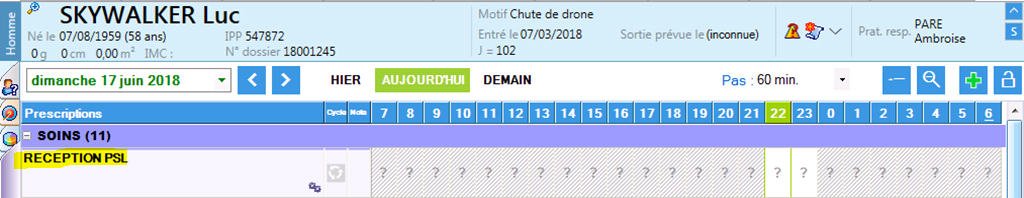
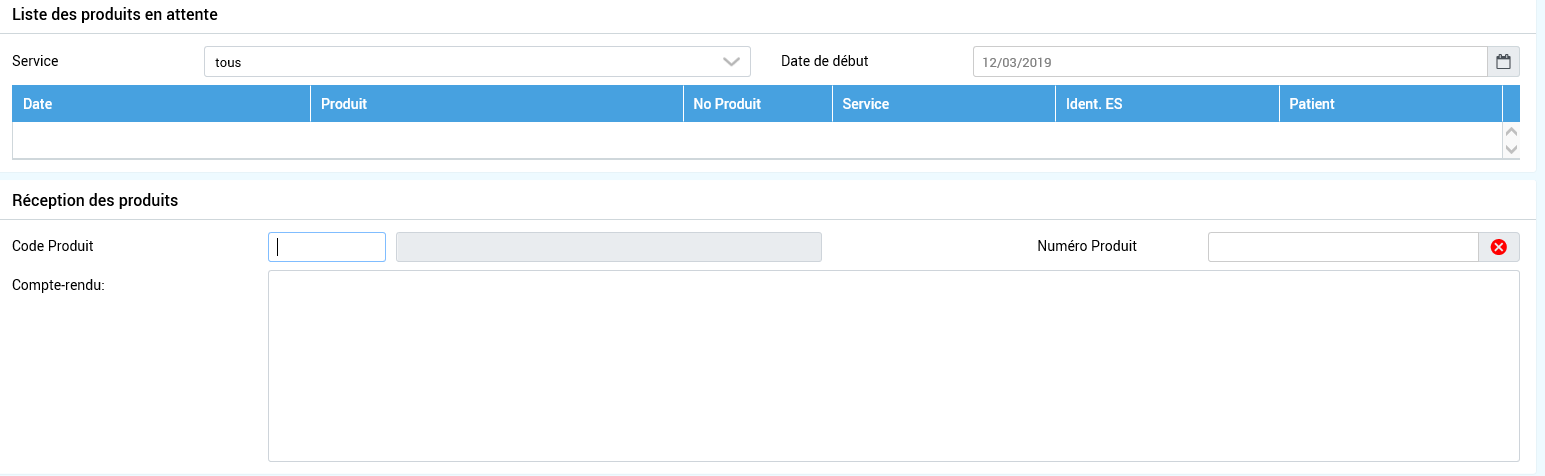
**Documentation Dopasoins**

# **Spécifications fonctionnelles**



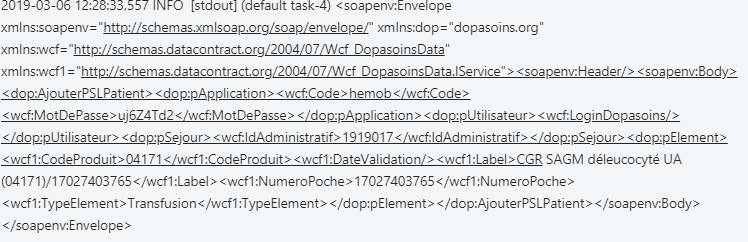
Depuis ce soin de réception l’infirmière peut faire un clic droit sur un appel contextuel lui permettant d’appeler le logiciel eHemobank pour accéder à la page de réception des produits.



Lorsque la réception est faite :

1. **eHemobank** appelle le webservice **DOPA-SOINS** pour valider la réception du produit en ajoutant une tache validée sur la ligne de réception précédemment ajoutée. Pour ce faire il faudra passer la valeurs<<date validation dans la requête soap envoyé lors de l'appel au webservice (on peut visualiser la requête dans le fichier log du serveur).

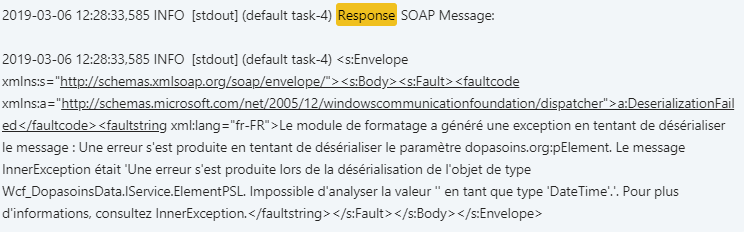
**Exemple de requete (Soap Request):**



Alors 3 cas se présentent:

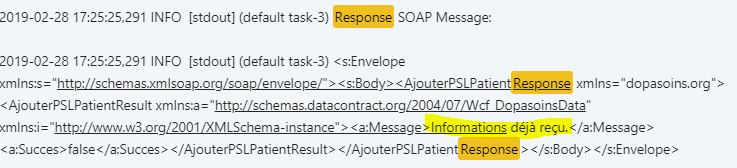
* si un problème de connexion au webservice est survenu ou une exception est invoqué un message d'erreur s'affiche dans l'application **eHemobank** indiquant *"problème de connexion au web service*". Dans ce cas on peut aller dans le fichier log du serveur afin de traiter la réponse soap envoyé par l'application **DOPA-SOINS** contenant une trace sur le problème.

**Exemple:**



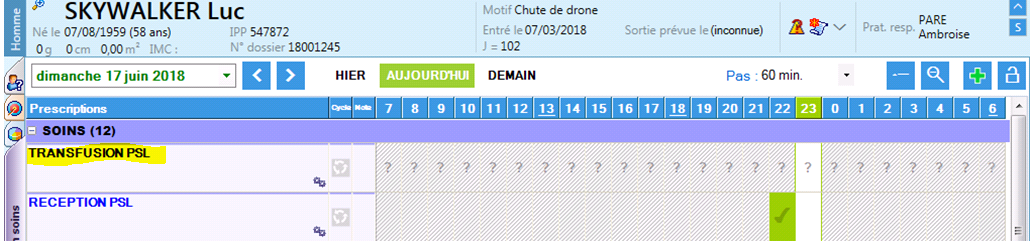
* Sinon si tout est bon on reçoit dans la réponse qui contient le résultat. Grace à lui on est informé du succès ou de l'échec de traitement:
* Success :true si succès ;False si échec.
* Message: contient un message en cas d'échec.

**Exemple de success= false:**

****

Dans ce cas un message d'erreur s'affiche dans l'application **eHemobank** indiquant *"problème de connexion au web service*" concaténé avec le contenu du tag "Message" dans la réponse "Informations déjà réçu".

* Si Success=true ,un nouvel appel au web Service sera fait pour ajouter une ligne de transfusion. Dans ce cas le champ <<Date validation>> sera vide et le champ <<typeElement>> sera "Transfusion".



1. Depuis ce soin de transfusion (image ci dessus) l’infirmière peut faire un clic droit sur un appel contextuel lui permettant d’appeler le logiciel **eHemobank** pour accéder à la page de retour d’information.

Lorsque le retour d’information est validé ( Confirmation, destruction, ou retour au dépôt),**eHemobank** appelle **DOPA-SOINS** pour valider la transfusion (ce traitement est dans retInformationServiceImpl,on a trois appel de dopasoins :***saveConfirmation(),completeDestroy(),retourDepot()***).

1. **Spécifications techniques**
2. **Paramétrage Globale:**

* Pour pourvoir gérer des système externes différents en fonction du code l’UF nous allons changer la structure de **l’item5** (Type de liaison externe de planification de tâches) de la clé **WGENE.**

Cet item va maintenant contenir une liste sous la forme :

**CodeFiness1=CodeUF1=Type de lien1!; CodeUF2=Type de lien2!;~~=Type de lien par defaut pour le Finess1**

~~=Type de lien par defaut

Les type de lien sont DPLA ( CristalNet) ou DPSO ( DOPASYS)

**Exemple:**

**030780092=5000=DPLA!;002=DPLA!; ~~=DPSO**

* Le type de lien devra ensuite être celui paramétré dans le champ **CODE** de la clé **WPLID**.

Dans le cas ou le type de lien est DPSO

La structure de la clé WPLID sera la suivante :

* Item 1 : Liste uniface
* SVC=localisation du service externe (soapEndpointUrl si SVC est vide on utilise soapEndpointUrl=wsdl)
* WSDL = Localisation du fichier WSDL
* LOGAPP = Login identifiant de l’application
* PWDAPP=Mode de passe de l’application
* LOGDEF=Utilisateur DOPA-SOINS par défaut
* Item 2 : url d’appel contextuel pour la reception des produits avec les paramètres à compléter.

Par exemple :

https://localhost:8443/eHematos/load.action?mode=01&ID1=&typID1=IPP&cliID1=.&typID2=NHP&cliID2=GHM&creat=O&prodCd=numProd=&prog=W\_RECEPTIONPRODUIT =NHP&cliID2=GHM&creat=O&prodCd=&numProd=&prog=W\_RECEPTIONPRODUIT

* Item 3 : url d’appel contextuel pour la confirmation des transfusions avec les paramètres à compléter

Par exemple :

https://localhost:8443/eHematos/load.action?mode=01&ID1=&typID1=IPP&cliID1=.&typID2=NHP&cliID2=GHM&creat=O&prodCd=&numProd=&prog=W\_RETINFORMATION

1. Etapes d'appel à **DOPA-SOINS:**
2. **Préparation de la requête dans la méthode createSoapEnvelope() :**

* Initialiser les données de connexion au webservice à partir de l’item 1 de la clé WPLID
* Initialiser les champs des objets des différentes **classes** de DOPA-SOINS à partir des données passées en entrée de l’appel contextuel:
* **Classe « Application »**
* Code = Login identifiant de l’application ( LOGAPP item1 clé WPLID)
* MotDePasse = Mode de passe de l’application ( CPWDAPP item1 clé WPLID)
* **Classe « UtilisateurDopasys »**
* LoginDOPA-SOINS = Code utilisateur(userNTLM récupéré de l'appel contextuel)
* **Classe « SejourDopasys »**
* IdAdministratif = N° de séjour du patient ( Valeur de ID2 de l’appel contextuel)
* **Classe « ElementPSL »**
* CodeProduit = Code du produit réceptionnée(Valeur de prodCd d'appel contextuel)
* NumeroPoche = Numéro de la poche réceptionnée(Valeur de numProd d'appel contextuel)
* Label = "Produits : *Code du produit / Numéro de la poche*" ( Ex 04171 / 38023000817)
* TypeElement = "Reception" ou "Transfusion.
* DateValidation = Date du jour au format YYYYMMDDHH24MISSTT (vide en cas de création d'une tache et différent de vide en cas de validation de tache reception/transfusion).

Appeler le webService de DOPA-SOINS : ***AjouterPSLPatient*** pour valider la tache de réception. **Exemple de requete:**

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"

xmlns:dop="dopasoins.org"

xmlns:wcf="http://schemas.datacontract.org/2004/07/Wcf\_DopasoinsData"

xmlns:wcf1="http://schemas.datacontract.org/2004/07/Wcf\_DopasoinsData.IService">

<soapenv:Header/>

<soapenv:Body>

<dop:AjouterPSLPatient>

<dop:pApplication>

<wcf:Code>hemob</wcf:Code>

<wcf:MotDePasse>uj6Z4Td2</wcf:MotDePasse>

</dop:pApplication>

<dop:pUtilisateur>

<wcf:LoginDopasoins/></wcf:LoginDopasoins>

</dop:pUtilisateur>

<dop:pSejour>

<wcf:IdAdministratif>1919017</wcf:IdAdministratif>

</dop:pSejour>

<dop:pElement>

<wcf1:CodeProduit>04171</wcf1:CodeProduit>

<wcf1:DateValidation>/<wcf1:DateValidation/>

<wcf1:Label>CGR SAGM déleucocyté UA (04171)/17027403765</wcf1:Label>

<wcf1:NumeroPoche>17027403765</wcf1:NumeroPoche>

<wcf1:TypeElement>Transfusion</wcf1:TypeElement>

</dop:pElement>

</dop:AjouterPSLPatient>

</soapenv:Body>

</soapenv:Envelope>

1. **Connexion au webService**

// SAAJ - SOAP Client Testing

**public** String callSoapWebService(Map<String, String> params, String soapEndpointUrl, String soapAction)

**throws** Exception {

// Create SOAP Connection

SOAPConnectionFactory soapConnectionFactory = SOAPConnectionFactory.*newInstance*();

SOAPConnection soapConnection = soapConnectionFactory.createConnection();

// Send SOAP Message to SOAP Server

SOAPMessage soapRequest = *createSOAPRequest*(soapAction, params, soapEndpointUrl);

SOAPMessage soapResponse = soapConnection.call(soapRequest, soapEndpointUrl);

// Print the SOAP Response

System.***out***.println("Response SOAP Message:");

ByteArrayOutputStream out = **new** ByteArrayOutputStream();

soapResponse.writeTo(out);

soapResponse.writeTo(System.***out***);

System.***out***.println();

String message = "";

String succes = "";

***logger***.debug("DEBUT DEBUG");

SOAPBody sb = soapResponse.getSOAPBody();

***logger***.debug("SOAPBody :" + sb.toString());

***logger***.debug("SOAPBody getElementsByTagName :" + sb.getElementsByTagName("a:Succes"));

***logger***.debug("SOAPBody getElementsByTagName item :" + sb.getElementsByTagName("a:Succes").item(0));

Node succesNode = sb.getElementsByTagName("a:Succes").item(0);

***logger***.debug("succesNode :" + succesNode);

succes = succesNode.getFirstChild().getTextContent();

***logger***.debug("succes :" + succes);

***logger***.debug("Response SOAP succes:");

***logger***.debug(succes);

**if** ("false".equals(succes)) {

Node MsgNode = (Node) sb.getElementsByTagName("a:Message").item(0);

message = MsgNode.getFirstChild().getTextContent();

***logger***.debug("Response SOAP Message content:");

***logger***.debug(message);

soapConnection.close();

**return** message;

}

**if** ("true".equals(succes))

soapConnection.close();

**return** **null**;

}