acteur du CdU : Utilisateur, Controleur

- *listeArticle()* : La liste des produits est envoyée au client.
CdU concerné : *MAJ d'une commande*
- *produitNonStock(art)* : L'article *art* n'est plus en stock.
CdU concerné : *MAJ d'une commande*

5.7 SystèmeBalance

L'interface balance qui notifie un manque en quantité de l'article.

Acteur du CdU : Balance, Controleur

- *alarm(art, qte)* : Une quantité *qte* est demandée pour l'article *art* associé à la balance afin d'être ajoutée à la commande.
CdU concerné : *MAJ d'une commande*

5.8 ListeCommande

La liste des commandes de l'utilisateur divisée en articles, avec chacun ses caractéristiques.

Acteur du CdU : Utilisateur, Controleur

- *estArticleCoch(art)* : Un article *art* de la liste des commande a été coché ou décoché.
CdU concerné : *Valider une commande*
- *changerQuantit(art, qte)* : Demande de modification de la quantité de l'article *art* par la quantité *qte*.
CdU concerné : *Valider une commande*
- *majCommande()* : Demande du *SystmeWeb* de mettre à jour la liste des commandes de l'utilisateur.
CdU concerné : *Valider une ommande, MAJ d'une commande*

5.9 PanneauInformation

Un écran pour afficher les différentes informations sur l'état du système (du texte, des lumières, des graphiques, des boutons, ...).

Acteur du CdU : Utilisateur, Controleur

- *afficherErreur(err)* : Demande du *SystmeWeb* d'afficher l'erreur *err* rencontrée.
CdU concerné : *Valider une ommande, MAJ d'une commande*
- *produitNonDispo(art, b)* : Si *b* est vrai, le *SystmeWeb* demande l'affichage d'un texte annonçant que l'article *art* n'est pas disponible en stock. Sinon, le *SystmeWeb* n'affiche rien.
CdU concerné : *MAJ d'une commande*
- *commandeValide(b)* : Si *b*, le *SystmeWeb* demande l'affichage d'un texte annonçant que la commande a été validée. Sinon, le *SystmeWeb* n'affiche rien.
CdU concerné : *Valider une commande*
- *afficherBalance(bal)* : Demande du *SystmeWeb* d'afficher la balance *bal* sélectionnée depuis la liste des balances de l'utilisateur.
CdU concerné : *Configurer les balances*
- *afficherCommande(cmd)* : Demande du *SystmeWeb* d'afficher la liste des commandes *cmd* de l'utilisateur.
CdU concerné : *Valider une commande*
- *afficherFacture(cmd, prixTotal)* : Demande du *SystmeWeb* d'afficher la facture relative à la commande *cmd* avec le prix total *prixTotal* liée à la commande.
CdU concerné : *Valider une commande*
- *confirmAssignBal()* : L'assignement du produit a bien eu lieu.
CdU concerné : *Configurer les balances*
- *erreurAssignBal()* : Une erreur a eu lieu pendant l'assignement de la balance.
CdU concerné : *Configurer les balances*
- *confirmConfigBal()* : La configuration a bien eu lieu.
CdU concerné : *Configurer les balances*
- *erreurConfigBal()* : Une erreur a eu lieu durant la configuration.
CdU concerné : *Configurer les balances*