



***Simplify***  
**Protocole d'accès HTTP  
au Service Simplify**

**v1.7**

**31 Octobre 2018**

***Simplify***

2, rue Georges Méliès  
78390 BOIS D'ARCY

Tél: +33 (0)1 61 38 14 05 - <http://www.simplify.fr> - [contact@simplify.fr](mailto:contact@simplify.fr)

## Table des matières

<b>1</b>	<b>INTRODUCTION .....</b>	<b>3</b>
1.1	Destinataires du document .....	3
1.2	Modifications par rapport à la version 1.1 .....	3
1.3	Modifications par rapport à la version 1.2 .....	3
1.4	Modifications par rapport à la version 1.3 .....	3
1.5	Modifications par rapport à la version 1.4 .....	4
1.6	Modifications par rapport à la version 1.5 .....	4
1.7	Modifications par rapport à la version 1.6 .....	4
<b>2</b>	<b>FONCTIONNEMENT.....</b>	<b>5</b>
2.1	Généralités .....	5
2.2	Communications réseau.....	5
2.3	Chiffrement SSL des flux.....	5
<b>3</b>	<b>MANUEL DU PROTOCOLE D'ACCES HTTPS A SIMPLIFY .....</b>	<b>6</b>
3.1	HttpAuth : Requête de Login.....	6
3.2	HttpSendFile : Requête d'envoi de fichier .....	7
3.3	HttpGetJobsInfo : Requête de récupération des informations de Jobs .....	10
3.4	HttpGetJobStatus : Récupération du statut d'un CR.....	12
3.5	HttpRecvFile : Requête de réception d'un fichier RTF .....	13
3.6	HttpGetDoctorModels : Requête de récupération des Modèles du Docteur .....	14
3.7	HttpSetJobQuality : Requête de contrôle qualité du CR.....	15
3.8	HttpGetJobQuality : Récupération du contrôle qualité du CR .....	17
3.9	HttpDisconnect : Requête de déconnexion .....	18
3.10	Exemples d'appel depuis un poste client en Java.....	18
<b>4</b>	<b>SUPPORT .....</b>	<b>19</b>

# 1 INTRODUCTION

Ce document décrit le protocole d'accès au service Simplify via une connexion HTTPS au travers de programmes. Ces accès se font sous forme d'API, via l'appel d'actions HTTPS.

Ce protocole permet à tout programme de communiquer à l'intérieur de programmes avec le serveur Simplify (<https://service.simplify.fr>) pour :

- S'authentifier auprès du Serveur Simplify.
- Envoyer un fichier de Compte Rendu Audio au serveur Simplify.
- Récupérer depuis le Serveur Simplify le Compte Rendu RTF tapé.
- Récupérer depuis le Serveur Simplify la liste de tous les Compte Rendu pour un Docteur.
- Se déconnecter du serveur Simplify.

Les accès client/serveur se font entièrement en HTTPS.

Le mode opératoire fonctionnel est équivalent à celui de l'interface utilisateur en mode Web.

## 1.1 Destinataires du document

Ce document s'adresse aux développeurs (Java, Visual Basic, C/C++, Delphi, PowerBuilder, 4D, etc.) qui souhaitent incorporer dans leur programmes une interface directe avec le service Simplify.

Une connaissance des notions de base du fonctionnement du Service Simplify en tant qu'utilisateur (Docteur) est supposé connue du lecteur.

## 1.2 Modifications par rapport à la version 1.1

Le code erreur **error\_invalid\_doctor\_name** a été rajouté. Il est renvoyé par les fonctions **HttpSendFile**, **HttpGetJobsInfo** et **HttpRecvFile** si un doctor\_name passé en paramètres ne correspond à aucun docteur existant sur le serveur.

## 1.3 Modifications par rapport à la version 1.2

Il est précisé que les requêtes doivent être au format UTF-8.

## 1.4 Modifications par rapport à la version 1.3

Ajout de l'API **HttpGetDoctorModels**.

*Cf. 3.6 [HttpGetDoctorModels](#) : Requête de récupération des Modèles du Docteur.*

## **1.5 Modifications par rapport à la version 1.4**

Ajout de l'API **HttpSetJobQuality** qui permet de définir et commenter la qualité d'un Compte Rendu rédigé par Simplify.

*Cf. 3.7 [HttpSetJobQuality](#) : Requête de contrôle qualité du CR*

## **1.6 Modifications par rapport à la version 1.5**

Ajout de l'API **HttpGetJobQuality** qui permet de récupérer la qualité marquée auparavant par **HttpSetJobQuality** d'un Compte Rendu rédigé par Simplify.

*Cf. 3.8 [HttpGetJobQuality](#) : Récupération du contrôle qualité du CR*

## **1.7 Modifications par rapport à la version 1.6**

Ajout de l'API **HttpGetJobStatus** qui permet de récupérer le statut d'un Job.

## 2 FONCTIONNEMENT

### 2.1 Généralités

Le protocole d'accès HTTP au service Simplify se présente sous la forme d'une liste de requêtes HTTP correspondant aux différentes actions disponibles.

Ces requêtes se présentent toutes sous la forme : <https://service.simplify.fr/Action>  
Chaque Action ayant une liste de paramètres spécifique.

### 2.2 Communications réseau

Les opérations d'envoi et de réceptions des Comptes Rendus sont exclusivement faites en mode HTTP et HTTPS. Il faut et il suffit simplement que les ports suivants soient ouverts sur le PC de l'ordinateur :

- Port 80.
- Port 443.

### 2.3 Chiffrement SSL des flux

Les flux sont tous chiffrés en SSL par le lien HTTPS entre le PC et le serveur Simplify.

***Aucune information (login/password de connexion, fichiers Audio, CR RTF, etc.) ne circule en clair entre le PC et le serveur Simplify.***

### 3 MANUEL DU PROTOCOLE D'ACCES HTTPS A SIMPLIFY

#### 3.1 HttpAuth : Requête de Login

Action	HttpAuth		
Adresse	<a href="https://service.simplify.fr/simplify/HttpAuth">https://service.simplify.fr/simplify/HttpAuth</a>		
Type	GET ou POST		
Description	Login & Authentification en SSL de l'utilisateur auprès du Serveur Simplify.		
Paramètres de la requête	text	login	Login du Docteur
	text	password	Password du Docteur
Format de la réponse	text/plain	La réponse contient une des chaînes suivantes : <ul style="list-style-type: none"> <li>• token alphanumérique d'authentification.</li> <li>• <b>error_invalid_login</b> en cas d'erreur sur le login/password.</li> <li>• <b>error_stack_trace</b> suivi de la stack trace de l'exception si une exception s'est produite lors du traitement de la requête.</li> </ul>	

**Note :**

- Les logins et passwords ne doivent pas être hardcodés et sont susceptibles d'être modifiés à tout moment par Simplify. Ils doivent être demandés aux Utilisateurs/Docteurs lors de l'authentification.
- Le login utilisé pour l'authentification et le token de retour doivent être conservés en mémoire pendant la session, ils sont nécessaires à l'exécution des autres requêtes.

- |   |
|---|
| <ul style="list-style-type: none"> <li>• Cette requête ainsi que toutes les autres requêtes doit être formatée en UTF-8.</li> </ul> |
|---|

### **3.2 HttpSendFile : Requête d'envoi de fichier**

Cette requête permet d'envoyer un fichier audio au Serveur Simplify pour traitement par un Opérateur.

Il est également possible de joindre à la requête un ou plusieurs Modèle de documents qui seront associés au fichier audio. Les Modèles doivent être envoyés au format RTF.

La taille maximum de chaque fichier envoyé est de 25 Mo.  
La taille totale des fichiers envoyés ne doit pas dépasser 50 Mo.

#### **La requête doit être au format Multipart POST et UTF-8**

La requête renvoie un « Job Id » qui est un numéro unique affecté au Compte Rendu à venir. C'est le Job Id qui sera utilisé pour récupérer ultérieurement le Compte Rendu tapé au format RTF.

Action	HttpSendFile		
Adresse	<a href="https://service.simplify.fr/simplify/HttpSendFile">https://service.simplify.fr/simplify/HttpSendFile</a>		
Type	Multipart POST		
Description	Envoi en SSL au serveur Simplify d'un fichier audio avec un libellé.		
Paramètres de la requête	text	login	Login utilisé lors de l'appel de HttpAuth
	text	token	Token d'authentification retourné par l'appel de HttpAuth
	text	doctor_name	Nom du Docteur. Doit exister au préalable et être unique dans le système.
	text	label	Libellé du fichier peut être nul.
	file	audioFile	Le fichier audio
	file	modelFile1	Le premier Modèle associé au fichier audio ( <i>facultatif</i> )
	file	modelFileN	Le nième Modèle associé au fichier audio ( <i>facultatif</i> )
Format de la réponse	text/plain	Code Retour : <ul style="list-style-type: none"> <li>• Numéro de Job &gt; 0 affecté au Compte Rendu si l'envoi s'est produit correctement</li> <li>• <b>error_invalid_login</b> si le login et/ou le token ne correspondent pas à ceux utilisés/retournés par HttpAuth.</li> <li>• <b>error_invalid_doctor_name</b> si le doctor_name n'existe pas dans le système.</li> <li>• <b>error_invalid_request</b> si les paramètres ne sont pas tous renseignés.</li> <li>• <b>error_maximum_authorized_file_length_exceeded</b> si le fichier envoyé est trop lourd.</li> <li>• <b>error_invalid_extension</b> si l'extension du fichier n'est pas supportée par le serveur.</li> <li>• <b>error_file_upload_size_zero</b> si la taille du fichier uploadé est nulle.</li> <li>• <b>error_stack_trace</b> suivi de la stack trace de l'exception si une exception s'est produite lors du traitement de la requête.</li> </ul>	

Note :

- Il est indispensable de garantir l'unicité du Nom de Docteur, c'est pourquoi nous recommandons l'utilisation le numéro d'identifiant national des médecins pour doctor\_name.
- le doctor\_name doit existe au préalable dans la base Simplify. Veuillez SVP contacter Simplify pour insérer les doctor\_name nécessaires.
- La taille maximum des fichiers audio est limitée à 25 Mo.

## Format Multipart POST de la requête

La requête doit être équivalente à un formulaire HTML de type « Multipart Post » et « UTF-8 » comme suit :

```
<html>
  <head>
    <title>Simplify HttpSendFile Example</title>
  </head>
  <body>
    <center>

      Envoi d'un compte rendu sur le serveur Simplify avec HttpSendFile
    <br><br>

      <!--
      Commentaires :
      Le programme client devra émuler le formulaire post suivant.
      Noter que le type est :
      enctype="multipart/form-data" et acceptcharset="UTF-8
      -->

      <form action="http://service.simplify.fr/simplify/HttpSendFile"
        method="post"
        enctype="multipart/form-data" acceptcharset="UTF-8">

        <br> login      : <input type="text" size=30 name="login">
        <br> token      : <input type="text" size=30 name="token">
        <br> doctor_name : <input type="text" size=30 name="doctor_name">
        <br> label       : <input type="text" size=30 name="label">
        <br> file        : <input type="file" size=30 name="file"><br>

        <br> <input type="submit" value="Envoyer le fichier">

      </form>
    </center>
  </body>
</html>
```

Vous pouvez, pour vos tests, accéder à ce formulaire à l'adresse : <http://service.simplify.fr/simplify/HttpSendFileFormulaire.html>.

### 3.3 HttpGetJobsInfo : Requête de récupération des informations de Jobs

Cette requête permet de récupérer une liste de Jobs et informations associées suivant différents critères. Les infos sont récupérées sous la forme d'une suite de lignes, chaque ligne correspondant à un job. Les informations sont séparées par « ; » :

**Job Id ; Statut ; Label ; Date De Début ; Date de Fin ; Taille du fichier RTF**

Action	HttpGetJobsInfo		
Adresse	<a href="https://service.simplify.fr/simplify/HttpGetJobsInfo">https://service.simplify.fr/simplify/HttpGetJobsInfo</a>		
Type Appel	GET ou POST		
Description	Récupération d'une liste de Jobs affectés à un Docteur.		
Paramètres de la requête	text	login	Login utilisé lors de l'appel de HttpAuth
	text	token	Token retourné par l'appel de HttpAuth
	text	doctor_name	Nom du Docteur « propriétaire » des Jobs.
	text	status	<ul style="list-style-type: none"> <li>• <b>status_all</b> pour tous les statuts.</li> <li>• <b>status_waiting_to_be_processed</b></li> <li>• <b>status_being_processed</b></li> <li>• <b>status_waiting_to_be_controlled</b></li> <li>• <b>status_being_controlled</b></li> <li>• <b>status_ready</b></li> <li>• <b>status_archived</b></li> </ul>
	text	start_date	Date au format AAAAMMJJ d'envoi sur le Serveur du fichier Audio
	text	end_date	Date au format AAAAMMJJ de traitement du Compte Rendu sur le Serveur.
	text	offset	Rang à partir de 0. (Valeur numérique).
	text	limit	Nombre de maximum lignes à récupérer (Valeur numérique)
Format de la réponse	text/plain	<ul style="list-style-type: none"> <li>• Liste des jobs correspondant à la requête (un par ligne)</li> <li>• <b>error_invalid_login</b> si le login et/ou le token ne correspondent pas à ceux utilisés/retournés par HttpAuth.</li> <li>• <b>error_invalid_doctor_name</b> si le doctor_name n'existe pas dans le système.</li> <li>• <b>error_invalid_request</b> si les paramètres ne sont pas tous renseignés ou si une valeur numérique est mal renseignée.</li> <li>• <b>error_status_unknown</b> si le statut est invalide.</li> <li>• <b>error_invalid_startdate</b> si la date de début est invalide.</li> <li>• <b>error_invalid_enddate</b> si la date de fin est invalide.</li> <li>• <b>error_stack_trace</b> suivi de la stack trace de l'exception si une exception s'est produite lors du traitement de la requête.</li> </ul>	

Voici les valeurs par défaut pour les paramètres non renseignés :

Argument	Valeur par défaut si non renseigné (champ vide ou à 0)
status	STATUS_ALL
start_date	aujourd'hui.
end_date	aujourd'hui.
offset	0
limit	1000

Note :

- La date de fin retournée est **null** si le fichier n'est pas encore en traitement.

### 3.4 HttpGetJobStatus : Récupération du statut d'un CR

Cette requête permet de récupérer le statut d'un Compte Rendu RTF correspondant à un fichier audio traité par Simplify et envoyé avec HttpSendFile.

Action	HttpGetStatus		
Adresse	<a href="https://service.simplify.fr/simplify/HttpGetJobStatus">https://service.simplify.fr/simplify/HttpGetJobStatus</a>		
Type Appel	GET ou POST		
Description	Récupération du statut de traitement Compte Rendu sur le Serveur Simplify.		
Paramètres de la requête	text	login	Login utilisé lors de l'appel de HttpAuth
	text	token	Token retourné par l'appel de HttpAuth
	text	doctor_name	Nom du Docteur « propriétaire » du Compte Rendu et du Job Id.
	text	job_id	Numéro de Job (affecté par un précédent appel par httpsSendFile).
Format de la réponse	text/plain	<ul style="list-style-type: none"> <li>• <b>status_waiting_to_be_processed</b></li> <li>• <b>status_being_processed</b></li> <li>• <b>status_waiting_to_be_controlled</b></li> <li>• <b>status_being_controlled</b></li> <li>• <b>status_ready</b></li> <li>• <b>status_archived</b></li> </ul> <p>Si une erreur survient:</p> <ul style="list-style-type: none"> <li>• <b>error_status_unknown</b> si le statut est inconnu.</li> <li>• <b>error_invalid_jobid</b> si le job id n'existe pas ou n'est pas affecté au docteur passé en paramètre.</li> <li>• <b>error_invalid_login</b> si le login et/ou le token ne correspondent pas à ceux utilisés/retournés par HttpAuth.</li> <li>• <b>error_invalid_doctor_name</b> si le doctor_name n'existe pas dans le système.</li> <li>• <b>error_stack_trace</b> suivi de la stack trace de l'exception si une exception s'est produite lors du traitement de la requête.</li> </ul>	

### 3.5 HttpRecvFile : Requête de réception d'un fichier RTF

Cette requête permet de récupérer un Compte Rendu RTF correspondant à un fichier audio traité par Simplify et envoyé avec HttpSendFile.

Action	HttpRecvFile		
Adresse	<a href="https://service.simplify.fr/simplify/HttpRecvFile">https://service.simplify.fr/simplify/HttpRecvFile</a>		
Type Appel	GET ou POST		
Description	Récupération d'un Compte Rendu au format RTF à partir du Serveur Simplify.		
Paramètres de la requête	text	login	Login utilisé lors de l'appel de HttpAuth
	text	token	Token retourné par l'appel de HttpAuth
	text	doctor_name	Nom du Docteur « propriétaire » du Compte Rendu et du Job Id.
	text	job_id	Numéro de Job (affecté par un précédent appel par httpsSendFile).
Format de la réponse	application/rtf	Le flux correspondant au fichier RTF du compte rendu demandé.	
	text/plain	<p>Si le compte rendu n'est pas prêt:</p> <ul style="list-style-type: none"> <li>• <b>status_waiting_to_be_processed</b></li> <li>• <b>status_being_processed</b></li> <li>• <b>status_waiting_to_be_controlled</b></li> <li>• <b>status_being_controlled</b></li> </ul> <p>Si une erreur survient:</p> <ul style="list-style-type: none"> <li>• <b>error_status_unknown</b> si le statut est inconnu.</li> <li>• <b>error_invalid_jobid</b> si le job id n'existe pas ou n'est pas affecté au docteur passé en paramètre.</li> <li>• <b>error_invalid_login</b> si le login et/ou le token ne correspondent pas à ceux utilisés/retournés par HttpAuth.</li> <li>• <b>error_invalid_doctor_name</b> si le doctor_name n'existe pas dans le système.</li> <li>• <b>error_stack_trace</b> suivi de la stack trace de l'exception si une exception s'est produite lors du traitement de la requête.</li> </ul>	

Note :

- Le type de réponse est différent suivant que le fichier est créé (application/rtf) ou pas (text/plain avec contenu du statut ou de l'erreur).

### 3.6 HttpGetDoctorModels : Requête de récupération des Modèles du Docteur

Cette requête permet de récupérer la liste des Modèles d'un Docteur.

Les infos sont récupérées sous la forme d'une suite de lignes, chaque ligne correspondant à un Modèle. Les informations sont séparées par « ; » :

**Nom du Modèle ; URL du Modèle**

L'URL du Modèle permet de télécharger sous forme de GET ou POST le fichier du Modèle.

Action	HttpGetDoctorModels		
Adresse	<a href="https://service.simplify.fr/simplify/HttpGetDoctorModels">https://service.simplify.fr/simplify/HttpGetDoctorModels</a>		
Type Appel	GET ou POST		
Description	Récupération d'une liste des Modèles d'un Docteur.		
Paramètres de la requête	text	login	Login utilisé lors de l'appel de HttpAuth
	text	token	Token retourné par l'appel de HttpAuth
	text	doctor_name	Nom du Docteur « propriétaire » des Modèles
Format de la réponse	text/plain	<ul style="list-style-type: none"> <li>Liste des Modèles associés à Docteur</li> <li><b>error_invalid_login</b> si le login et/ou le token ne correspondent pas à ceux utilisés/retournés par HttpAuth.</li> <li><b>error_invalid_request</b> si les paramètres ne sont pas tous renseignés ou si une valeur numérique est mal renseignée.</li> <li><b>error_stack_trace</b> suivi de la stack trace de l'exception si une exception s'est produite lors du traitement de la requête.</li> </ul>	

Exemple de réponse :

```
operatoire;https://service.simplify.fr/simplify/models_mobile/model_131_10.doc
principal;https://service.simplify.fr/simplify/models_mobile/model_131_14.doc
observation;https://service.simplify.fr/simplify/models_mobile/model_131_12.doc
```

### 3.7 HttpSetJobQuality : Requête de contrôle qualité du CR

Cette requête permet de spécifier et commenter, pour un Job / CR, la qualité du résultat du fichier Word remis par Simplify.

L'appel à HttpSetJobQuality est facultatif : par défaut la qualité d'un CR est OK (quality\_ok).

L'API permet de passer en option, un commentaire sous forme de champ texte au format brut.

Action	HttpSetJobQuality		
Adresse	<a href="https://service.simplify.fr/simplify/HttpSetJobQuality">https://service.simplify.fr/simplify/HttpSetJobQuality</a>		
Type Appel	GET ou POST		
Description	Récupération d'une liste des Modèles d'un Docteur.		
Paramètres de la requête	text	login	Login utilisé lors de l'appel de HttpAuth
	text	token	Token retourné par l'appel de HttpAuth
	text	doctor_name	Nom du Docteur « propriétaire » du Compte Rendu et du Job Id.
	text	job_id	Numéro de Job (affecté par un précédent appel par httpsSendFile).
	text	quality	<ul style="list-style-type: none"> <li>• <b>quality_ok</b></li> <li>• <b>quality_bad_presentation</b></li> <li>• <b>quality_bad_spelling</b></li> <li>• <b>quality_serious_error</b></li> <li>• <b>quality_nonsense</b></li> <li>• <b>quality_other</b></li> </ul>
	text	comments	Commentaires libres (longueur < 2048 caractères). Paramètre facultatif.
Format de la réponse	text/plain	<ul style="list-style-type: none"> <li>• <b>ok</b> si la requête a été acceptée.</li> <li>• <b>error_invalid_login</b> si le login et/ou le token ne correspondent pas à ceux utilisés/retournés par HttpAuth.</li> <li>• <b>error_invalid_quality</b> si le paramètre quality n'a pas une valeurs définies ci-dessous.</li> <li>• <b>error_stack_trace</b> suivi de la stack trace de l'exception si une exception s'est produite lors du traitement de la requête.</li> </ul>	

Voici les significations du paramètre **quality**.

Valeur	Signification
quality_ok	La qualité du CR est satisfaisante (valeur pas défaut).
quality_bad_presentation	La présentation n'est pas conforme ou satisfaisante.
quality_bad_spelling	Le CR contient des fautes d'orthographe.
quality_serious_error	Le CR contient des erreurs graves.
quality_nonsense	Le CR contient des parties de non-sens.
quality_other	Autre problème de qualité.

### 3.8 HttpGetJobQuality : Récupération du contrôle qualité du CR

Cette requête permet récupérer pour un Job / CR, la qualité du résultat du fichier Word remis par Simplify telle que taguée auparavant avec l'API HttpSetJobQuality.

Action	HttpSetJobQuality		
Adresse	<a href="https://service.simplify.fr/simplify/HttpGetJobQuality">https://service.simplify.fr/simplify/HttpGetJobQuality</a>		
Type Appel	GET ou POST		
Description	Récupération d'une liste des Modèles d'un Docteur.		
Paramètres de la requête	text	login	Login utilisé lors de l'appel de HttpAuth
	text	token	Token retourné par l'appel de HttpAuth
	text	doctor_name	Nom du Docteur « propriétaire » du Compte Rendu et du Job Id.
	text	job_id	Numéro de Job (affecté par un précédent appel par httpsSendFile).
Format de la réponse	text/plain	<ul style="list-style-type: none"> <li>• Si la requête est OK, une chaîne de format : <b>ok;&lt;quality&gt;;&lt;comments&gt;</b></li> <li>• <b>error_invalid_login</b> si le login et/ou le token ne correspondent pas à ceux utilisés/retournés par HttpAuth.</li> <li>• <b>error_stack_trace</b> suivi de la stack trace de l'exception si une exception s'est produite lors du traitement de la requête.</li> </ul>	

Notes sur la chaîne retournée **ok;<quality>;<comments>**

- Les valeurs de **quality** sont décrites dans la page précédente. La valeur par défaut est **quality\_ok** si le CR n'a pas été tagué par HttpSetJobQuality.
- Les accents et caractères spéciaux des commentaires **comments** sont au format HTML. **comments** est à **null** si aucun commentaire existe ou si la valeur de **quality** est **quality\_ok**

### 3.9 HttpDisconnect : Requête de déconnexion

Cette requête permet de se déconnecter immédiatement du Service Simplify.

Action	HttpDisconnect	
Adresse	<a href="https://service.simplify.fr/simplify/HttpDisconnect">https://service.simplify.fr/simplify/HttpDisconnect</a>	
Type Appel	GET ou POST	
Description	Déconnexion du programme utilisateur du Serveur Simplify.	
Paramètres de la requête	text	login
	text	token
Format de la réponse	text/plain	<ul style="list-style-type: none"> <li>• <b>ok</b> si la déconnexion est effective</li> <li>• <b>error_invalid_login</b> si le login et/ou le token ne correspondent pas à ceux utilisés/retournés par HttpAuth</li> <li>• <b>error_stack_trace</b> suivi de la stack trace de l'exception si une exception s'est produite lors du traitement de la requête.</li> </ul>

### 3.10 Exemples d'appel depuis un poste client en Java

Le projet Eclipse **simplify\_http\_api.zip** contient le code source et les libraires d'une classe exemple qui couvre toutes les APIs.

Le projet Eclipse est accessible ici :

[https://service.simplify.fr/simplify/download/http\\_api/simplify\\_http\\_api.zip](https://service.simplify.fr/simplify/download/http_api/simplify_http_api.zip)

Le code source de la classe Java est accessible ici :

[https://service.simplify.fr/simplify/download/http\\_api/HttpAPITest.java](https://service.simplify.fr/simplify/download/http_api/HttpAPITest.java)

## 4 SUPPORT

Pour toute question technique, veuillez vous adresser au partenaire technique de Simplify qui gère le service Simplify :

KawanSoft SAS  
62, boulevard Pereire  
75017 Paris, France  
Tel: +33 (0)1 77 69 59 58  
Email : [contact@kawansoft.com](mailto:contact@kawansoft.com)  
Web: <http://www.kawansoft.com>

---