

Gestion des Bannettes

GedBannetteList

GedBannetteList permet de récupérer la liste des bannettes GED au format XML.

```
string GedBannetteList(  
    char *User  
);
```

Paramètres

User : Login de l'utilisateur



Conseil

Le paramètre User permet de filtrer les résultats aux seules bannettes accessibles à cet utilisateur.

Codes retour

Chaine XML ou code erreur

```
<bannettes>  
  <bannette id= "ID">  
    <CHAMP>VALEUR</CHAMP>  
    <CHAMP>VALEUR</CHAMP>  
    ...  
  <bannette id= "ID">  
    <CHAMP>VALEUR</CHAMP>  
    <CHAMP>VALEUR</CHAMP>  
    ...  
  </bannette>  
  ...  
</bannettes>
```

Exemple (php/SOAP)

```
$ApiGed = new  
SoapClient('http://srvGED/Interconnexion/SoapService.php?wsdl');  
$result = $ApiGed->GedBannetteList('admin');  
echo $result;
```

GedBannetteImporterDocument

GedBannetteImporterDocument permet d'importer un document dans une bannette.

```
string GedBannetteList(  
    char *Bannette,  
    char *User,  
    char *FileStream,  
    char *FileName  
);
```

Paramètres

Bannette : Nom de la bannette

User : Login de l'utilisateur

FileStream : Contenu du fichier à importer au format base64

FileName : Nom à donner au fichier à importer (NOM.EXT)

Codes retour

0 ou code erreur (-1,-19,-21,-26,-27)

Exemple (php/SOAP)

```
$ApiGed = new  
SoapClient('http://srvGED/Interconnexion/SoapService.php?wsdl');  
$result = $ApiGed->GedBannetteImporterDocument('Bannette accueil', 'admin',  
base64_encode(file_get_contents('C:\\test.pdf')), 'Fichier a classer.pdf');  
echo $result;
```

GedBannetteDocumentList

GedBannetteDocumentList permet de récupérer la liste des documents d'une bannette dont le nom est passé en paramètre et suivant des paramètres de pagination

```
string GedBannetteDocumentList(  
    char *bannetteName,  
    char *login,  
    int page,  
    int nbDocPerPage,  
    char format  
);
```

Paramètres

bannetteName : Nom de la bannette

login : Login de l'utilisateur

page : Indice de la page demandée.

Ce nombre peut être compris de 1 à n. (Valeur par défaut : 1)

nbDocPerPage : Nombre de documents par page.

Ce nombre peut être compris de 1 à 500. (Valeur par défaut : 500)

format : Format de retour.

Valeurs possibles :

- Pour obtenir une chaîne XML, la valeur à passer est : 'XML' (valeur par défaut)
- Pour obtenir une chaîne JSON, la valeur à passer est : 'JSON'

Codes retour

Chaîne XML ou JSON ou code erreur :

- Chaîne XML – Exemple :

```
<?xml version= "1.0" encoding= "UTF-8"?>
<GedBannetteDocumentList>
  <data>
    <currentPage>1</currentPage>
    <currentPageNbDoc>2</currentPageNbDoc>
    <nbDocPerPage>500</nbDocPerPage>
    <totalNbDoc>2</totalNbDoc>
  </data>
  <documents>
    <document id="8904">
      <iddoc>8904</iddoc>
      <file><![CDATA[mon_document_001.docx]]></file>
      <ext><![CDATA[docx]]></ext>
      <size>11457</size>
      <uid>1969</uid>
    </document>
    <document id="9437">
      <iddoc>9437</iddoc>
      <file><![CDATA[mon_document _002.pdf]]></file>
      <ext><![CDATA[pdf]]></ext>
      <size>83716</size>
      <uid>2679</uid>
    </document>
  </documents>
</GedBannetteDocumentList>
```

- Chaîne JSON – Exemple :

```

{
  « data »:
  {
    « currentPage »:1,
    « currentPageNbDoc »:2,
    « nbDocPerPage »:500,
    « totalNbDoc »:2
  },
  « documents »:
  {
    « 8904 »:
    {
      « iddoc »:8904,
      « file »: »mon_document_001.docx »,
      « ext »: »docx »,
      « size »:11457,
      « uid »:1969
    },
    « 9437 » :
    {
      « file »: » mon_document_002.pdf »,
      « ext »: »pdf »,
      « size »:83716,
      « uid »:2679
    }
  }
}

```

- Code erreur :
 - '-1' : Problème de connexion à la base de données

Explications :

- *currentPage* : indice de la page dont la liste des documents est retournée
- *currentPageNbDoc* : nombre de documents dans la page en cours
- *nbDocPerPage* : nombre de documents demandé par page
- *totalNbDoc* : nombre total de documents dans la bannette

Remarque :

currentPageNbDoc et *nbDocPerPage* sont normalement identiques sauf pour la dernière page où ils peuvent être différents.

Mode UID activé

Ce paragraphe surcharge les indications données précédemment.

- Codes retour :

Chaine XML ou JSON ou code erreur :

- Chaîne XML : l'attribut *id* prend la valeur de l'UID – Exemple :

```
<?xml version= »1.0" encoding= »UTF-8"?>
<GedBannetteDocumentList>
  <data>
    <currentPage>1</currentPage>
    <currentPageNbDoc>2</currentPageNbDoc>
    <nbDocPerPage>500</nbDocPerPage>
    <totalNbDoc>2</totalNbDoc>
  </data>
  <documents>
    <document id="1969">
      <iddoc>8904</iddoc>
      <file><![CDATA[mon_document_001.docx]]></file>
      <ext><![CDATA[docx]]></ext>
      <size>11457</size>
      <uid>1969</uid>
    </document>
    <document id="2679">
      <iddoc>9437</iddoc>
      <file><![CDATA[mon_document _002.pdf]]></file>
      <ext><![CDATA[pdf]]></ext>
      <size>83716</size>
      <uid>2679</uid>
    </document>
  </documents>
</GedBannetteDocumentList>
```

- Chaîne JSON : la clé *id* prend la valeur de l'UID – Exemple :

```
{
  « data »:
  {
    « currentPage »:1,
    « currentPageNbDoc »:2,
    « nbDocPerPage »:500,
    « totalNbDoc »:2
  },
  « documents »:
  {
    « 1969 »:
    {
      « iddoc »:8904,
      « file »: »mon_document_001.docx »,
      « ext »: »docx »,
      « size »:11457,
      « uid »:1969
    },
    « 2679 » :
    {
```

```

        « iddoc »:9437,
        « file »: » mon_document_002.pdf »,
        « ext »: »pdf »,
        « size »:83716,
        « uid »:2679
    }
}
}

```

- Code erreur :
 - '-1' : Problème de connexion à la base de données

Exemple (php/SOAP)

```

$ApiGed = new
SoapClient('http://srvGED/Interconnexion/SoapService.php?wsdl');
$result = $ApiGed->GedBannetteDocumentList('Bannette accueil', 'admin', 2,
100, 'JSON');
echo $result;

```

GedBannetteDocumentListById

GedBannetteDocumentListById permet de récupérer la liste des documents d'une bannette dont l'identifiant est passé en paramètre et suivant des paramètres de pagination

```

string GedBannetteDocumentListById(
    char *bannetteId,
    char *login,
    int page,
    int nbDocPerPage,
    char format
);

```

Paramètres

bannetteId : Numéro système de la bannette

login : Login de l'utilisateur

page : Indice de la page demandée.

Ce nombre peut être compris de 1 à n. (Valeur par défaut : 1)

nbDocPerPage : Nombre de documents par page.

Ce nombre peut être compris de 1 à 500. (Valeur par défaut : 500)

format : Format de retour.

Valeurs possibles :

- Pour obtenir une chaîne XML, la valeur à passer est : 'XML' (valeur par

défaut)

- Pour obtenir une chaîne JSON, la valeur à passer est : 'JSON'

Codes retour

Chaîne XML ou JSON ou code erreur :

- Chaîne XML – Exemple :

```
<?xml version= »1.0" encoding= »UTF-8"?>
<GedBannetteDocumentList>
  <data>
    <currentPage>1</currentPage>
    <currentPageNbDoc>2</currentPageNbDoc>
    <nbDocPerPage>500</nbDocPerPage>
    <totalNbDoc>2</totalNbDoc>
  </data>
  <documents>
    <document id= »8904« >
      <iddoc>8904</iddoc>
      <file><![CDATA[mon_document_001.docx]]></file>
      <ext><![CDATA[docx]]></ext>
      <size>11457</size>
      <uid>1969</uid>
    </document>
    <document id= »9437« >
      <iddoc>9437</iddoc>
      <file><![CDATA[mon_document _002.pdf]]></file>
      <ext><![CDATA[pdf]]></ext>
      <size>83716</size>
      <uid>2679</uid>
    </document>
  </documents>
</GedBannetteDocumentList>
```

- Chaîne JSON – Exemple :

```
{
  « data »:
  {
    « currentPage »:1,
    « currentPageNbDoc »:2,
    « nbDocPerPage »:500,
    « totalNbDoc »:2
  },
  « documents »:
  {
    « 8904 »: {
      « iddoc »:8904,
      « file »: »mon_document_001.docx »,
      « ext »: »docx »,

```

```

    « size »:11457,
    « uid »:1969
  },
  « 9437 » :{
    « iddoc »:9437,
    « file »: » mon_document_002.pdf »,
    « ext »: »pdf »,
    « size »:83716,
    « uid »:1969
  }
}
}
}

```

- Code erreur :
 - '-1' : Problème de connexion à la base de données

Explications :

- *currentPage* : indice de la page dont la liste des documents est retournée
- *currentPageNbDoc* : nombre de documents dans la page en cours
- *nbDocPerPage* : nombre de documents par page
- *totalNbDoc* : nombre total de documents dans la bannette

Remarque :

currentPageNbDoc et *nbDocPerPage* sont normalement identiques sauf pour la dernière page où ils peuvent être différents.

Mode UID activé

Ce paragraphe surcharge les indications données précédemment.

- Codes retour :

Chaîne XML ou JSON ou code erreur :

- Chaîne XML : l'attribut *id* prend la valeur de l'UID – Exemple :

```

<?xml version= »1.0" encoding= »UTF-8"?>
<GedBannetteDocumentList>
  <data>
    <currentPage>1</currentPage>
    <currentPageNbDoc>2</currentPageNbDoc>
    <nbDocPerPage>500</nbDocPerPage>
    <totalNbDoc>2</totalNbDoc>
  </data>
  <documents>
    <document id= »1969">
      <iddoc>8904</iddoc>
      <file><![CDATA[mon_document_001.docx]]></file>
      <ext><![CDATA[docx]]></ext>
    </document>
  </documents>
</GedBannetteDocumentList>

```

```

        <size>11457</size>
        <uid>1969</uid>
    </document>
<document id= »2679">
    <iddoc>9437</iddoc>
    <file><![CDATA[mon_document _002.pdf]]></file>
    <ext><![CDATA[pdf]]></ext>
    <size>83716</size>
    <uid>2679</uid>
</document>
</documents>
</GedBannetteDocumentList>

```

- Chaîne JSON : la clé *id* prend la valeur de l'UID – Exemple :

```

{
  « data »:
  {
    « currentPage »:1,
    « currentPageNbDoc »:2,
    « nbDocPerPage »:500,
    « totalNbDoc »:2
  },
  « documents »:
  {
    « 1969 »:
    {
      « iddoc »:8904,
      « file »: »mon_document_001.docx »,
      « ext »: »docx »,
      « size »:11457,
      « uid »:1969
    },
    « 2679 » :
    {
      « iddoc »:9437,
      « file »: » mon_document_002.pdf »,
      « ext »: »pdf »,
      « size »:83716,
      « uid »:2679
    }
  }
}

```

- Code erreur :
 - '-1' : Problème de connexion à la base de données

Exemple (php/SOAP)

```
$ApiGed = new
```

```
SoapClient('http://srvGED/Interconnexion/SoapService.php?wsdl');
$result = $ApiGed->GedBannetteDocumentListById(23, 'admin', 2, 100, 'JSON');
echo $result;
```

GedBannetteDocumentIndex

GedBannetteDocumentIndex indexe un document d'une bannette et pour laquelle le nom est passé en paramètre

```
string GedBannetteDocumentIndex(
    char *bannetteName,
    char *login,
    int *iddoc,
    char *armoire,
    int *idDossier,
    char classifying,
    int iddt,
    char format
);
```

Paramètres

bannetteName : Nom de la bannette

login : Login de l'utilisateur

iddoc : Identifiant système du document en bannette à indexer

armoire : Nom de l'armoire de destination

idDossier : Identifiant système du dossier GED de destination

classifying : Classement (= élément de Plan De Classement) de destination : SD/SSD/FILE

Valeur par défaut : chaîne vide

Important : ce paramètre est écrasé par le classement par défaut de la bannette s'il en existe un

iddt : Identifiant système du dossier de travail

Valeur par défaut : 0

format : Format de retour

Valeurs possibles :

- Pour obtenir une chaîne XML, la valeur à passer est : 'XML' (valeur par défaut)
- Pour obtenir une chaîne JSON, la valeur à passer est : 'JSON'

Pré-requis et cohérence des paramètres

- L'utilisateur (login) doit exister

- La bannette (nom) doit exister
- L'utilisateur être habilité sur la bannette et doit avoir le droit d'indexation sur la bannette
- L'armoire (nom) doit exister
- L'utilisateur doit être habilité sur l'armoire
- L'armoire doit être autorisée pour la bannette
- Le document (id) doit exister dans la bannette
- Le dossier (id) doit exister dans l'armoire
- L'utilisateur doit être autorisé à accéder au dossier (faire attention s'ils existent aux champs permission, permissions sur dossier, etc)
- Le plan de classement paramétré par défaut pour la bannette doit toujours exister dans l'armoire
- Il n'est pas nécessaire de transmettre un classement (paramètre *classifying*) **sauf si** aucun classement par défaut n'est paramétré pour la bannette
- Le classement (paramètre *classifying*) doit être valide :
 - Pas plus de 2 slashes (/) au total dans toute la chaîne
 - Interdiction d'avoir 2 slashes d'affilés
 - La chaîne ne doit pas comporter les caractères * ? » < > | :
 - Chaque sous-chaîne (chaînes séparées par un slash (/)) ne doit pas excéder 128 caractères

Remarque : le caractère anti-slash (\) peut être employé à la place du caractère slash (/)
- Si le classement (paramètre *classifying*) transmis est hors plan de classement, l'utilisateur ne doit pas être en plan de classement obligatoire
- L'utilisateur doit avoir le droit d'ajout sur le classement
- Si l'indexation doit se faire dans un dossier de travail (= un identifiant système de dossier de travail est transmis), alors :
 - Le dossier de travail (id) doit exister dans le dossier GED
 - Le classement transmis doit être dans le Plan De Classement (PDC) car le hors PDC n'est pas autorisé dans un dossier de travail
 - Le classement doit être autorisé à être créé dans un dossier de travail
- Si l'indexation ne se fait pas dans un dossier de travail (= aucun un identifiant système de dossier de travail n'est transmis ou bien il est transmis avec la valeur 0) :
 - Le classement ne doit pas être autorisé à être créé uniquement dans un dossier de travail

Codes retour

Chaîne XML ou JSON ou code erreur :

- Chaîne XML – Exemple :

```
<?xml version="1.0" encoding="UTF-8"?>
<GedBannetteDocumentIndex>
  <iddoc>2365</iddoc>
  <uid>3023</uid>
</GedBannetteDocumentIndex>
```

- Chaîne JSON – Exemple :

```
{
  "iddoc": 2365,
  "uid": 3023
}
```

- Code erreur : -1, -2, -3, -4, -5, -7, -12, -15, -27, -28 – Explications voir Annexe – Liste des codes retour

Mode UID activé

Ce paragraphe surcharge les indications données précédemment.

- Paramètres :

bannetteName : Laisser vide

iddoc : Identifiant unique du document (UID) à indexer

Exemple (php/SOAP)

```
$ApiGed = new
SoapClient('http://srvGED/Interconnexion/SoapService.php?wsdl');
$result = $ApiGed->GedBannetteDocumentIndex('Bannette accueil', 'admin', 236,
'individu', 98526, 'identite/passeport', 0, 'JSON');
echo $result;
```

GedBannetteDocumentIndexById

GedBannetteDocumentIndexById indexe un document d'une bannette et pour laquelle l'identifiant système est passé en paramètre

```
string GedBannetteDocumentIndex(
  char *bannetteId,
  char *login,
  int *iddoc,
  char *armoire,
  int *idDossier,
  char classifying,
  int iddt,
  char format
);
```

Paramètres

bannetteName : Identifiant système de la bannette

login : Login de l'utilisateur

iddoc : Identifiant système du document en bannette à indexer

armoire : Nom de l'armoire de destination

idDossier : Identifiant système du dossier GED de destination

classifying : Classement de destination : SD/SSD/FILE

Valeur par défaut : chaîne vide

iddt : Identifiant système du dossier de travail

Valeur par défaut : 0

format : Format de retour

Valeurs possibles :

- Pour obtenir une chaîne XML, la valeur à passer est : 'XML' (valeur par défaut)
- Pour obtenir une chaîne JSON, la valeur à passer est : 'JSON'

Pré-requis et cohérence des paramètres

- L'utilisateur (login) doit exister
- La bannette (id) doit exister
- L'utilisateur être habilité sur la bannette et doit avoir le droit d'indexation sur la bannette
- L'armoire (nom) doit exister
- L'utilisateur doit être habilité sur l'armoire
- L'armoire doit être autorisée pour la bannette
- Le document (id) doit exister dans la bannette
- Le dossier (id) doit exister dans l'armoire
- L'utilisateur doit être autorisé à accéder au dossier (faire attention s'ils existent aux champs permission, permissions sur dossier, etc)
- Le plan de classement paramétré par défaut pour la bannette doit toujours exister dans l'armoire
- Il n'est pas nécessaire de transmettre un classement (paramètre *classifying*) **sauf si** aucun classement par défaut n'est paramétré pour la bannette
- Le classement (paramètre *classifying*) doit être valide :
 - Pas plus de 2 slashes (/) au total dans toute la chaîne
 - Interdiction d'avoir 2 slashes d'affilés
 - La chaîne ne doit pas comporter les caractères * ? » < > | :
 - Chaque sous-chaîne (chaînes séparées par un slash (/)) ne doit pas excéder 128 caractères

Remarque : le caractère anti-slash (\) peut être employé à la place du caractère slash (/)
- Si le classement (paramètre *classifying*) transmis est hors plan de classement, l'utilisateur ne doit pas être en plan de classement obligatoire
- L'utilisateur doit avoir le droit d'ajout sur le classement
- Si l'indexation doit se faire dans un dossier de travail (= un identifiant système de dossier de travail est transmis), alors :
 - Le dossier de travail (id) doit exister dans le dossier GED

- Le classement transmis doit être dans le Plan De Classement (PDC) car le hors PDC n'est pas autorisé dans un dossier de travail
- Le classement doit être autorisé à être créé dans un dossier de travail
- Si l'indexation ne se fait pas dans un dossier de travail (= aucun un identifiant système de dossier de travail n'est transmis ou bien il est transmis avec la valeur 0) :
 - Le classement ne doit pas être autorisé à être créé uniquement dans un dossier de travail

Codes retour

Chaine XML ou JSON ou code erreur :

- Chaîne XML – Exemple :

```
<?xml version="1.0" encoding="UTF-8"?>
<GedBannetteDocumentIndex>
  <iddoc>2365</iddoc>
  <uid>3023</uid>
</GedBannetteDocumentIndex>
```

- Chaîne JSON – Exemple :

```
{
  "iddoc": 2365,
  "uid": 3023
}
```

- Code erreur : -1, -2, -3, -4, -5, - 7, -12, -15, -27, -28 – Explications voir Annexe – Liste des codes retour

Exemple (php/SOAP)

```
$ApiGed= new SoapClient('http://srvGED/Interconnexion/SoapService.php?wsdl');
$result= $ApiGed->GedBannetteDocumentIndex(2, 'admin', 236, 'individu',
98526, 'identite/passeport', 0, 'JSON');
echo $result;
```