

# Service Postal



## API d'envoi de courrier Service Postal Kit de développement PHP

Version 11.3 – 23/10/2015

# Table des matières

1	Vue d'ensemble.....	3
2	Pré-requis et informations techniques.....	3
3	Déroulé d'une session de travail Service Postal .....	4
3.1	Connexion – Authentification et Initialisation.....	4
3.2	Envoi et/ou suivi de courriers.....	4
3.3	Déconnexion.....	4
4	Exemple de code : envoi de courrier (basique).....	4
5	Envoi d'une lettre : déroulé pas à pas .....	5
5.1	Préparation de la lettre .....	5
5.2	Envoi de la lettre (direct ou en 2 temps).....	7
5.3	Suivi de la lettre.....	7
5.4	Annulation d'un courrier .....	9
6	Envoi d'un mailing : déroulé pas à pas .....	10
6.1	Préparation du mailing.....	10
6.2	Envoi du mailing (direct ou en 2 temps).....	10
6.3	Suivi du mailing .....	11
6.4	Annulation d'un mailing .....	11
7	Gestion des envois : déroulé pas à pas .....	12
7.1	Suivi d'un job .....	12
7.2	Suivi global.....	12
7.3	Annulation d'un job.....	12
8	Mailing : modèle de document et fichier de données .....	13
9	Liste des codes produits.....	14
10	FAQ – Questions fréquentes .....	15

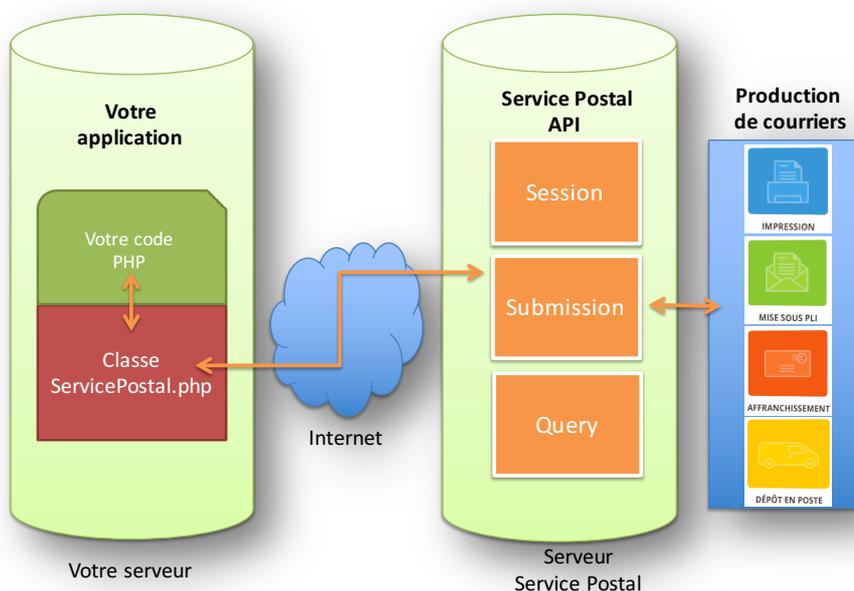
# 1 Vue d'ensemble

Service Postal met à disposition une API complète sous la forme d'un web service SOAP. Le détail exhaustif des fonctionnalités est décrit dans le guide de référence (en Anglais).

Afin de faciliter l'utilisation de l'API, Service Postal a développé un **kit de développement PHP** qui comprend :

- un ensemble d'objets PHP qui prend en charge la communication avec l'API
- des exemples de codes complets et fonctionnels
- le présent document qui décrit pas à pas l'utilisation du kit de développement

**Le kit de développement permet de gagner du temps et de se concentrer sur les aspects fonctionnels**



## 2 Pré-requis et informations techniques

Le kit de développement PHP est basé sur PHP version 5.6 ou supérieur.

Le kit de développement PHP est formé de plusieurs classes qu'il faut installer sur votre serveur, puis il faut inclure le fichier ServicePostal.php (qui inclura les autres fichiers qui doivent être dans le même dossier).

Le kit de développement permet d'accéder de manière simple à toutes les fonctions de l'API. Il est possible d'accéder à l'API : référez-vous au guide de référence de l'API (en anglais) et créez votre propre code PHP.

Le kit de développement PHP utilise le namespace : SP. Notre code respecte la norme de documentation phpDoc : si votre éditeur de code le supporte, vous bénéficierez de l'aide contextuelle et la complétion de code.

Chaque méthode réalise contrôle le résultat attendu ; si une erreur survient, une exception est générée et un message d'erreur est relayé dans l'exception. Il est recommandé de capturer les exception, par exemple :

```
try //Initialisation du service
{
    $sp_session = new SP\Session("login", "password", TRUE);
}
catch (\Exception $e)
{
    die("Erreur lors de l'initialisation de Service Postal : ".$e->getMessage() );
}
```

Remarque : pour faciliter la lecture de ce document, les exemples n'incluent pas de gestion des exceptions.

Pour faciliter le développement, chaque option appartient à une classe et peut être utilisée avec cette syntaxe : \SP\Options\NomClasse::NOM\_CONSTANTE, par exemple : \SP\Options\Affranchissement::LETTRE\_RECOMMANDEE

# 3 Déroulé d'une session de travail Service Postal

Toute session de travail comporte 3 étapes :

## 3.1 Connexion – Authentification et Initialisation

Instancier la classe `\SP\Session` en fournissant en paramètres : login ServicePostal, mot de passe ServicePostal, et un booléen indiquant si on est en test (TRUE) ou en production (par défaut ou FALSE)

```
$sp_session = new SP\Session(
    "login",           // login de l'API
    "mot de passe",  // mot de passe de l'API
    TRUE              // mode test (TRUE) ou mode production (FALSE ou vide)
);
```

L'objet `SP\Session` est disponible pour la durée de la session et servira de point de départ pour d'autres fonctions.

## 3.2 Envoi et/ou suivi de courriers

Pour créer et gérer un nouveau courrier, il faut créer un objet job :

```
$ma_lettre = $spSession->nouveauLettreJob() ; // envoyer une lettre
```

ou

```
$mon_mailing = $spSession->nouveauMailingJob() ; // envoyer un mailing
```

Si on souhaite suivre ou gérer un envoi déjà transmis à la plate-forme, plusieurs possibilités existent :

```
$ma_lettre = $spSession->chargeLettreJob( $jobID ) ; // gérer une lettre
```

ou

```
$mon_mailing = $spSession->chargeMailingJob( $jobID ) ; // gérer un mailing
```

ou

```
$ma_lettre = $spSession->chargeLettreFromMailingJob( $jobID, $index ) ; // gérer une des lettres d'un mailing
```

## 3.3 Déconnexion

```
$spSession->logout();
```

# 4 Exemple de code : envoi de courrier (basique)

Le kit du développeur permet d'envoyer un courrier postal en quelques lignes de code. Voici la version la plus concise d'un envoi :

```
require_once 'lib/vendor/ServicePostal.php';

// Authentification
$maSessionSP = new SP\Session("login", "mot de passe", TRUE);

// Création d'un nouvel envoi (job)
$maLettre = $maSessionSP->nouveauLettreJob()
    ->setImpression(SP\Options\Couleur::COULEUR,
        SP\Options\Enveloppe::AUTO,
        SP\Options\Recto::RECTO,
        SP\Options\PorteAdresse::ACTIF)
    ->setAffranchissement(SP\Options\Affranchissement::LETTRE_RECOMMANDEE)
    ->setReferenceExterne("2016-12346abc")
    ->setExpéditeur("Ronan", "PAUL", "", "9 rue Ambroise Thomas", "", "75009", "Paris")
    ->setDestinataire("", "", "Service Postal", "4 rue Edouard Branly", "", "78190", "Trappes")
    ->setDocument("/usr/local/apache/htdocs/myproject/docs/document.docx");

$result = $maLettre->envoyerDirectement();
echo "Le courrier a été envoyé à Service postal et porte le numéro ", $maLettre->jobID;

// Sauvegarder l'identifiant service postal ( $maLettre->jobID ) dans votre base de données
$maSessionSP ->logout();
```

# 5 Envoi d'une lettre : déroulé pas à pas

## 5.1 Préparation de la lettre

### 5.1.1 Création de l'objet lettreJob

```
$maLettre = $spSession->nouveauLettreJob() ;
```

Il est possible de créer un ou plusieurs objets avec la méthode `nouveauLettreJob()`.

### 5.1.2 Définir les paramètres d'impression

```
$maLettre->setImpression(  
    SP\Options\Couleur::COULEUR,           // Couleur du courrier  
    SP\Options\Enveloppe::AUTO,           // Taille de l'enveloppe  
    SP\Options\Recto::RECTO_VERSO,       // Impression recto ou recto/verso  
    SP\Options\PorteAdresse::ACTIF       // Porte-adresse (optionnel)  
);
```

Les paramètres d'impression sont définis parmi les constantes suivantes :

#### Classe SP\Options\Couleur

Valeurs	Description
NOIR_ET_BLANC	Impression noir et blanc
COULEUR	Impression couleur

#### Classe SP\Options\Enveloppe

Valeurs	Description
AUTO	Taille d'enveloppe automatique en fonction du nombre de feuilles : C6 jusqu'à 3, C5 jusqu'à 8, C4 sinon.
C6	Enveloppe C6 (ou DL) : A4 plié en 3. 3 feuilles maximum. Inclus.
C5	Enveloppe C5 : A4 plié en 2. 8 feuilles maximum. Coût spécifique.
C4	Enveloppe C4 : A4 non plié. 75 feuilles maximum. Coût spécifique.

#### Classe SP\Options\Recto

Valeurs	Description
RECTO	Impression recto
RECTO_VERSO	Impression recto/verso

#### Classe SP\Options\PorteAdresse

Valeurs	Description
ACTIF	Option activée
INACTIF	Option non activée

#### Classe SP\Options\FormatPapier

Remarque : cette option n'est pas disponible pour le moment.

Valeurs	Description
A4	Papier format A4
A3	Papier format A3

### 5.1.3 Définir le type d'affranchissement

```
$maLettre->setAffranchissement( SP\Options\Affranchissement::LETTRE_RECOMMANDEE );
```

Le type d'affranchissement est l'une des constantes suivantes :

#### Classe SP\Options\Affranchissement

Valeurs	Description
LETTRE_RECOMMANDEE	Lettre recommandée (sans accusé de réception)
LETTRE_RECOMMANDEE_AVEC_AR	Lettre recommandée avec accusé de réception
LETTRE_PRIORITAIRE	Lettre prioritaire
LETTRE_ECOPLI	Lettre écopli
LETTRE_VERTE	Lettre verte

### 5.1.4 Définir l'expéditeur

```
$maLettre->setExpéditeur(  
    "Ronan", // Prénom  
    "PAUL", // Nom de famille  
    "Service Postal", // Société  
    "9 rue Ambroise Thomas", // Adresse 1 (obligatoire)  
    "Bâtiment B", // Adresse 2 (optionnel)  
    "13858", // Code postal (obligatoire)  
    "Paris", // Ville (obligatoire)  
    "France" // Pays (optionnel)  
)
```

L'expéditeur est obligatoire. Il faut au moins un nom/prénom ou un nom de société (ou les 2), une ligne d'adresse, un code postal et une ville. La 2ème ligne d'adresse peut être vide. Le pays peut être vide (France par défaut).

### 5.1.5 Définir le destinataire

```
$maLettre->setDestinataire(  
    "Jovana", // Prénom  
    "Stamenkivic", // Nom de famille  
    "Net4Com", // Société  
    "4 rue Edouard Branly", // Adresse 1 (obligatoire)  
    "Zone d'activité de la louvère", // Adresse 2 : (optionnel)  
    "78190", // Code postal (obligatoire)  
    "Trappes", // Ville (obligatoire)  
    "France" // Pays (optionnel)  
);
```

Le destinataire est obligatoire. Il faut au moins un nom/prénom ou un nom de société (ou les 2), une ligne d'adresse, un code postal et une ville. La 2ème ligne d'adresse peut être vide. Le pays peut être vide (France par défaut).

### 5.1.6 Ajout du document à envoyer

```
$maLettre->setDocument("https://www.servicepostal.com/Images/service-postal.png");
```

Le document peut être une ressource locale (sur votre serveur) ou distante (une URL publique).

De nombreux types de document sont acceptés, et notamment .doc, .docx, .pdf, .png, .jpg, etc.

L'envoi du document est géré de manière transparente selon la méthode appropriée de l'API.

### 5.1.7 Ajout d'une référence externe

```
$maLettre->setReferenceExterne( "2016-123456ab" ); // Référence externe (optionnel)
```

La référence externe est optionnelle, elle correspond à votre référence pour cet envoi. Cette valeur sera disponible dans les différents outils de suivi fournis par Service Postal.

### 5.1.8 Obtenir le coût d'un envoi (optionnel)

Cette fonction permet de connaître le coût estimé d'un envoi avant de le valider.

Elle se base sur un nombre de pages fourni en paramètre, et non sur le contenu des documents.

## 5.2 Envoi de la lettre (direct ou en 2 temps)

Il existe 2 modes d'envoi du courrier : l'envoi direct et l'envoi en 2 temps (préparation, validation).

### 5.2.1 Option 1 : envoi en 2 temps (prévisualisation puis validation)

Une 1ere étape transmet le courrier à la plate-forme ServicePostal.

```
$letterPreviewResult = $maLettre->preparer();
```

Le job obtient un identifiant (que vous devrez enregistrer dans votre base de données) :

```
echo "Votre document porte le numéro :", $maLettre->jobID ;
```

Vous pouvez pré-visualiser le document PDF tel qu'il sera envoyé par Service Postal, grâce au lien fourni :

```
echo " et peut être <a href='", $letterPreviewResult->spOutputFile->spURL, "'>pré-visualisé</a><br />" ;
```

Vous pouvez déterminer quand le courrier sera produit par Service Postal :

```
if ($letterPreviewResult->spExpectedDispatchNotice == 0)
    echo "et sera produit aujourd'hui <br />" ;
else
    echo "et sera produit dans ", $letterPreviewResult->spExpectedDispatchNotice , "jour(s)<br />" ;
```

Le job aura un statut SP\_PREVIEWED et ne sera pas produit tant qu'il n'aura pas été validé (voir ci-dessous).  
pour le valider :

```
$letterValidResult = $maLettre ->valider();
```

```
echo "Votre document va être envoyé, le coût sera de : ", $letterValidResult->spTotalCost, " €.<br />" ;
```

Le job aura alors le statut SP\_SUBMITTED et est prêt à être produit.

### 5.2.2 Option 2 : envoi direct

Il est possible d'envoyer le job directement, sans étape de préparation, le job aura un statut SP\_SUBMITTED :

```
$result = $maLettre->envoyerDirectement();
```

```
echo "Le courrier a été envoyé sur la plate-forme Service postal et porte le numéro ", $maLettre->jobID;
```

Pour pouvoir gérer ce courrier par la suite, vous devez enregistrer son identifiant ServicePostal dans votre application.

## 5.3 Suivi de la lettre

### 5.3.1 Accès à une lettre transmise

```
$maLettre = $spSession->chargeLettreJob( $jobID ) ;
```

A partir de la référence Service Postal (en général sauveé dans votre base de données), vous pouvez charger une lettre déjà transmise.

### 5.3.2 Statut de l'envoi

```
$queryStatusResult = $maLettre->queryStatut();
```

```
echo "Le statut de l'envoi est : ", $queryStatusResult->spStatus, "<br />" ;
```

Les différents statuts possibles sont :

#### Classe SP\Options\DocumentStatut

Enum values	Description
SP_PREVIEWED	Le courrier a été transmis
SP_SUBMITTED	Le courrier a été validé et est prêt à être produit
SP_PRINTED	Le courrier a été imprimé
SP_SCANNED	Le courrier a été scanné après avoir été imprimé et mis sous enveloppe
SP_DELIVERED	Le courrier a été délivré à son destinataire (uniquement pour les courriers recommandés)
SP_DEPOSIT_SLIP_RECEIVED	La preuve de réception a été reçue par service

### 5.3.3

	postal
SP_UNKNOWN	Le courrier n'a pas été trouvé dans notre plateforme



# 6 Envoi d'un mailing : déroulé pas à pas

Service Postal peut se charger de composer vos courriers : vous fournissez un fichier modèle et un fichier de données, et nous créons un job qui va contenir autant de courrier que de destinataires, avec un document que nous allons personnaliser à chaque destinataire (coordonnées, mais ainsi que le contenu).

## 6.1 Préparation du mailing

### 6.1.1 Création de l'objet mailingJob

```
$monMailing = $maSessionSP->nouveauMailingJob();
```

Il est possible de créer un ou plusieurs objets « mailingJob ». Tout l'envoi sera piloté avec cet objet.

### 6.1.2 Définir les paramètres d'impression

```
$monMailing->setImpression(  
    SP\Options\Couleur::COULEUR,           // Couleur du courrier  
    SP\Options\Enveloppe::AUTO,           // Taille de l'enveloppe  
    SP\Options\Recto::RECTO_VERSO,       // Impression recto ou recto/verso  
    SP\Options\PorteAdresse::ACTIF       // Porte-adresse (optionnel)  
);
```

Les options d'impression sont détaillées dans le paragraphe 5.1.2 - Définir les paramètres d'impression.

### 6.1.3 Définir le type d'affranchissement

```
$monMailing->setAffranchissement( SP\Options\Affranchissement::LETTRE_RECOMMANDEE );
```

Les options d'affranchissement sont détaillées dans le paragraphe 5.1.3 - Définir le type d'affranchissement.

### 6.1.4 Définir l'expéditeur et les destinataires

En mode mailing, il faut ajouter un fichier de donnée qui décrira l'ensemble des expéditeurs/destinataires. La commande pour envoyer le fichier est :

```
$monMailing->setDocumentModele("/usr/local/apache/htdocs/myproject/docs/document_modele.docx ")
```

Le format du fichier de données est :

### 6.1.5 Ajout du document modèle lettre type

```
$monMailing->setDocumentDonnees("/usr/local/apache/htdocs/myproject/docs/donnees.csv");
```

Le document peut être une ressource locale ou distante (URL). De nombreux types de document sont acceptés, et notamment .doc, .docx, .pdf, .png, .jpg, etc.

Remarque : à ce stade, le document n'est pas encore envoyé au serveur.

### 6.1.6 Ajout d'une référence externe

La référence externe est propre à chaque destinataire, elle est donc à inclure dans le fichier de donnée (voir ci-dessus). Cette référence sera disponible dans les différents outils de suivi fournis par Service Postal.

### 6.1.7 Obtenir le coût d'un envoi (optionnel)

Cette fonction permet de connaître le coût estimé d'un envoi avant de le valider.

Elle se base sur un nombre de pages fourni en paramètre, et non sur le contenu des documents.

## 6.2 Envoi du mailing (direct ou en 2 temps)

Il existe 2 modes d'envoi du courrier : l'envoi direct et l'envoi en 2 temps (préparation, validation).

### 6.2.1 Option 1 : envoi du courrier en 2 temps (prévisualisation puis validation)

Une 1ere étape transmet le courrier à la plate-forme ServicePostal.

```
$letterPreviewResult = $monMailing->preparer();
```

Le job obtient alors un identifiant que vous devrez enregistrer dans votre base de données. Il est possible de prévisualiser le document

```
echo "Votre document porte le numéro :", $monMailing->jobID ;
```

```
echo " et peut être <a href='\" , $letterPreviewResult->spOutputFile->spURL, "\">pré-visualisé</a><br />";
```

Le job aura un statut SP\_PREVIEWED. Vous pouvez vérifier le statut du job à tout moment :

```
$queryStatusResult = $monMailing->queryStatut();  
echo "Le statut de l'envoi est : ", $queryStatusResult->spStatus, "<br />";
```

Après avoir vérifié le document vous pouvez valider le job, qui aura alors le statut SP\_SUBMITTED :

```
$letterValidResult = $monMailing->valider();  
echo "<h1>Validation</h1>";  
echo "Votre document va être envoyé, le coût sera de : ", $letterValidResult->spTotalCost, " €.<br />";
```

## 6.2.2 Option 2 : envoi direct du mailing

Il est possible d'envoyer le job directement, sans étape de préparation, le job aura un statut SP\_SUBMITTED :

```
$result = $monMailing->envoyerDirectement();  
echo "Le mailing a été envoyé sur la plate-forme Service postal et porte l'ID :", $monMailing->jobID;
```

## 6.3 Suivi du mailing

### 6.3.1 Accès à un mailing transmis

```
$monMailing = $spSession->chargeMailingJob( $jobID );
```

A partir de la référence Service Postal (en général sauveé dans votre base de données), vous pouvez charger un mailing déjà transmis. Si le job existe mais n'est pas un mailing, une erreur sera levée.

### 6.3.2 Statut du mailing

```
$queryStatusResult = $monMailing->queryStatut();  
echo "Le statut de l'envoi est : ", $queryStatusResult->spStatus, "<br />";
```

Vous devez fournir la référence de la lettre, au sein de la page.

Les différents statuts possibles sont :

### 6.3.3 Coût total de l'envoi

```
$queryCostResult = $monMailing->queryCoutTotal();  
echo "Le coût du service est de : ", $queryCostResult, " €<br />";
```

### 6.3.4 Détail du coût de l'envoi

```
$queryRateResult = $monMailing->queryCoutDetail();
```

Exemple d'affichage :

```
echo "<table border='1'><tr><th>code</th><th>prix</th><th>qté</th></tr>";  
foreach($queryRateResult->spServiceCodeList->SP_ServiceCode as $code)  
    echo "<tr><td>", $code->spCode, "</td><td>", $code->spPrice, "</td><td>", $code->spQuantity,  
    "</td></tr>";  
echo "</table>";  
echo "Poids du courrier : ", $queryRateResult->spWeight, " g<br />";  
echo "Prix du service : ", $queryRateResult->spServicePrice, " €<br />";  
echo "Prix de l'affranchissement du service : ", $queryRateResult->spStampPrice, " €<br />";
```

La liste des codes produits est disponible en fin de document.

### 6.3.5 Télécharger les documents de l'envoi

```
$queryDocumentResult = $monMailing->queryDocument(SP\Options\DocumentType::DOCUMENT);  
echo "Votre document peut être <a href='\" , $queryDocumentResult->spURL, "\">visualisé</a><br />";
```

## 6.4 Annulation d'un mailing

```
$cancelResult = $monMailing->annuler();
```

Attention : l'annulation sera refusée si le courrier a déjà été produit.

# 7 Gestion des envois : déroulé pas à pas

Pour suivre ou gérer des envois, il n'est pas nécessaire de créer un objet lettreJob ou mailingJob : ces actions peuvent être réalisées directement à partir de l'objet Session.

## 7.1 Suivi d'un job

### 7.1.1 Statut d'un job

```
$queryStatusResult = $spSession->queryJobStatut();  
echo "Le statut de l'envoi est : ", $queryStatusResult->spStatus, "<br />";
```

Les différents statuts possibles sont détaillés dans la partie

### 7.1.2 Coût total d'un job

```
$queryCostResult = $my_job->queryJobCoutTotal();  
echo "Le coût du service est de : ", $queryCostResult, " €<br />";
```

### 7.1.3 Détail du coût du job

```
$queryRateResult = $my_job->queryJobCoutDetail();
```

Exemple d'affichage :

```
echo "<table border='1'><tr><th>code</th><th>prix</th><th>qté</th></tr>";  
foreach($queryRateResult->spServiceCodeList->SP_ServiceCode as $code)  
    echo "<tr><td>", $code->spCode, "</td><td>", $code->spPrice, "</td><td>", $code->spQuantity,  
    "</td></tr>";  
echo "</table>";  
echo "Poids du courrier : ", $queryRateResult->spWeight, " g<br />";  
echo "Prix du service : ", $queryRateResult->spServicePrice, " €<br />";  
echo "Prix de l'affranchissement du service : ", $queryRateResult->spStampPrice, " €<br />";
```

La liste des codes produits est disponible en fin de document.

### 7.1.4 Télécharger les documents du job

```
$queryDocumentResult = $spSession->queryJobDocument(SP\Options\DocumentType::DOCUMENT);  
echo "Votre document peut être <a href='", $queryDocumentResult->spURL, "'>visualisé</a><br />";
```

## 7.2 Suivi global

Il est possible d'avoir une vision globale de vos envois :

```
$cancelResult = $my_job->annuler();
```

## 7.3 Annulation d'un job

```
$cancelResult = $my_job->annuler();
```

Attention : l'annulation sera refusée si le courrier a déjà été produit.

## 8 Mailing : modèle de document et fichier de données

L'intérêt du mailing est de confier d'un coup l'envoi de plusieurs dizaines, centaines ou milliers de courriers à Service Postal.

Vous pouvez soit créer les documents à l'avance (auquel cas ils devront être fournis au format PDF accompagnés d'un fichier de données qui permettra d'associer chaque document à ses paramètres d'envoi) soit confier à Service Postal la charge de créer les documents (auquel cas vous fournirez un modèle de document au format .docx et un fichiers de données contenant les paramètres de chaque envoi).

Les champs nécessaires dans le fichier de données d'un mailing sont décrits ci-dessous.

Si un champ obligatoire est manquant, l'ensemble du mailing sera refusé.

L'ordre des champs dans le fichier de données n'est pas important, du moment que les intitulés des colonnes sont respectés. Il est possible de créer des champs de données supplémentaires, en ajoutant des colonnes. Le nom des colonnes est sensible à la casse, n'utilisez que des majuscules.

Le modèle de document doit être un fichier .docx ou .doc. Les champs du fichier de données peuvent être insérés à n'importe quel endroit du document, en utilisant la syntaxe suivante : le nom du champ entouré par deux '#' (par exemple : ##NOM\_DESTINATAIRE##).

Champs destinataire	Type	Longueur max	Obligatoire	Description
NOM_DESTINATAIRE	String	32	Oui	Nom de famille
PRENOM_DESTINATAIRE	String	32	Oui	Prénom
NOM_SOCIETE_DESTINATAIRE	String	32	Non	Société
ADRESSE_LIGNE1_DESTINATAIRE	String	32	Oui	Adresse postale
ADRESSE_LIGNE2_DESTINATAIRE	String	32	Non	Complément d'adresse
CODE_POSTAL_DESTINATAIRE	Numeric	5	Oui	Code postal
VILLE_DESTINATAIRE	String	26	Oui	Ville
PAYS_DESTINATAIRE	String	32	Non	Pays (France par défaut)
Champs expéditeur	Type	Longueur max	Obligatoire	Description
NOM_EXPEDITEUR	String	32	Oui si recommandé	Nom de famille
PRENOM_EXPEDITEUR	String	32	Oui si recommandé	Prénom
NOM_SOCIETE_EXPEDITEUR	String	32	Non	Société
ADRESSE_LIGNE1_EXPEDITEUR	String	32	Oui si recommandé	Adresse postale
ADRESSE_LIGNE2_EXPEDITEUR	String	32	Non	Complément d'adresse
CODE_POSTAL_EXPEDITEUR	Numeric	5	Oui si recommandé	Code postal
VILLE_EXPEDITEUR	String	26	Oui si recommandé	Ville
PAYS_EXPEDITEUR	String	32	Non	Pays (France par défaut)
REFERENCE_CLIENT	String	32	Non	La référence de cet envoi pour vous

# 9 Liste des codes produits

Chaque envoi est composé de plusieurs codes produits parmi cette liste :

Description	Code	TVA
<b>Affranchissement</b>		
Timbre Ecopli	TIM-ECO	Non
Timbre Lettre prioritaire	TIM-LP	Non
Timbre Lettre verte	TIM-LV	Non
Timbre Lettre recommandée avec accusé de réception	TIM-LRAR	Non
Timbre Lettre recommandée sans accusé de réception	TIM-LR	Non
<b>Enveloppe</b>		
<b>Enveloppe au format C6/DL (feuilles A4 pliées en trois)</b>	<b>ENV-C6-1F</b>	<b>Oui</b>
Enveloppe au format C5 (feuilles A4 pliées en deux)	ENV-C5-1F	Oui
Enveloppe au format C4 (feuilles A4 non pliées)	ENV-C4-1F	Oui
<b>Porte-Adresse</b>		
<b>Page de porte-adresse</b>	<b>PORT-ADR</b>	<b>Oui</b>
<b>Impression des pages</b>		
Impression première page en noir et blanc	PAGE-1-NB	Oui
Impression première page en couleur	PAGE-1-COUL	Oui
Impression page suivante en noir et blanc recto	PAGE-S-NB	Oui
Impression page suivante en noir et blanc recto/verso	PAGE-S-NB-RV	Oui
Impression page suivante en couleur recto	PAGE-S-COUL	Oui
Impression page suivante en couleur recto/verso	PAGE-S-COUL-RV	Oui
<b>Gestion des recommandés</b>		
Fabrication d'un recommandé	LR-AR	Oui
Gestion du retour des AR	LR-AR-RETOUR	Oui
Option lettre recommandé internationale	LR-AR-INTER	Oui
<b>Stockage</b>		
Option stockage AR 3 ans	LR-AR-STOCK1	Oui
Option stockage AR 6 ans	LR-AR-STOCK6	Oui
Option stockage AR 10 ans	LR-AR-STOCK10	Oui
<b>Gestion des plis non distribués</b>		
Gestion des plis non distribués	PND	Oui

## Exemples :

- Lettre recommandée de 4 pages couleur recto/verso avec porte-adresse :

Quantité	Code	TVA
1	TIM-LRAR	Non
1	PORTE-ADR	Oui
1	PAGE-1-COUL	Oui
3	PAGE-S-COUL	Oui
1	LR-AR	Oui

- Lettre écopli de 10 pages recto/verso noir et blanc sans porte adresse :

Quantité	Code	TVA
1	TIM-ECO	Non
1	PAGE-1-NB	Oui
9	PAGE-S-NB	Oui
1	ENV-C5-1F	Oui

Remarque : Il n'y a pas de prix pour l'enveloppe C6/DL car elle est incluse dans le prix de la 1ere page.

# 10 FAQ – Questions fréquentes

## Authentification

Q : J'obtiens un message « You do not have permission to access API » quand je me connecte

R : L'accès à l'API n'a pas été ouvert pour votre compte. Veuillez contacter notre équipe commerciale.

## Options d'envoi

Q : Si je dépasse le nombre de pages autorisées dans le format d'enveloppe que j'ai choisi, que va-t-il se passer ?

R : Le courrier sera refusé, une exception sera levée : Wrong Type of envelope.

Vous pouvez opter pour le format d'enveloppe ENVELOPPE\_AUTO qui choisira l'enveloppe la plus adaptée dans cet ordre : C6 -> C5 -> C4 (du moins coûteux au plus coûteux).

Q : Si je dépasse le nombre de pages autorisée, que va-t-il se passer ?

R : Le courrier sera refusé, une exception sera levée : Wrong Type of envelope

## Mailings

Q: Est-il possible d'insérer des champs dans le contenu de ma lettre (et pas seulement dans l'adresse) ?

R : Vous pouvez insérer des champs à tout endroit de votre lettre type.

Q: Est-il possible d'utiliser des champs personnalisés ?

R : Vous pouvez créer autant de champs personnalisés que vous le souhaitez, et les insérer dans votre lettre type.

## Passage en production

Q: Je n'ai pas l'adresse des WSDL de production !

R : Les WSDL de production ne sont pas disponibles sur le web, elles vous seront envoyées au moment du passage en production et doivent être stockées sur votre serveur, à une adresse non accessible publiquement.

Q: Mes tests sont OK, comment passer en production ?

R : Le passage en production suppose une étape commerciale : vous devez avoir validé le contrat de souscription.

Vous recevrez alors vos comptes d'accès de production (login / mot de passe) ainsi que les fichiers WSDL à placer sur votre serveur (ces fichiers ne doivent pas être accessibles publiquement).

## Délai de production

Q: Au bout de combien de temps mes courriers seront-ils produits ?

R : Les courriers soumis par l'API sont produits au fil de l'eau, en général avec un délai d'une heure durant lequel vous pouvez les annuler. Lorsque les courriers sont soumis peu de temps avant la clôture, ils peuvent être produits immédiatement, ou repoussés au lendemain.

Q: Comment être sûr que mes courriers seront produits le jour même

R : Notre chaîne de production automatisée optimise la production des courriers, sachant que vos courriers sont « en concurrence » avec les courriers de nos autres clients. En soumettant vos courriers tôt dans la journée, vous augmentez la possibilité que vos courriers soient produits le jour même. A l'inverse, lorsque vous soumettez vos courriers tard, proche de la clôture, moins la probabilité que vos courriers soient produits le jour même.

## Sécurité

Q: Quelle est la sécurité de mon compte ?

R : Tous les échanges sont réalisés sur un protocole crypté (SSL) avec un certificat de sécurité. Votre compte est donc protégé, charge à vous de vous assurer de la protection de vos codes d'accès.

Votre compte peut disposer d'un niveau de sécurité supplémentaire par l'ajout d'une liste d'adresses IP autorisées à accéder à votre compte. Nous vous recommandons de demander la mise en place de cette option auprès de votre chargé de compte.