

**Table des matières**

1. Informations techniques .....	2
1.1 Technologies utilisées .....	2
1.2 Pré requis technique .....	2
2 Configuration.....	2
2.1 MAVEN .....	2
2.2 Tomcat7 ou tomcat8 .....	2
2.3 Base de données.....	3
2.4 WebServices authentifié via les rôles tomcat .....	3
2.4 Thème des modules de vues .....	4
2.5 Module transferts départ & gestionnaire.....	4
2.6 Liste des paramètres .....	9
2.7 Base de données.....	9
2.8 Module transferts accueil – Facultatif.....	9
2.9 Mise en place des tests unitaires.....	9
2.10 Mise en place du scheduler version 1.8.5 ( <a href="http://quartz-scheduler.org/">http://quartz-scheduler.org/</a> ) .....	9
2.11 Interopérabilités .....	10

## 1. Informations techniques

### 1.1 Technologies utilisées

- Esup-commons2 version 0.3.2
- PrimeFaces version 6 avec thèmes en version 1.0.10
- JPA
- Hibernate
- CXF
- JSF 2.0

### 1.2 Pré requis technique

#### 1.2.1 Obligatoire

- Tomcat 7 ou version supérieure
- Java(TM) SE Runtime Environment (build 1.8.0\_102b14) (ne pas installer d'openjdk)
- Apache Maven 3.2.5
- svn, version 1.6.17
- Base de données Oracle 11G minimum **ou MySql 5.1 (testé sur la 5.1.49)**
- Annuaire LDAP de type OpenLdap
- Authentification par SSO CAS

#### 1.2.2 Facultatif

- WebServices Apogée (*Facultatif si ré-implémentation de l'interface « `DomainServiceScolariteImpl.java` »*)
- Si WebService Apogée version 4.90la (**dernier version**)

## 2 Configuration

### 2.1 MAVEN

Chez moi pour que maven fonctionne, ajout dans les variables d'environnement :

```
export M2_HOME=/usr/local/maven
export M2=$M2_HOME/bin
export PATH=$JAVA_HOME/bin:$M2:$PATH
```

### 2.2 Tomcat7 ou tomcat8

Vérifier que dans le fichier /etc/default/

Si vous êtes derrière un proxy :

```
JAVA_OPTS="{JAVA_OPTS} -Dhttp.proxyHost=cache-XXX.univ-XXX.fr -Dhttp.proxyPort=8080
Dhttp.nonProxyHosts=XXX.XXX.XXX.XXX"
```

Eviter les problèmes de « PermGenSpace » :

```
JAVA_OPTS="-Djava.awt.headless=true -Xmx128m -XX:+UseConcMarkSweepGC"
JAVA_OPTS="{JAVA_OPTS} -XX:PermSize=512m -XX:MaxPermSize=512m"
```

#### 2.2.1 Particularités Centos

catalina.sh dans le dossier bin sous centos du coup indiquer plutôt dans les variables d'environnement tomcat

Ajoutez ça pour que le JDBC fonctionne -

```
Djavax.sql.DataSource.Factory=org.apache.commons.dbcp.BasicDataSourceFactory"
```

## 2.3 Base de données

Si la base Oracle est en version 12c transfert n'arrivera pas à se connecter l'erreur "ORA-28040: No matching authentication protocol" s'affichera. Il faut modifier le fichier sqlnet.ora du dossier network/admin du serveur oracle pour ajouter la variable suivante :

SQLNET.ALLOWED\_LOGON\_VERSION=8

## 2.4 WebServices authentifié via les rôles tomcat

### 2.4.1 Web.xml

```
<security-constraint>
  <web-resource-collection>
    <web-resource-name>WS - esup-transferts</web-resource-name>
    <url-pattern>/services/*</url-pattern>
  </web-resource-collection>
</security-constraint>

<auth-constraint>
  <role-name>transferts</role-name>
</auth-constraint>
</security-constraint>

<!-- Define the Login Configuration for this Application -->
<login-config>
  <auth-method>BASIC</auth-method>
  <realm-name>Tomcat Manager Application</realm-name>
</login-config>

<!-- Security roles referenced by this web application -->
<security-role>
  <description> Accés au WS - esup-transferts</description>
  <role-name>transferts</role-name>
</security-role>
```

### 2.4.2 Tomcat-users.xml

```
<role rolename="transferts"/>
<user username="farid" password="soso" roles="transferts"/>
```

## 2.4 Thème des modules de vues

La liste des thèmes est disponible ici : <http://www.primefaces.org/themes.html>

Pour changer le thème :

- Allez dans le fichier pom.xml du module de vues (ex : esup-transferts-web-jsf-servlet/pom.xml)
 

```
<dependency>
  <groupId>org.primefaces.themes</groupId>
  <artifactId>all-themes</artifactId>
  <version>1.0.10</version>
</dependency>
```
- Puis dans le fichier web.xml (esup-transferts-web-jsf-servlet/src/main/webapp/WEB-INF/web.xml)
 

```
<context-param>
  <param-name>primefaces.THEME</param-name>
  <param-value>hot-sneaks</param-value>
</context-param>
```

Puis exécuter un mvn clean install dans le module de vues et redéployer le .war généré

## 2.5 Module transferts départ & gestionnaire

### 2.5.1 Dépôt SVN

Voici la commande pour faire le checkout depuis le dépôt sourcesup :

#### 2.5.1.1 Environnement de test

svn co <http://subversion.renater.fr/esup-transferts/trunk/esup-transferts/> esup-transferts\_trunk (pour la TEST la release en dvl)

#### 2.5.1.2 Environnement de test

svn co <http://subversion.renater.fr/esup-transferts/tags/v2.4.0/> esup-transferts\_tags (pour la PROD la release définitive)

### Mise à jour de version mineur d'Esup- transferts

svn info (pour vérifier le N° de version avant MAJ)

svn update

svn info (pour vérifier le N° de version après MAJ)

### 2.5.2 Configuration de l' application

#### 2.5.2.1 config.properties

- Renommez et renseignez le fichier esup-transferts-web-jsf-servlet/src/main/resources/properties/config-exemple.properties en esup-transferts-web-jsf-servlet/src/main/resources/properties/config.properties

NOTE : concenrnat l'option smtp.interceptAll=true (**ATTENTION à ne pas mettre en true que pour les tests, tous les emails seront redirigés qu'à cette adresse**)

#### 2.5.2.2 log4j.properties

- Renommez et renseignez le fichier esup-transferts-web-jsf-servlet/src/main/resources/log4j-exemple.properties en esup-transferts-web-jsf-servlet/src/main/resources/log4j.properties

### 2.5.2.3 WS Apogée

- Renommez et renseignez le fichier esup-transferts-web-jsf-servlet/src/main/resources/configUrlServices-exemple.properties en configUrlServices.properties

NOTE : mettre tout d'abord WS\_SSL\_MODE à « false » pour les premiers essais car il est assez difficile de faire fonctionner les WS en https (Il n'y aucune erreur dans les logs). Pour tester il suffit d'afficher l'offre de formation dans l'application transferts, celle-ci doit s'afficher.

- Mise à jour de librairie java pour l'appel au WS Apogée

```
cd esup-transferts-web-jsf-servlet/src/main/webapp/WEB-INF/lib/
```

Vérifiez la version du client WS Apogée (apo-webservicesclientXXX.jar fournie par l'AMUE)

Si besoin la recopier et le renommer apo-webservicesclient490la.jar (ou lien symbolique)

- Module départ :

```
esup-transferts-web-jsf-servlet/src/main/webapp/WEB-INF/lib/
```

```
apo-webservicesclient.jar -> ../../../../../../apogee/apo-webservicesclient490la.jar
```

- Module accueil

```
esup-transferts-accueil-web-jsf-servlet/src/main/webapp/WEB-INF/lib/
```

```
apo-webservicesclient.jar -> ../../../../../../apogee/apo-webservicesclient490la.jar
```

### 2.5.2.4 Fichiers \*.xsl (génération des fichiers pdf)

- Renommez et renseignez le fichier esup-transferts-web-jsf-servlet/src/main/webapp/media/xmlxsl/etudiant-exemple.xsl en etudiant.xsl

- Renommez et renseignez le fichier esup-transferts-web-jsf-servlet/src/main/webapp/media/xmlxsl/etudiant\_accueil-exemple.xsl en etudiant\_accueil.xsl

Les logos servant aux documents sont à placer dans /usr/local/src/transferts-apos/esup-transferts-web-jsf-servlet/src/main/webapp/media/xmlxsl/images (Sinon Logo UA sur les demandes de transferts)

### 2.5.2.5 Personnalisation des mails envoyés par l'application

Il est possible de personnaliser les messages de l'application transferts départ et du contenu des emails envoyés par l'application

Il suffit de recopier les lignes à modifier présentes dans esup-transferts-test/WEB-INF/classes/properties/i18n/bundles/messages\_fr.properties dans le fichier Custom\_fr.properties qui est situé sous ce même dossier. Ce fichier écrasera les chaînes par défaut du fichier messages\_fr.properties.

Ce fichier existe également pour le module accueil dans esup-transferts-accueil-test/WEB-INF/classes/properties/i18n/bundles/

### 2.5.3 Compilation

Voici la commande pour faire le déploiement une fois à la racine du projet :

```
mvn -Dhttp.proxyHost=cache-XXX.univ-XXX.fr -Dhttp.proxyPort=8080 -Dhttps.proxyHost=cache-XXX.univ-XXX.fr -Dhttps.proxyPort=8080 clean install
```

```
[INFO] -----
[INFO] Reactor Summary:
[INFO]
[INFO] esup-transferts ..... SUCCESS [ 0.241 s]
[INFO] esup-transferts-domain-beans ..... SUCCESS [ 1.395 s]
[INFO] esup-transferts-dao ..... SUCCESS [ 0.775 s]
[INFO] esup-transferts-utils ..... SUCCESS [ 0.680 s]
[INFO] esup-transferts-domain-services ..... SUCCESS [ 1.055 s]
[INFO] esup-transferts-web-jsf-servlet ..... SUCCESS [ 5.579 s]
[INFO] esup-transferts-accueil-web-jsf-servlet ..... SUCCESS [ 3.555 s]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 13.403 s
[INFO] Finished at: 2017-03-30T13:56:07+02:00
[INFO] Final Memory: 81M/215M
[INFO] -----
```

#### 2.5.4 Déploiement (primo installe version 2.4.0)

Au premier démarrage de l'application, mettre le paramètre `hibernate.generateDdl=true` dans le fichier de configuration "config.properties". Cela va créer le modèle physique de la Base de Données. Une fois le schéma de Base de données créer, exécutez le script SQL fourni à la racine de l'application :

**ATTENTION : Les fichiers SQL sont encodés en UTF8, vérifiez bien votre encodage de caractère lors des créations/insertions.**

/sql/ nomenclatures-esup-transferts-primo-install.sql

- `cp esup-transferts-web-jsf-servlet/target/esup-transferts.war /var/lib/tomcat7/webapps/`
- `/etc/init.d/tomcat7 restart`

#### 2.5.5 Déploiement (mise à jour 2.3.0 vers 2.4.0)

Au premier démarrage de l'application, mettre le paramètre `hibernate.generateDdl=true` dans le fichier de configuration "config.properties". Cela va créer le modèle physique de la Base de Données. Une fois le schéma de Base de données créer, exécutez le script SQL fourni à la racine de l'application :

**ATTENTION : Les fichiers SQL sont encodés en UTF8, vérifiez bien votre encodage de caractère lors des créations/insertions.**

/sql/update\_v2.3.0\_vers\_v2.4.0-esup-transferts.sql

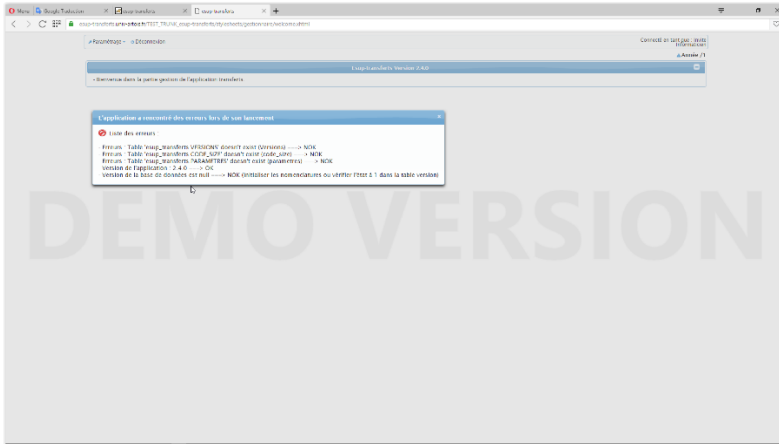
- `cp esup-transferts-web-jsf-servlet/target/esup-transferts.war /var/lib/tomcat6/webapps/`
- `/etc/init.d/tomcat7 restart`
- Passez le script SQL /sql/update\_v2.3.0\_vers\_v2.4.0-esup-transferts.sql **!!! une fois l'application lancée !!!**

#### 2.5.6 Explotation

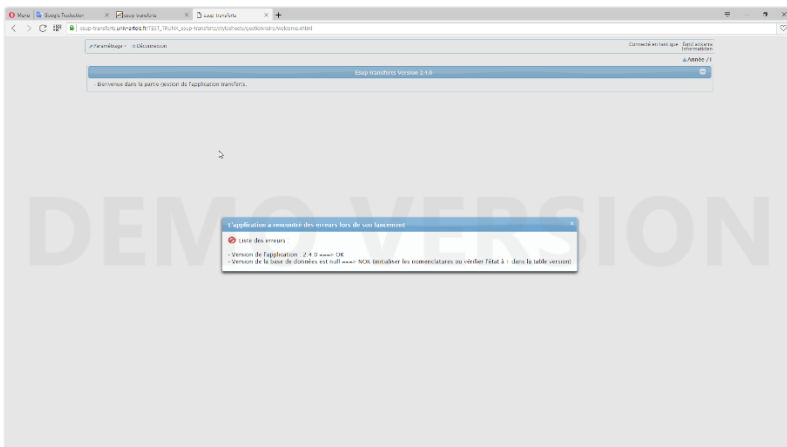
- URL :
  - Module gestionnaire :  
`http(s)://monserveurtransferts/mawebapptransfertsdepart/stylesheets/gestionnaire/welcome.xhtml`
  - Module départ  
`https://monserveurtransferts/mawebapptransfertsdepart`
  - Module accueil  
`http(s)://monserveurtransferts/mawebapptransfertsaccueil`
- Pour augmenter le niveau de log passez la variable `log4j.rootLogger` du fichier `log4g.properties` à `DEBUG` et re-déployer l'application
- Quand on passe en prod, ne pas oublier de modifier le `InterceptAll` du fichier `config.properties`, le passer à `FALSE` pour que les emails sont bien envoyés aux étudiants. Ne pas oublier de modifier les URLS de test dans les emails (`Custom_fr.properties`).

## 2.5.7 Résultats

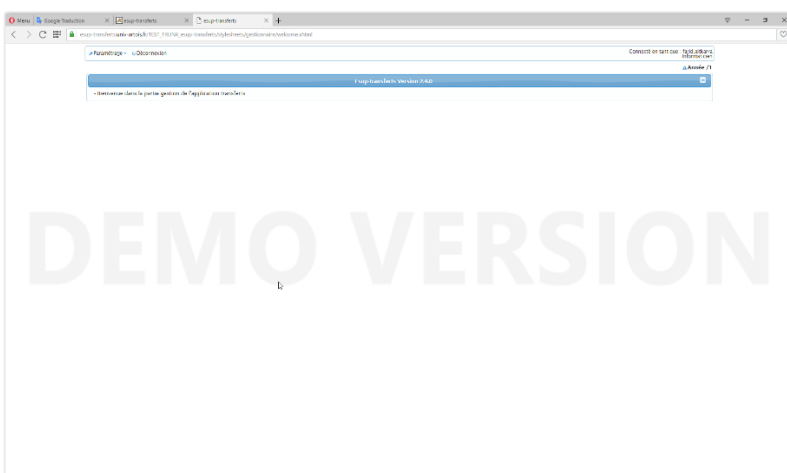
### 1. Base de données non créée



### 2. Base de données créée mais nomenclatures absentes

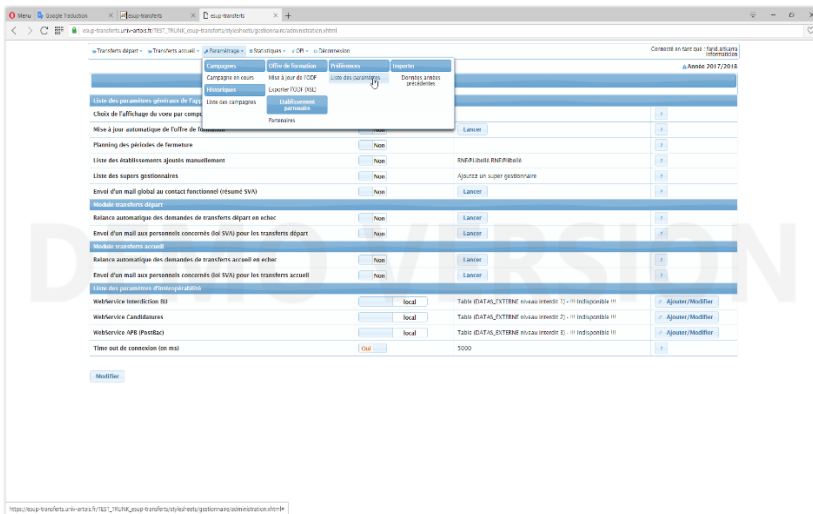


### 3. Base de données et nomenclatures OK





## 2.6 Liste des paramètres



Des boutons d'aides sont disponible sur chaque paramètre

## 2.7 Base de données

### 2.7.1 Table "DATAS\_EXTERNE" (exemple des interdits BU, VAP ou autre...)

Les tests d'interdits (exemple BU) se font lors de l'authentification d'une part dans les interdits BDD SCOLARITE puis d'autre part dans la table "DATAS\_EXTERNE" du schéma transferts. Si les interdits BU ou autre ne sont pas saisis dans la BDD SCOLARITE, il est faut renseigner cette table via une procédure PL/SQL avec un DBLINK par exemple ou d'une autre manière.

## 2.8 Module transferts accueil – Facultatif

Voir déploiement module transferts départ et gestionnaire

## 2.9 Mise en place des tests unitaires

Il faut renommer le fichier en esup-transferts-web-jsf-servlet/src/test/resources/META-INF/config\_TEST\_UNITAIRE-exemple.properties en config\_TEST\_UNITAIRE.properties puis le configurer.

Renommez et renseignez le fichier esup-transferts-web-jsf-servlet/src/test/resources/META-INF/configUrlServices-exemple.properties en configUrlServices.properties (Apogée)

Attention : éviter de faire des tests unitaires sur la base de données de production

!!! TODO !!!

## 2.10 Mise en place du scheduler version 1.8.5 (<http://quartz-scheduler.org/>)

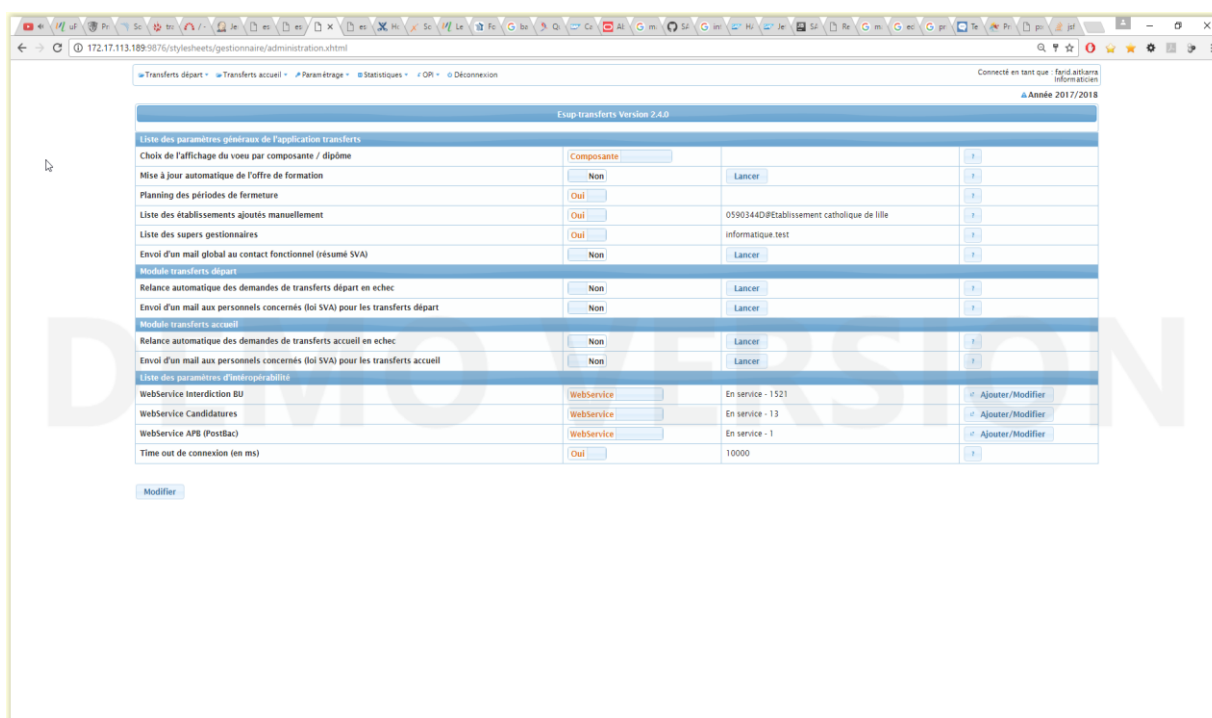
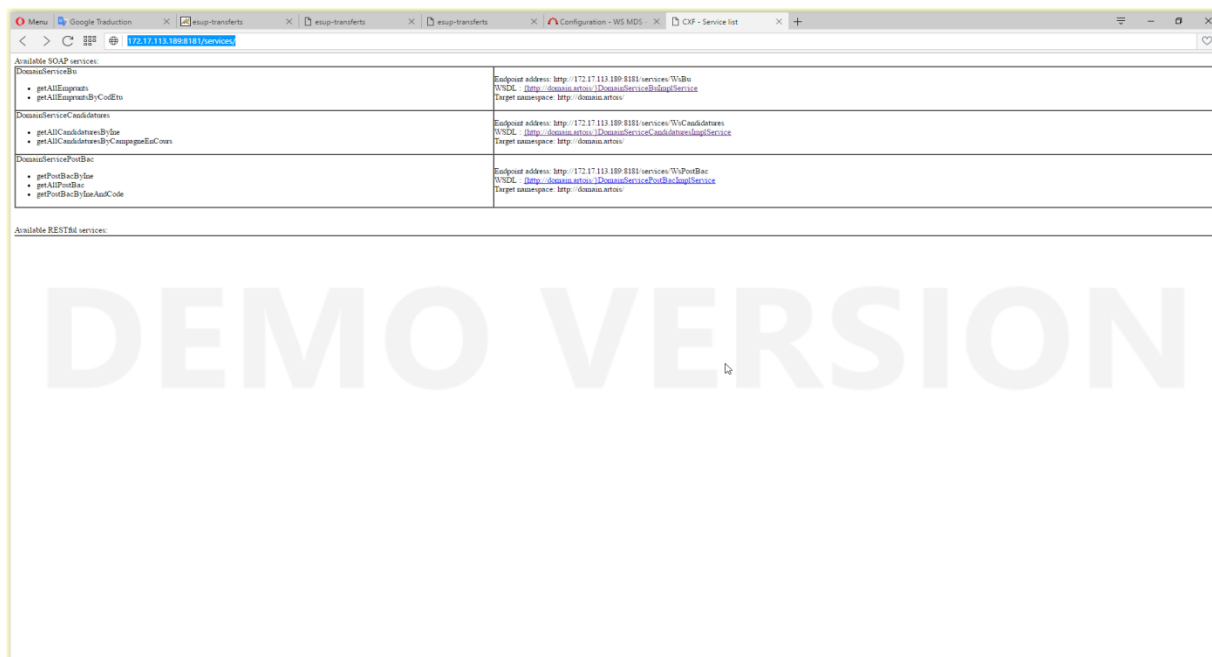
Dans le fichier esup-transferts-web-jsf-servlet/src/main/resources/properties/config.properties, renseigner la propriété scheduler.cronExpression= exemple(0 30 20 \* \* ? \*, <http://quartz-scheduler.org/api/2.2.0/org/quartz/CronExpression.html>)

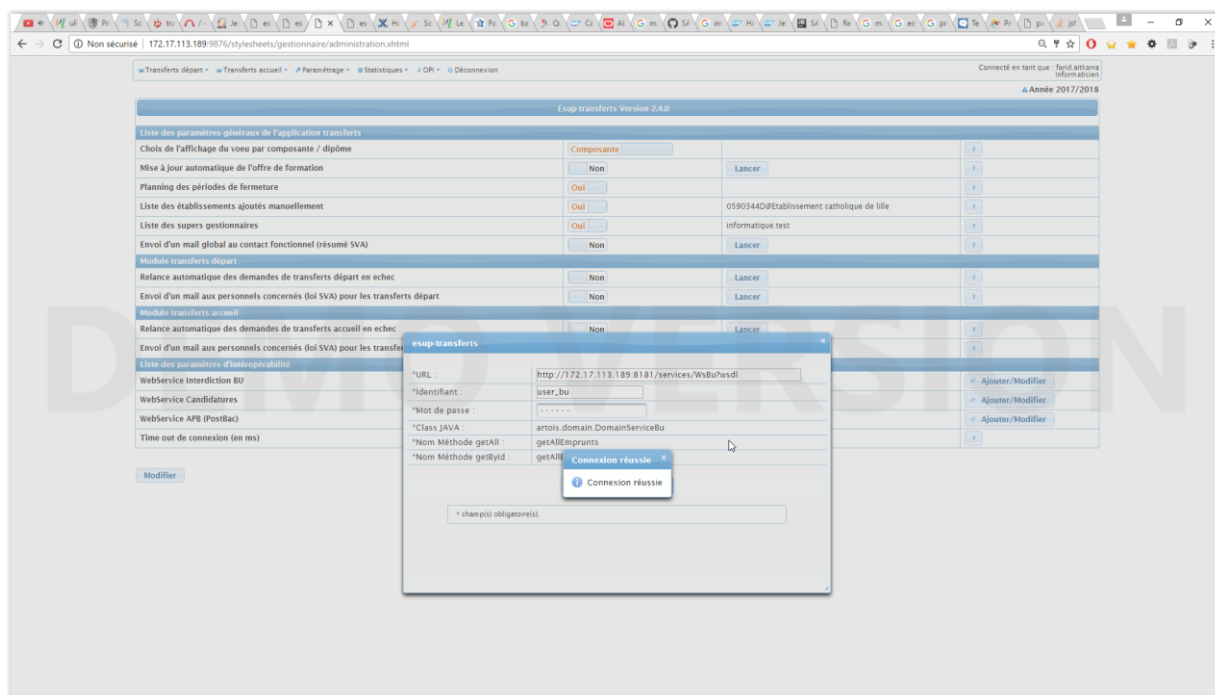
## 2.11 Interopérabilités

L'interopérabilité peut être possible avec d'autres système d'information :

- SIGB
- Candidatures
- APB PostBac

Un webService multi-datasources est également disponible sur le dépôt : <http://svn.univ-artois.fr/svn/wssi/trunk>





Vous pouvez tester les retours des webServices avec le logiciel : <https://www.soapui.org/>