



**SOLUTION COLLECT**

# **API Webservices REST Marketplace**

## **Guide d'implémentation**

Version du document 1.7

# Sommaire

<b>1. HISTORIQUE DU DOCUMENT.....</b>	<b>4</b>
<b>2. OBTENIR DE L'AIDE.....</b>	<b>6</b>
<b>3. GLOSSAIRE DE LA MARKETPLACE.....</b>	<b>7</b>
<b>4. PRÉSENTATION DE LA MARKETPLACE.....</b>	<b>8</b>
Principe de commission.....	8
Calcul de la commission minimale.....	8
Méthodes d'envoi.....	9
<b>5. PRÉSENTATION DU WEBSERVICE.....</b>	<b>11</b>
5.1. Les grandes étapes du paiement.....	11
5.2. Les Ressources du Webservice.....	11
5.3. Prérequis.....	12
<b>6. COMPRENDRE LE DÉROULEMENT D'UN PAIEMENT MARKETPLACE.....</b>	<b>13</b>
<b>7. S'IDENTIFIER LORS DES ÉCHANGES.....</b>	<b>16</b>
<b>8. COMPRENDRE LES CODES RETOUR DE STATUT HTTP ENVOYÉS PAR LE WEBSERVICE.....</b>	<b>17</b>
<b>9. CONSULTER LES SOUS-MARCHANDS ENREGISTRÉS DANS LA MARKETPLACE.....</b>	<b>18</b>
<b>10. RÉALISER UN PAIEMENT.....</b>	<b>19</b>
10.1. Paiement en N fois.....	23
Paiements réguliers : MULTI.....	23
Echéancier personnalisé : MULTI_EXT.....	24
10.2. Paiement avec délai de capture.....	26
10.3. Paiement avec création d'un alias.....	26
10.4. Paiement par alias.....	28
10.5. Paiement à l'initiative du marchand.....	29
10.6. Paiement en préautorisation (validation manuelle).....	30
10.7. Paiement par Titres-Restaurant Dématérialisés.....	33
<b>11. ANALYSER LE RÉSULTAT DU PAIEMENT.....</b>	<b>36</b>
<b>12. GESTION DES ALIAS.....</b>	<b>37</b>
12.1. Créer un token.....	37
12.2. Analyser le résultat d'une demande de token.....	38
12.3. Modifier un alias.....	39
12.4. Récupérer le détail d'un alias.....	41
<b>13. TRAITER LE RETOUR VERS LA MARKETPLACE.....</b>	<b>42</b>
<b>14. MODIFIER UNE COMMANDE.....</b>	<b>43</b>
14.1. Mise à jour d'une commande réglée en plusieurs fois.....	44
<b>15. ANNULER UNE COMMANDE.....</b>	<b>46</b>
<b>16. REMBOURSER UN PAIEMENT.....</b>	<b>47</b>
16.1. Créer la demande de remboursement.....	47
16.2. Suivre l'évolution de la demande de remboursement.....	49
16.3. Modifier une demande de remboursement.....	50

16.4. Annuler une demande de remboursement.....	51
<b>17. DÉCLENCHEMENT MANUEL DU PAIEMENT DES ITEMS.....</b>	<b>52</b>
<b>18. CONSULTER LES CASHOUTS DE LA MARKETPLACE.....</b>	<b>53</b>
18.1. Liste des cashouts.....	53
18.2. Détails des cashouts.....	54
<b>19. DICTIONNAIRE DE DONNÉES.....</b>	<b>55</b>
19.1. Objet Address.....	55
19.2. Objet Alias.....	55
19.3. Objet Buyer.....	56
19.4. Objet Item.....	56
19.5. Objet Marketplace.....	59
19.6. Objet Order.....	60
19.7. Objet Refund.....	61
19.8. Objet Seller.....	62
19.9. Objet Shipping.....	63
19.10. Objet Token.....	64
19.11. Objet Transaction.....	64

# 1. HISTORIQUE DU DOCUMENT

Version	Auteur	Date	Commentaire
1.7	Lyra Collect	11/08/2020	<ul style="list-style-type: none"><li>• Ajout de la description de l'objet Transaction.</li><li>• Correction de la formule de la commission minimale.</li><li>• Correction de l'URL d'appel dans l'exemple de remboursement.</li></ul>
1.6	Lyra Collect	11/06/2020	Ajout du chapitre <i>Paiement par Titres-Restaurant Dématérialisés</i> .
1.5	Lyra Collect	28/05/2020	<ul style="list-style-type: none"><li>• Ajout des statuts de commande autorisant l'annulation.</li><li>• Mise à jour des statuts de commande autorisant le remboursement.</li><li>• Ajout du calcul de la commission minimale.</li><li>• Ajout du chapitre <i>Paiement à l'initiative du marchand</i>.</li></ul>
1.4	Lyra Collect	06/05/2020	<ul style="list-style-type: none"><li>• Ajout de la création des paiements en N fois.</li><li>• Ajout de la modification d'une commande réglée en N fois.</li><li>• Mise à jour du principe de commission.</li><li>• Mise à jour de la ressource items.</li><li>• Mise à jour du chapitre <i>Réaliser un paiement</i>.</li><li>• Correction de l'URL d'appel POST ORDER dans les exemples.</li></ul>
1.3	Lyra Collect	16/07/2019	Ajout de la ressource <b>Cashouts</b> . Mise à jour des ressources <b>Items</b> , <b>Refunds</b> et <b>Order</b> . Objet <b>Order</b> : ajout de l'attribut <b>form_token</b> . Ajout des actions: <ul style="list-style-type: none"><li>• Lister les cashouts d'une marketplace,</li><li>• Obtenir le détail d'un cashout,</li><li>• Modifier une demande de remboursement,</li><li>• Annuler une demande de remboursement,</li><li>• Débloquer le paiement d'un item préalablement bloqué.</li></ul> Mise à jour des chapitres: <ul style="list-style-type: none"><li>• Suivre l'évolution de la demande de remboursement,</li><li>• Modifier une commande,</li><li>• Présentation du Webservice,</li><li>• Les grandes étapes du paiement,</li><li>• Comprendre le déroulement d'un paiement marketplace,</li><li>• Réaliser un paiement.</li></ul> Ajout des chapitres: <ul style="list-style-type: none"><li>• Déclenchement manuel du paiement des items,</li><li>• Consulter les cashouts de la marketplace,</li><li>• Modifier une demande de remboursement,</li><li>• Annuler une demande de remboursement,</li><li>• Paiement en préautorisation (validation manuelle).</li></ul>
1.2	Lyra Collect	22/01/2019	Ajout de la définition du terme " <b>order</b> ". Ajout de la description du <b>Principe de commission</b> . Ajout des ressources <b>Tokens</b> et <b>Refunds</b> . Ajout du cycle de vie des ressources <b>Tokens</b> et <b>Refunds</b> .

Version	Auteur	Date	Commentaire
			<p>Ajout de l'action <b>Annuler une commande</b>.</p> <p>Modification du chapitre <b>Réaliser un paiement</b>:</p> <ul style="list-style-type: none"> <li>• exemples de code mis à jour,</li> <li>• ajout de l'utilisation de la ressource Tokens.</li> </ul> <p>Ajout des chapitres:</p> <ul style="list-style-type: none"> <li>• Analyser le résultat d'une demande de token.</li> <li>• Récupérer le détail d'un token</li> <li>• Modifier un order</li> <li>• Paiement avec délai de capture</li> <li>• Objet Token</li> <li>• Objet Alias</li> <li>• Objet Refund</li> </ul>
1.1	Lyra Collect	12/07/2018	<p>Précisions sur les chapitres : ajout de noms dans les objets</p> <p>Suppression de la ressource transferts</p> <p>Ajout de chapitres :</p> <ul style="list-style-type: none"> <li>• Consulter les sous-marchands enregistrés dans la marketplace</li> <li>• Rembourser un paiement</li> </ul>
1.0	Lyra Collect	12/03/2015	Version initiale

Ce document et son contenu sont strictement confidentiels. Il n'est pas contractuel. Toute reproduction et/ou distribution de tout ou partie de ce document ou de son contenu à une entité tierce sont strictement interdites ou sujettes à une autorisation écrite préalable de Lyra Collect. Tous droits réservés.

## 2. OBTENIR DE L'AIDE

---

Vous cherchez de l'aide? Consultez notre FAQ sur notre site

<https://lyra.com/doc/fr/collect/faq/sitemap.html>

Pour toute question technique ou demande d'assistance, nos services sont disponibles du lundi au vendredi de 9h à 18h

par téléphone au :

**0811900475**

Service 0,06 € / min  
+ prix appel

par e-mail :

[support-ecommerce@lyra-collect.com](mailto:support-ecommerce@lyra-collect.com)

et via votre Back Office Expert, menu **Aide** > **Contacteur le support**

Pour faciliter le traitement de vos demandes, il vous sera demandé de communiquer votre identifiant de boutique (numéro à 8 chiffres) .

### 3. GLOSSAIRE DE LA MARKETPLACE

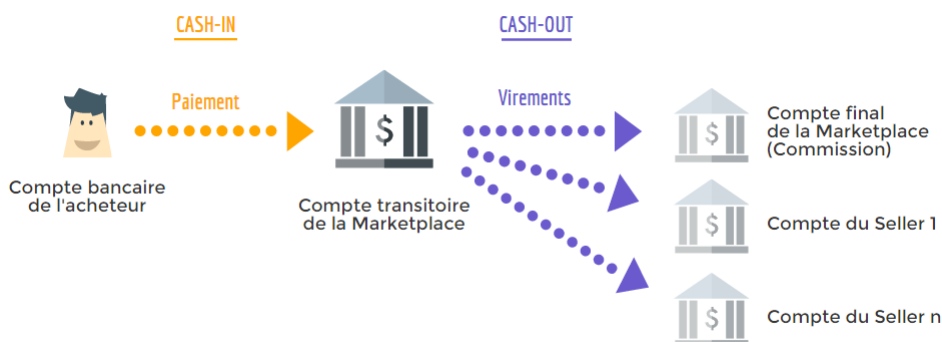
---

Vocabulaire	Description
Cash-in	Paiement de l'acheteur à la Marketplace (CB, VISA...).
Cash-out	Virement de la Marketplace vers les Sellers.
Marketplace	Place de marché (site Internet proposant des articles de divers vendeurs/prestataires)
Order	Un order est composé d'items, il correspond à une commande de la Marketplace. Chaque item est rattaché à un seller.
Seller	Vendeur sur la Marketplace, aussi appelé prestataire. Peut être une société (titulaire d'un SIRET) ou un particulier.

## 4. PRÉSENTATION DE LA MARKETPLACE

La plateforme de paiement propose des services de paiement pour des Marketplaces (places de marché), qui vendent des produits ou des services sur Internet pour le compte de vendeurs avec lesquels elles ont signé un accord commercial en ce sens.

Principe des flux bancaires :



Vocabulaire	Description
Cash-in	Paieement de l'acheteur à la Marketplace (CB, VISA...).
Cash-out	Virement de la Marketplace vers les Sellers.

Les paieements des acheteurs sont ventilés vers les comptes de paieements des sellers. La plateforme de paieement déclenche ensuite des virements vers les comptes bancaires des Sellers de la Marketplace.

### Principe de commission

La commission représente la part du montant de la commande qui revient à la Marketplace.

**Remarque :**

C'est sur cette commission qu'est déduite la part affectée à Lyra Collect. Par conséquent, la commission de la Marketplace doit permettre de couvrir les frais de Lyra Collect. Pour cette raison, le montant global des commissions comprises dans la commande est contrôlé avant le paieement de la commande.

### Calcul de la commission minimale

Pour éviter de soulever une erreur lors contrôle de la commission, vous pouvez préalablement vérifier que le montant de la commission envoyé à l'API marketplace est supérieur ou égal à la commission minimale, dont la formule est donnée ci-dessous :

Symbole	Définition	Exemple 1	Exemple 2
$Com_{min}$	Commission minimale		
$a$	<b>Coefficient</b> de la commission au prorata	2%	1%
$b$	Commission fixe <b>par transaction</b>	50	20
$M$	Montant <b>hors commission</b> de la commande	10000	10000



Symbole	Définition	Exemple 1	Exemple 2
$n$	Nombre de transactions de la commande	1	3
$tva$	Taux de TVA	20%	20%

Remarque :

- $a$ ,  $b$  et  $tva$  sont paramétrés au niveau de l'API marketplace. Si vous ne les connaissez pas, renseignez-vous auprès de votre contact commercial.
- De son côté, la marketplace qui transmet l'order à l'API marketplace maîtrise  $M$  et  $n$ .

La commission minimale s'obtient par la formule suivante :

$$Com_{min}(M) = \frac{aM + bn}{\frac{1}{1 + tva} - a}$$

Avec les montants donnés en exemple, on a donc une commission minimale à :

$$Com_{min}(M) = \frac{10000 \times 0.02 + 50}{\frac{1}{1 + 0.2} - 0.02} = 307.37$$

$$Com_{min}(M) = \frac{10000 \times 0.01 + 3 \times 20}{\frac{1}{1 + 0.2} - 0.01} = 194.33$$

L'unité étant le centime, la commission minimale totale de la commande devra donc être au moins égale à 308 dans l'exemple 1, et de 195 dans l'exemple 2.

## Méthodes d'envoi

Deux méthodes permettent de renseigner le montant de la commission :

- 1. la commission est définie au niveau de la commande** : le montant de la commission est indiqué à travers un item de type commission, marqué par l'attribut **is\_commission=true**. Le seller de cet item doit être le seller Marketplace, c'est-à-dire celui dont l'attribut **is\_marketplace\_seller** vaut **"true"**. Dans ce cas, le montant de la commission s'ajoute au montant des autres items.

Exemple :

```
"items": [{
  "seller": "4d20a9d4-0526-4474-b452-e936dc25418d",
  "reference": "produit_sous_marchand",
  "description": "Produit",
  "amount": 10000,
  "quantity": 1
}, {
  "seller": "72ccc2ff-b455-4653-847e-deb6fee99f8d",
  "reference": "marketplace_commission",
  "description": "Commission",
  "amount": 1000,
  "quantity": 1,
  "is_commission": true
}]
```

Dans cet exemple, si la devise de la commande est EUR et qu'elle contient 2 items, dont 1 item de 100€, plus 1 item de 10€ de commission, alors le montant total de la commande est de 100 + 10 = 110€.

Le seller recevra 100€ et la marketplace recevra 10€ (moins la part affectée à Lyra Collect).

- 2. la commission est définie au niveau d'un item** : le montant de la commission est indiqué en valorisant l'attribut **commission\_amount** de l'item. Dans ce cas, le montant de la commission vient en déduction du montant associé à l'item.

Exemple :

```
{
  "seller": "72ccc2ff-b455-4653-847e-deb6fee99f8d",
  "reference": "abcdef",
  "description": "Restaurant",
  "amount": 10000,
  "quantity": 1,
  "commission_amount": 1000,
  "is_commission": false
}
```

Dans cet exemple, si la devise de la commande est EUR, l'item vaut 100€ et l'attribut **commission\_amount** est renseigné à 10€, la Marketplace percevra 10€, et le marchand de l'item recevra  $100 - 10 = 90€$ .

Cette modalité est utile pour maîtriser la répartition des commissions entre les différents items (et donc entre les différents sellers).

**Notes :**

- Il est possible de combiner les deux modes, c'est-à-dire de définir un **commission\_amount** dans un ou plusieurs items, et d'ajouter un item de type commission. Dans ce cas, les commissions définies au niveau de la commande s'ajoutent aux commissions définies au niveau des items.
- En revanche, il ne fait pas sens – et il n'est pas possible – de définir un **commission\_amount** sur un item de type commission.
- La valeur par défaut de l'attribut **is\_commission** est **"false"**. Il n'est donc pas nécessaire de l'indiquer pour les items non-commission.

Un exemple de création d'ordre est donné au chapitre Réaliser un paiement à la page 19.

## 5. PRÉSENTATION DU WEBSERVICE

Ce document présente le Webservice Marketplace qui permet de :

- Créer une commande
- Faire payer l'acheteur sur la page de paiement Lyra Collect ou *via* le formulaire embarqué;
- Consulter les informations d'une commande

Le Webservice Marketplace a été développé suivant le protocole REST.

### 5.1. Les grandes étapes du paiement

1. Si le site marchand a souscrit et installé le formulaire embarqué, l'acheteur saisit directement ses informations de paiement depuis le formulaire. Sinon, le **site marchand** redirige l'**acheteur** vers la page de paiement Lyra Collect.

La page de paiement affiche le montant total à régler.

2. Lyra Collect fait payer l'**acheteur** le montant total.

3. Si le paiement réussi, Lyra Collect crée les virements vers le(s) **Prestataire(s)** présent(s) dans le panier.

4. Lyra Collect effectue la remise en banque de la transaction effectuée par carte bancaire et envoie le fichier de virement à la **Banque**.

### 5.2. Les Ressources du Webservice

Le Webservice Marketplace est disponible à l'adresse :

- **Mode Test** (phase d'intégration): <https://secure.lyra.com/marketplace-test/>
- **Production**: <https://secure.lyra.com/marketplace/>

Le Webservice propose des ressources accessibles par des méthodes HTTP différentes:

Ressources	Action	Méthode HTTP	URI
<b>Marketplaces</b>	Récupérer la liste des Marketplaces	GET	/marketplaces/
	Récupérer les informations de la Marketplace	GET	/marketplaces/{id_marketplace}
<b>Sellers</b>	Récupérer un prestataire	GET	/sellers/{id_seller}
	Récupérer la liste des prestataires	GET	/marketplaces/{id_marketplace}/sellers
<b>Orders</b>	Créer une commande	POST	/orders?expand=items
	Modifier une commande	PUT	/orders/{id_order}
	Récupérer une commande	GET	/orders/{id_order}
	Récupérer la liste des commandes de la Marketplace	GET	/orders
	Finaliser une commande et préparer la page de paiement	GET	/orders/{id_order}/execute
	Finaliser une commande et récupérer le token du formulaire embarqué	GET	/orders/{id_order}/embedded-execute
	Annuler une commande	DELETE	/orders/{id_order}
<b>Items</b>	Récupérer un élément d'une commande	GET	/items/{id_item}
	Récupérer les éléments d'une commande	GET	/orders/{id_order}/items

Ressources	Action	Méthode HTTP	URI
	Récupérer les éléments d'un prestataire	GET	/sellers/{id_seller}/items
	Créer un élément (article) dans une commande	POST	/orders/{id_order}/items
	Débloquer le paiement d'un item préalablement bloqué	POST	/items/{id_item}/activate
<b>Refunds</b>	Créer un remboursement	POST	/refunds
	Récupérer les éléments d'un remboursement	GET	/refunds/{id_refund}
	Récupérer la liste des remboursements d'une commande	GET	/{id_order}/refunds
	Modifier un remboursement	PUT	/refunds/{id_refund}/
	Supprimer un remboursement	DELETE	/refunds/{id_refund}/
<b>Tokens</b>	Demande de création / modification de token	POST	/tokens
	Récupérer une demande de token simple	GET	/tokens/{id_token}
	Récupérer une demande de token liée à un order	GET	/tokens/{id_order}
<b>Alias</b>	Récupérer les éléments d'un alias	GET	/marketplaces/ {id_marketplace}/alias/ {id_alias}
<b>Cashouts</b>	Lister les cashouts vers les sellers d'une marketplace	GET	/cashouts
	Obtenir le détail d'un cashout	GET	/cashouts/{id_cashout}

### 5.3. Prérequis

---

Contactez Lyra Collect pour activer l'accès à la Marketplace et obtenir l'environnement.

#### Pré requis pour la Marketplace

- Souscrire à l'offre Marketplace.  
En souscrivant à cette offre, la plateforme de paiement vous envoie en retour des données indispensables pour accéder à la Marketplace :
  - l'**ID** de votre Marketplace
  - un **login** et un **mot de passe** indispensables pour vous authentifier.

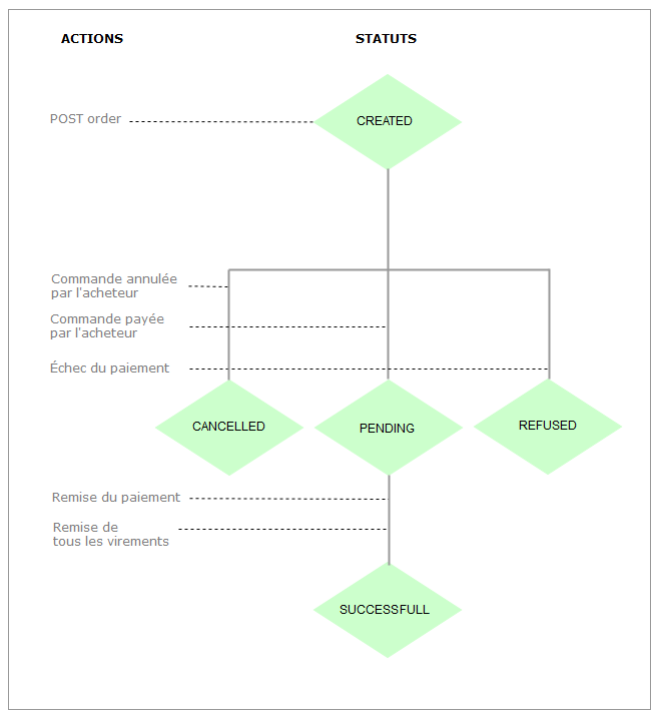
## 6. COMPRENDRE LE DÉROULEMENT D'UN PAIEMENT MARKETPLACE

Ci-dessous la cinématique du déroulement d'un paiement sur la Marketplace :

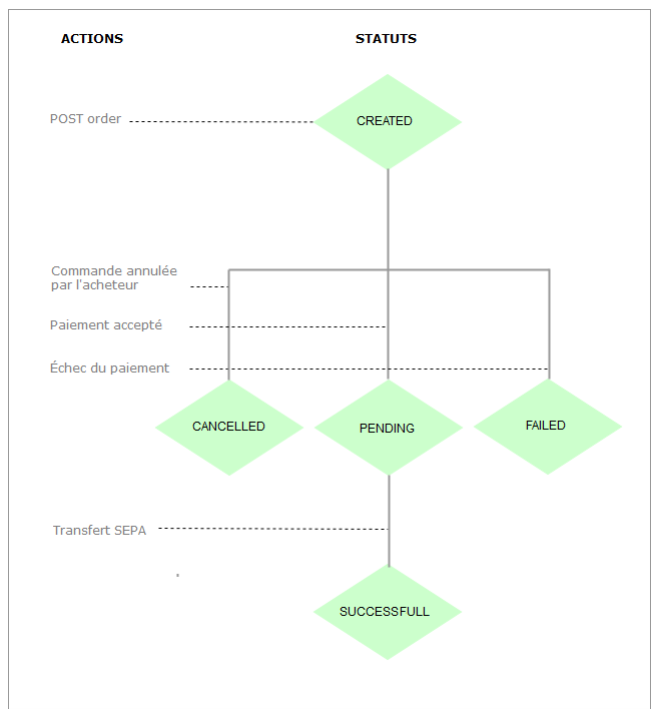
Étapes	Acteurs	Actions
1	Acheteur	Valide son panier sur le site de la Marketplace.
2	Marketplace	Créé un Order via l'API REST (POST ORDER) contenant des items.
3	Marketplace	Figé l'Order via l'API REST (GET ORDER). Finalise l'Order et prépare la page de paiement ou le token du formulaire embarqué.
4	Plateforme de paiement	Créé le contexte de paiement de l'Order. Si le formulaire embarqué est activé, retourne à la marketplace un token (form-token) à insérer dans le formulaire embarqué. Sinon, retourne à la Marketplace une URL pour rediriger l'acheteur vers la page de paiement.
5	Marketplace	Si le formulaire embarqué est activé, construit le formulaire sur la page du site marchand. Sinon, redirige l'acheteur vers l'URL donnée par la plateforme de paiement récupérée par le GET ORDER.
6	Acheteur	Saisit ses informations bancaires (+ authentification 3D Secure sur les pages de paiement).
7	Plateforme de paiement	Valide et enregistre la transaction.
8	Plateforme de paiement	Met à jour le statut des Items et de l'Order.
9	Plateforme de paiement	Envoie une notification à la Marketplace (et éventuellement au formulaire embarqué) indiquant que le statut de l'Order a changé.
10	Marketplace	Appelle la plateforme de paiement pour connaître le statut de l'Order et met à jour son système (GET ORDER).
11	Plateforme de paiement	Redirige l'acheteur vers la Marketplace (URL de retour donnée par la Marketplace = site marchand).
12	Plateforme de paiement	Effectue la remise en banque de la transaction par carte bancaire (ou autre). Créé les virements sur la boutique du Seller. Envoie le fichier d'ordres de virement à la banque.
13	Plateforme de paiement	Notifie la Marketplace du changement de statut de l'Order et des Items.
14	Marketplace	Appelle la plateforme de paiement pour connaître le statut de l'Order et met à jour son système (GET ORDER).
15	Marketplace	Valide manuellement le fichier d'ordres de virement sur l'interface de sa banque.
16	Banque	Exécute les virements demandés.

Le statut des ressources varie tout au long du déroulement d'un paiement.

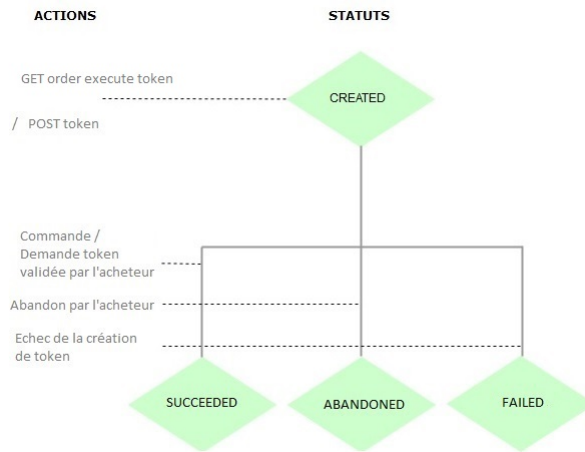
- Cycle de vie de la ressource **Order**



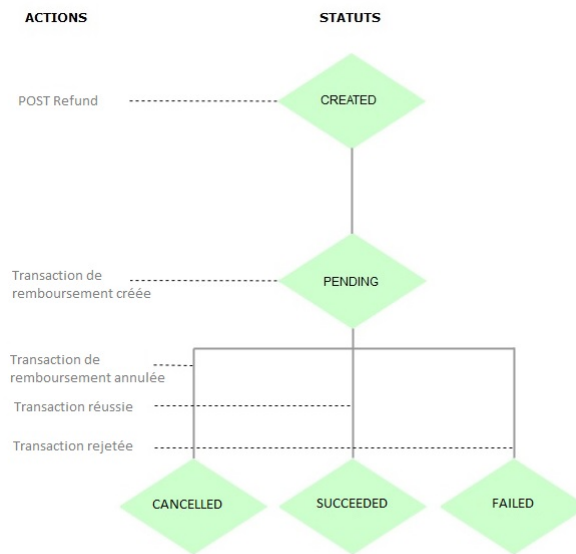
- Cycle de vie de la ressource **Item**



- Cycle de vie de la ressource Tokens



- Cycle de vie de la ressource Refunds



## 7. S'IDENTIFIER LORS DES ÉCHANGES

L'identification s'effectue au moyen d'un en-tête HTTP.

La méthode utilisée est **HTTP Basic Authentication**.

Dans chaque requête HTTP, l'en-tête doit contenir les informations permettant à la Marketplace de s'authentifier auprès du Webservice Marketplace.

Description des en-têtes HTTP :

En-têtes	Description	Valeurs à utiliser
<b>Accept</b>	Détermine le format du contenu qui sera retourné par le serveur. Architecture REST qui permet de faire des échanges au format json	'Accept: application/json'
<b>Authorization</b> <b>Description</b>	: Jeton d'identification selon le principe de "Basic Authentication over HTTPS". Se compose de l'identifiant et du mot de passe de l'utilisateur de l'API séparés par le caractère ":", le tout encodé en base 64.	'Authorization: Basic YWRtaW46YWRtaW4='
<b>Content-type</b>	Détermine le format du contenu qui est envoyé au serveur.	'content-type: application/json'
<b>Method</b>	Voir tableau au chapitre <b>Les Ressources du Webservice</b> pour connaître la méthode à utiliser en fonction de la ressource.	<b>GET   POST   PUT   DELETE</b>

Les étapes pour construire ces en-têtes sont les suivantes :

1. Utilisez la méthode **Basic Authentication**.
2. Spécifiez dans l'en-tête **Authorization** la méthode utilisée : **Basic** suivie de la représentation en Base 64 du login et du mot de passe séparés par le caractère ":".
3. Encodez le résultat obtenu en base 64.
4. Ajoutez la chaîne en "Basic".

### Remarque :

*Ne pas oublier de mettre un espace après **Basic**.*

- Exemple **cURL** :

```
$ curl 'https://secure.lyra.com/marketplace-test/123456/orders/' -H
'Authorization: Basic YWRtaW46YWRtaW4=' -H
'Content-Type: application/json' -H 'Accept: application/json' --data
'{}'
-i
```

- Exemple de requête complète en **Python** :

```
r = requests.post(
    <target url>,
    data=<json data>,
    auth=(<api login>, <api password>),
    headers={'content-type': 'application/json'},
    verify=False
)
```

- Exemple de requête en **.NET**

```
var myURL = "https://secure.lyra.com/marketplace-test/orders?expand=items"
HttpRequest myHttpRequest = (HttpRequest)WebRequest.Create(myURL);
myHttpRequest.ContentType = "application/json";
myHttpRequest.Accept = "application/json";
myHttpRequest.Method = "post";
string authInfo = userName + ":" + userPassword;
authInfo = Convert.ToBase64String(Encoding.Default.GetBytes(authInfo));
myHttpRequest.Headers["Authorization"] = "Basic " + authInfo;
```



## 8. COMPRENDRE LES CODES RETOUR DE STATUT HTTP ENVOYÉS PAR LE WEBSERVICE

---

Code erreur	Description
200 - 20X	Requête traitée avec succès
401	Accès non autorisé (problème dans le header Authorization)
403	Accès interdit (L'utilisateur de l'API n'a pas la permission d'effectuer cette requête)
400	Mauvaises données d'entrée (exemple : le format de certains paramètres n'est pas respecté)
404	L'objet demandé est introuvable (On a demandé un uuid qui n'existe pas)
500	Erreur interne du serveur

Ci-dessous quelques pistes d'analyse pour vous aider à résoudre rapidement les problèmes survenus :

### Erreur 400 :

- **"?expand=items"** est absent à la fin de la requête POST order
- les accents ne sont pas encodés en ANSI
- la référence d'un item comporte des espaces.

## 9. CONSULTER LES SOUS-MARCHANDS ENREGISTRÉS DANS LA MARKETPLACE

Pour consulter les sous-marchands enregistrés dans la marketplace, vous devez lancer l'appel en mode GET.

```
GET https://secure.lyra.com/marketplace/marketplaces/<marketplace_uid>/sellers
```

Remplacez <marketplace\_uid> par l'uuid fourni par Lyra Collect.

### Exemple d'appel :

```
GET https://secure.lyra.com/marketplace/marketplaces/30805a03-11ec-4447-93a5-243f39c89009/sellers
```

### Exemple de réponse à l'appel GET :

```
{
  "count": 1,
  "next": null,
  "previous": null,
  "results": [
    {
      "uuid": "eb94407d-c433-40ba-a5d3-d1baa0f7ed0a",
      "href": "https://secure.lyra.com/marketplace/sellers/eb94407d-c433-40ba-a5d3-d1baa0f7ed0a",
      "created_at": "2018-04-13T12:57:59.943702Z",
      "updated_at": "2018-04-13T12:57:59.943713Z",
      "marketplace": "30805a03-11ec-4447-93a5-243f39c89009",
      "reference": "MKPINTEG_Seller",
      "description": "",
      "bic": "",
      "iban": "",
      "status": "ACTIVE",
      "links": {
        "items": {
          "href": "https://secure.lyra.com/marketplace/sellers/eb94407d-c433-40ba-a5d3-d1baa0f7ed0a/items"
        },
        "transfers": {
          "href": "https://secure.lyra.com/marketplace/sellers/eb94407d-c433-40ba-a5d3-d1baa0f7ed0a/transfers"
        }
      }
    }
  ]
}
```

## 10. RÉALISER UN PAIEMENT

Pour réaliser un paiement, vous devez suivre les étapes suivantes :

1. Créer une commande en faisant un POST sur la ressource ORDER.
2. Modifier éventuellement la commande pour mettre à jour le panier ou les montants des commissions par exemple.
3. Figurer la commande pour effectuer le paiement en faisant un GET :
  - soit sur l'url de l'attribut **execute** fourni dans les links de l'ORDER, dans ce cas la réponse contient l'url de redirection (attribut **payment\_url**),
  - soit sur l'url de l'attribut **execute-embedded** pour initialiser le paiement par formulaire embarqué, et dans ce cas la réponse contient un formToken (attribut **form\_token**).
4. Rediriger l'acheteur vers l'URL de la page de paiement ou afficher le formulaire embarqué en utilisant le form\_token.

### Étape 1

Pour déclarer des items dans votre commande, vous **devez** mentionner l'argument **?expand=items** dans l'url (/marketplace/orders?expand=items).

Dans le cas contraire, vos items ne seront pas pris en compte (la liste d'items retournée par le serveur est vide) et la commande sera créée sans items. Lors de l'exécution de la commande (étape 3) vous rencontrerez une erreur 400 {"amount":"The sum of item amounts cannot be zero or negative."}.

### Exemple d'appel POST ORDER :

```
POST https://secure.lyra.com/marketplace/orders?expand=items
```

```
{
  "marketplace": "ebfb36ab-2d30-4326-adb9-e16b0c9a89f3",
  "reference": "Exemple MKP BURO",
  "description": "Commande fournitures",
  "currency": "EUR",
  "url_return": "https://URLreturn.com",
  "items": [
    {
      "seller": "d0f80202-0676-4d8d-9247-f455f30aec1b",
      "reference": "commburo",
      "description": "commission BURO",
      "amount": 150,
      "is_commission": true
    },
    {
      "seller": "5d0ef88c-3345-4b33-948e-80e23d553b73",
      "reference": "STB1",
      "description": "Stylos bleus",
      "amount": 1000,
      "is_commission": false
    }
  ],
  "buyer": {
    "type": "PRIVATE",
    "email": "jean.dupont@acheteur.fr",
    "phone_number": "33 (0)1 11 22 33 44",
    "reference": "ZhIsYM"
  },
  "shipping": {
    "shipping_method": "RELAY_POINT"
  }
}
```

## Exemple de retour du POST ORDER :

```
{
  "uuid": "1a2b51f8-9d62-408f-ab39-aa8e28ab15f0",
  "href": "https://secure.lyra.com/marketplace/orders/1a2b51f8-9d62-408f-ab39-aa8e28ab15f0",
  "created_at": "2019-01-21T16:46:46.517000Z",
  "updated_at": "2019-01-21T16:46:46.517000Z",
  "marketplace": "ebfb36ab-2d30-4326-adb9-e16b0c9a89f3",
  "reference": "Exemple MKP BURO",
  "description": "Commande fournitures",
  "alias": null,
  "buyer": [
    {
      "reference": "ZhIsYM",
      "title": null,
      "type": "PRIVATE",
      "first_name": null,
      "last_name": null,
      "legal_name": null,
      "phone_number": "33 (0)1 11 22 33 44",
      "email": "jean.dupont@acheteur.fr",
      "address": null
    }
  ],
  "shipping": [
    {
      "shipping_method": "RELAY_POINT",
      "delivery_company_name": null,
      "shipping_speed": null,
      "shipping_delay": null,
      "type": null,
      "first_name": null,
      "last_name": null,
      "legal_name": null,
      "phone_number": null,
      "address": null
    }
  ],
  "amount": null,
  "initial_amount": null,
  "currency": "EUR",
  "status": "CREATED",
  "webhook_result": null,
  "url_return": "https://URLreturn.com",
  "items": [
    {
      "uuid": "d33balab-793e-4d3d-943f-72b14a1e885b",
      "href": "https://secure.lyra.com/marketplace/items/d33balab-793e-4d3d-943f-72b14a1e885b",
      "created_at": "2019-01-21T16:46:46.510000Z",
      "updated_at": "2019-01-21T16:46:46.573000Z",
      "seller": "5d0ef88c-3345-4b33-948e-80e23d553b73",
      "order": "1a2b51f8-9d62-408f-ab39-aa8e28ab15f0",
      "reference": "STB1",
      "description": "Stylos bleus",
      "type": "ENTERTAINMENT",
      "amount": 1000,
      "quantity": 1,
      "transfer": null,
      "status": "CREATED",
      "links": null,
      "is_commission": false
    },
    {
      "uuid": "20034374-b425-4beb-91a8-a660ca5a3f9b",
      "href": "https://secure.lyra.com/marketplace/items/20034374-b425-4beb-91a8-a660ca5a3f9b",
      "created_at": "2019-01-21T16:46:46.502000Z",
      "updated_at": "2019-01-21T16:46:46.551000Z",
      "seller": "d0f80202-0676-4d8d-9247-f455f30aec1b",
      "order": "1a2b51f8-9d62-408f-ab39-aa8e28ab15f0",
      "reference": "commburo",
      "description": "Commission BURO",
      "type": "ENTERTAINMENT",
      "amount": 150,
      "quantity": 1,
      "transfer": null,
      "status": "CREATED",
      "links": null,
      "is_commission": true
    }
  ],
  "links": {
    "items": {
      "href": "https://secure.lyra.com/marketplace/orders/1a2b51f8-9d62-408f-ab39-aa8e28ab15f0/items"
    }
  },
  "refunds": {
```

```

    "href": "https://secure.lyra.com/marketplace/orders/1a2b51f8-9d62-408f-ab39-aa8e28ab15f0/
refunds"
  },
  "execute": {
    "href": "https://secure.lyra.com/marketplace/orders/1a2b51f8-9d62-408f-ab39-aa8e28ab15f0/
execute"
  },
  "execute-embedded": {
    "href": "https://secure.lyra.com/marketplace/orders/1a2b51f8-9d62-408f-ab39-aa8e28ab15f0/
execute-embedded"
  }
},
"vads_transaction_id": 0,
"vads_transaction_date": null,
"expected_capture_date": null,
"capture_delay": null
}

```

## Étape 2

Une commande peut être modifiée par une requête **PUT ORDER** (/marketplace/orders/<order\_uuid>?expand=items) à l'aide de l'**order\_uuid** retourné à l'étape 1, et ce, plusieurs fois.

Les attributs **buyer** et **shipping** doivent obligatoirement être renseignés.

La liste des items est renouvelée à chaque PUT. Si vous souhaitez ajouter un item supplémentaire, vous devez le transmettre avec l'ensemble des items déjà créés. Sinon la liste ne contiendra que cet item.

A contrario pour supprimer un item il suffit de renvoyer la liste complète des items sans celui que vous souhaitez supprimer.

Exemple d'appel PUT ORDER pour ajouter un item dans le panier:

```

PUT https://secure.lyra.com/marketplace/orders/515abac9-6cb2-4e21-8a25-b08d7e41e43c?
expand=items

```

```

{
  "marketplace": "ebfb36ab-2d30-4326-adb9-e16b0c9a89f3",
  "reference": "Exemple MKP BURO",
  "description": "Commande fournitures",
  "currency": "EUR",
  "url_return": "https://URLreturn.com",
  "items": [
    {
      "seller": "d0f80202-0676-4d8d-9247-f455f30aec1b",
      "reference": "commburo",
      "description": "commission BURO",
      "amount": 270,
      "is_commission": true
    },
    {
      "seller": "5d0ef88c-3345-4b33-948e-80e23d553b73",
      "reference": "STB1",
      "description": "Stylos bleus",
      "amount": 1000,
      "is_commission": false
    },
    {
      "seller": "5d0ef88c-3345-4b33-948e-80e23d553b73",
      "reference": "STR3",
      "description": "Stylos rouges",
      "amount": 1000,
      "is_commission": false
    }
  ],
  "buyer": {
    "type": "PRIVATE",
    "email": "jean.dupont@acheteur.fr",
    "phone_number": "33 (0)1 11 22 33 44",
    "reference": "ZhIsYM"
  },
  "shipping": {
    "shipping_method": "RELAY_POINT"
  }
}

```

### Etape 3

A l'exception des paiements en validation manuelle (voir chapitre Paiement en préautorisation (validation manuelle) à la page 30), vous devez vous assurer *avant* d'effectuer la requête d'exécution et de figer la commande que le montant des commissions est supérieur au minimum requis. Si ce n'est pas le cas, la requête d'exécution renverra une erreur de validation 400 avec le message {"commission": "Insufficient amount of commission."}

Vous trouverez des informations sur la commission minimum dans la section Principe de commission du chapitre Présentation de la Marketplace à la page 8.

#### Exemple d'appel GET via la redirection pour figer la commande :

```
GET
https://secure.lyra.com/marketplace/orders/515abac9-6cb2-4e21-8a25-b08d7e41e43c/execute
```

#### Exemple de réponse en GET :

```
{
  'payment_url': 'https://secure.lyra.com/vads-payment/
exec.refresh.a;jsessionid=HgKAcpcgZQA29qCZDjXw4kS.marketplacevad01?
cacheId=450272311503195000050'
}
```

#### Exemple d'appel GET via la le formulaire embarqué pour figer la commande :

```
GET https://secure.lyra.com/marketplace/orders/515abac9-6cb2-4e21-8a25b08d7e41e43c/execute-
embedded
```

#### Exemple de réponse en GET :

```
{"form_token": "fa0yfV2FQuR3aak1SwsgcuZg195eyJhbW91bnQiOjEwOTAwLCJjdXJyZW5jeSI6IkVVUiIsIm
1vzGUiOiJURVNUiwiidmVyc2lvbiiI6Mywib3JkZXJJZCI6IjVjNDk1N2E5LWNkOGEtNGJhZC1hMGNiLTgxMT
ZiMWE3ZDdmNiIsInNob3BOYWllIjoiTHlyYSBTTVMiLCJicmFuZFByaW9yaXR5IjpbIkJBTkNPTlRBQ1QiL
CJDQiIsIkUtQ0FSVEVCTEVVRSIsIk1BU1RFUkNBUCkQilCVGQVVMVCJ9fX0a702"}
```

Note : le form\_token peut peser jusqu'à 8ko

### Etape 4

Dans le cas d'un paiement par redirection, il suffit de rediriger l'acheteur vers le lien fourni. La même adresse peut être utilisée pour un affichage par iframe.

Dans le cas d'un formulaire de paiement embarqué, le **form\_token** doit être inséré dans le formulaire (attribut **kr-form-token**).

Exemple :

```
<body>
<!-- payment form -->
<div class="kr-embedded"
  kr-form-token="fa0yfV2FQuR3aak1SwsgcuZg195eyJhbW91bnQiOjEwOTAwLCJjdXJyZW5jeSI6IkVVUiIsIm
1vzGUiOiJURVNUiwiidmVyc2lvbiiI6Mywib3JkZXJJZCI6IjVjNDk1N2E5LWNkOGEtNGJhZC1hMGNiLTgxMT
ZiMWE3ZDdmNiIsInNob3BOYWllIjoiTHlyYSBTTVMiLCJicmFuZFByaW9yaXR5IjpbIkJBTkNPTlRBQ1QiL
CJDQiIsIkUtQ0FSVEVCTEVVRSIsIk1BU1RFUkNBUCkQilCVGQVVMVCJ9fX0a702">
<!-- payment form fields -->
<div class="kr-pan"></div>
<div class="kr-expiry"></div>
<div class="kr-security-code"></div>

<!-- payment form submit button -->
<button class="kr-payment-button"></button>
<!-- error zone -->
<div class="kr-form-error"></div>
</div>
</body>
```

## 10.1. Paiement en N fois

---

Dans le cas où l'acheteur souhaite étaler ses paiements dans le temps, il peut effectuer un paiement en plusieurs fois.

Lors d'un paiement en N fois, l'ensemble des transactions sont créées le jour du paiement.

Seule la première échéance peut faire l'objet d'une garantie pour le marchand à condition que sa date de présentation demandée soit antérieure à la date de validité de l'autorisation (en fonction du moyen de paiement).

Au moment du paiement, un contrôle est effectué pour vérifier que le moyen de paiement utilisé est valide pendant toute la durée de l'échéancier.

**Ce mode de paiement n'est pas supporté par le formulaire embarqué.**

La liste des transactions est accessible à l'adresse :

```
GET https://secure.lyra.com/marketplace/orders/{order_uuid}/transactions
```

Elle apparaît aussi dans la liste des liens ('links') en réponse à un GET ORDER, POST ORDER, ou PUT ORDER.

Notez cependant que les transactions sont générées après le paiement de la commande. Si l'adresse est consultée avant, la liste des transactions est vide.

Pour initialiser un paiement en N fois, il suffit d'ajouter l'attribut **payment\_config** dans la ressource POST ORDER, au moment de la création de la commande.

Exemple d'utilisation de l'attribut **payment\_config** :

```
{
  "marketplace": "9537e049-8862-400a-ae8d-da2ec9ca6051",
  "payment_config": "MULTI:first=30000;count=3;period=30"
  "reference": "cmd00053",
  "description": "Commande",
  "currency": "EUR",
  "url_return": "http://www.my-website.com",
  "language": "fr"
}
```

Notes :

- Par défaut, le paiement est mono-transactionnel. Si l'attribut **payment\_config** n'est pas renseigné au moment de la création de la commande, il prendra la valeur **SINGLE** ;
- Une fois la commande figée, il n'est pas possible de modifier la valeur de **payment\_config**. Toutefois, il reste possible de modifier certains paramètres des transactions en attente (cf. infra) ;
- Le paiement en N fois n'est actuellement pas compatible avec le paiement en cascade (et donc à fortiori pas non plus compatible avec le système Ticket Restaurant Dématérialisé).

Il est possible de définir deux types de paiements échelonnés différents, selon que la marketplace souhaite créer automatiquement un échéancier régulier, ou bien définir les dates et les montants à son gré.

### **Paiements réguliers : MULTI**

Dans le cas d'un échéancier régulier, le premier paiement intervient à la date de la création de la commande, et son montant doit être explicitement indiqué. Les montants et les dates des paiements suivants sont calculés en fonction de la différence restant à payer, le nombre et la fréquence des échéances.

**Syntaxe** : la valeur de **payment\_config** doit être préfixée de la chaîne « **MULTI:** » suivie dans l'ordre par les paires clé=valeur suivantes séparées par des « ; » :

- **first=** : montant du premier paiement (exprimé dans la plus petite unité de la devise).
- **count=** : nombre total d'échéances.
- **period=** : intervalle en nombre de jours entre 2 paiements.

Exemple :

Le montant du dernier versement est calculé en fonction du solde restant à payer.

Si un arrondi est nécessaire, il est pris en charge à ce moment-là.

Par exemple, dans le cas d'une commande de 1000 € passée le 15 janvier, le **payment\_config** suivant :

```
{
  ...
  "amount": "100000",
  "payment_config": "MULTI:first=30000;count=4;period=30",
  ...
}
```

correspond à un paiement en 4 fois, comprenant :

- Un premier versement de 300€ le 15 janvier,
- Un second versement de  $(1000 - 300) / 3 = 233,33€$  le 14 février,
- Un troisième versement de 233,33€ le 16 mars,
- Et un dernier versement de 233.34€ le 15 avril.

### **Echéancier personnalisé : MULTI\_EXT**

Dans le cas d'un échéancier personnalisé, tous les montants et les dates de transactions sont spécifiées par la marketplace. Il est ainsi possible de définir des montants variables, et la première transaction peut être ultérieure au jour de la commande.

Il faut cependant respecter certaines contraintes :

- La date de la première échéance ne doit pas être antérieure au jour du paiement.
- Il n'est possible d'enregistrer qu'une transaction par jour.
- Le délai entre la date du paiement et de la dernière transaction ne peut excéder 365 jours.

**Syntaxe** : La valeur de **payment\_config** doit être préfixée de la chaîne « **MULTI\_EXT:** » suivie de la liste des échéances au format date=montant séparées par des « ; ».

Les dates doivent être exprimées au format YYYYMMDD.

Les montants doivent être exprimés dans la plus petite unité de la devise.

Pour reprendre l'exemple précédent mais pour régler cette fois-ci les échéances de la commande au 15 de chaque mois, la valeur de **payment\_config** devra ressembler à :

```
{
  ...
  "amount": "100000",
```



```
"payment_config": "MULTI_EXT:20200115=30000;20200215=23333;20200315=23333;20200415=23334",  
...  
}
```

Note : L'utilisation de la valeur MULTI\_EXT nécessite la souscription de l'option **Paiement en plusieurs fois avancé**.

## 10.2. Paiement avec délai de capture

---

Un délai de capture de la transaction peut être défini lors de la création d'un order. Pour cela, on renseigne l'attribut **capture\_delay** avec le nombre de jours de délai voulu.

Cette valeur doit être comprise entre 0 et 6 :

```
POST https://secure.lyra.com/marketplace/orders?expand=items
```

```
{
  "marketplace": "ebfb36ab-2d30-4326-adb9-e16b0c9a89f3",
  "reference": "Exemple MKP BURO",
  "description": "Commande fournitures",
  "currency": "EUR",
  "capture_delay": "3",
  ...
}
```

Pour modifier la date de capture de la transaction associée à un order déjà exécuté ( de statut PENDING), on ne peut pas utiliser l'attribut **capture\_delay**. Il faut passer "**expected\_capture\_date**" avec la date de capture souhaitée, comprise entre aujourd'hui J et J+6. La date est au format "YYYYMMDDHHMMSS".

```
PUT https://secure.lyra.com/marketplace/orders/515abac9-6cb2-4e21-8a25-b08d7e41e43c
```

```
{
  "marketplace": "ebfb36ab-2d30-4326-adb9-e16b0c9a89f3",
  "reference": "Exemple MKP BURO",
  "description": "Commande fournitures",
  "currency": "EUR",
  "expected_capture_date": "20190425193000",
  ...
}
```

## 10.3. Paiement avec création d'un alias

---

En même temps que l'on crée un order, il est possible de demander l'enregistrement des informations de paiement sous la forme d'un alias qui pourra être réutilisé.

Pour cela, il faut utiliser la commande "**execute/token**" (ou "**execute-embedded/token**") à la place de "execute" lors de la validation de l'order.

Exemple d'appel GET pour figer la commande :

```
GET https://secure.lyra.com/marketplace/orders/515abac9-6cb2-4e21-8a25-b08d7e41e43c/execute/  
token
```

Lors de l'exécution de la commande, un objet nommé "token" est créé. **Il a pour uuid celui de l'Order.**

Il est composé de deux éléments:

- un objet "buyer" tel qu'il est défini dans la commande,
- un attribut "alias" dans lequel seront enregistrées les données du moyen de paiement.

*Consultez le chapitre [Gestion des alias](#) à la page 37 pour plus d'informations sur les tokens et les alias.*

Redirigez ensuite l'acheteur vers l'URL transmise dans **payment\_url** pour procéder au paiement avec enregistrement du moyen de paiement.

Après le paiement, un GET Order ne retournera pas l'alias créé.

Pour obtenir l'alias, réalisez un appel GET token avec l'uuid de l'Order.

Consultez le chapitre Analyser le résultat d'une demande de token à la page 38 pour voir le résultat du Get token.

## 10.4. Paiement par alias

---

Pour utiliser un token dont on a la référence dans un order, on ajoute l'attribut **alias** à la création de l'order :

```
POST https://secure.lyra.com/marketplace/orders?expand=items
```

```
{
  "marketplace": "ebfb36ab-2d30-4326-adb9-e16b0c9a89f3",
  "reference": "Exemple MKP BURO",
  "description": "Commande fournitures",
  "currency": "EUR",
  "alias": "1144951ea7ab42989c97159b3dfc0382",
  ...
}
```

En appelant **"execute"** sur cet order, le paiement se fera avec le token qui a pour alias "1144951ea7ab42989c97159b3dfc0382" via la page de paiement.

En appelant **"execute-embedded"** sur cet order, le paiement se fera avec le token qui a pour alias "1144951ea7ab42989c97159b3dfc0382" via le formulaire embarqué.

En appelant **"execute/token"**, le token sera également mis à jour avec les informations contenues dans l'order, via la page de paiement.

En appelant **"execute-embedded/token"**, le token sera également mis à jour avec les informations contenues dans l'order via le formulaire embarqué.

## 10.5. Paiement à l'initiative du marchand

---

L'utilisation d'un alias lors de la création d'un order permet en outre d'exécuter des paiements à l'initiative du marchand ("Merchant Initiated Transaction") : le paiement s'effectue alors *sans* interaction avec le porteur.

Cette fonctionnalité est notamment utile dans le cas d'abonnements.

Pour ce faire, créez un paiement par alias (voir chapitre Paiement par alias à la page 28), puis utilisez le lien **execute-mit** retourné par le serveur.

Par exemple :

```
POST https://secure.lyra.com/marketplace/orders?expand=items
```

```
{
  "marketplace": "ebfb36ab-2d30-4326-adb9-e16b0c9a89f3",
  "reference": "Exemple MKP BURO",
  "description": "Commande fournitures",
  "currency": "EUR",
  "alias": "1144951ea7ab42989c97159b3dfc0382",
  ...
}
```

Dans la réponse du serveur, l'attribut **links** contient le lien **execute-mit** à appeler.

```
{...
  "links": {
    "execute-mit": {
      "href": "http://monsite.site/marketplace/orders/5a439e70-4ccc-4d86-bf30-223552e2c74f/execute-mit"
    },
    ...
  }
}
```

## 10.6. Paiement en préautorisation (validation manuelle)

Par défaut, la validation de l'Order se fait de façon automatique au moment de l'étape 2 du paiement (lors du GET <url>/execute).

Il est également possible de dissocier les étapes d'autorisation de paiement et de remise en banque des transactions.

Cela permet de réaliser des autorisations alors que le dispatching final des items du panier entre montant vendeurs et montant commission marketplace n'est pas encore connu.

### Étape 1


Pour mettre en œuvre cette fonctionnalité, il suffit d'ajouter l'attribut **awaiting\_validation** lors de la création de l'Order et lui assigner la valeur **true**.

Exemple d'appel POST ORDER de création en validation manuelle:

```
POST https://secure.lyra.com/marketplace/orders?expand=items
```

```
{
  "marketplace": "ebfb36ab-2d30-4326-adb9-e16b0c9a89f3",
  "reference": "Exemple MKP BURO",
  "description": "Commande fournitures",
  "currency": "EUR",
  "url_return": "https://URLreturn.com",
  "awaiting_validation" : true,
  "items": [
    {
      "seller": "5d0ef88c-3345-4b33-948e-80e23d553b73",
      "reference": "STB1",
      "description": "Stylos bleus",
      "amount": 1150,
      "is_commission": false
    }
  ],
  "buyer": {
    "type": "PRIVATE",
    "email": "jean.dupont@acheteur.fr",
    "phone_number": "33 (0)1 11 22 33 44",
    "reference": "ZhIsYM"
  },
  "shipping": {
    "shipping_method": "RELAY_POINT"
  }
}
```

Dans cet exemple, le montant de l'item (montant acheteur) est connu, mais pas sa répartition entre la part sous-marchand et la part de la commission de l'opérateur (la marketplace).

 **Remarque :** Notez toutefois qu'il est possible de préciser dès cette étape les commissions de l'opérateur. Ces dernières pourront, si nécessaire, être mises à jour à l'étape 4.

### Étape 2

L'étape 2 est exécutée de la même manière que pour un paiement standard et l'url de redirection pour procéder au paiement est retournée.

Exemple d'appel GET pour figer la commande :

```
GET https://secure.lyra.com/marketplaces/marketplace/
orders/7fac13b0-7ab9-4382-9073-11ddb38d4427/execute
```

L'Order étant en attente de validation, les contrôles sur le montant minimum de commission ne sont pas exécutés.

### Etape 3

Une fois le paiement effectué avec succès, l'order passe à l'état PENDING et les transactions associées à au statut **TO\_VALIDATE**.

### Etape 4

Cette étape, facultative, permet d'indiquer le montant des commissions de l'opérateur si ces dernières n'ont pas été fournies à l'étape 1.

Il est ainsi possible d'ajouter :

- un item commission
- et/ou des commissions sur les items (cf paragraphe dédié).

Un ou plusieurs items peuvent également être supprimés (fonction standard).

Le montant total de l'Order peut être diminué mais il ne peut pas être augmenté.

En revanche, l'attribut **awaiting\_validation** ne peut pas être modifié.

### Etape 5

Le processus se termine par la validation de l'Order. C'est lors de cette étape que l'on contrôle que le montant de commission est suffisant.

Pour valider un Order, il faut faire un POST sur l'url de l'Order, attribut **validate**.

Exemple d'appel POST ORDER pour le valider manuellement :

```
POST https://secure.lyra.com/marketplace/orders/7fac13b0-7ab9-4382-9073-11ddb38d4427/validate
```

## Exemple de retour du POST ORDER valide :

Le détail de l'Order est retourné. L'attribut **awaiting\_validation** de l'Order prend alors la valeur **false** et les transactions associées passent de l'état **TO\_VALIDATE** à **PENDING**.

```
{
  "uuid": "7fac13b0-7ab9-4382-9073-11ddb38d4427",
  "href": "https://secure.lyra.com/marketplace/orders/7fac13b0-7ab9-4382-9073-11ddb38d4427",
  "created_at": "2019-05-27T09:30:24.434556Z",
  "updated_at": "2019-05-27T09:36:06.500936Z",
  "marketplace": "ebfb36ab-2d30-4326-adb9-e16b0c9a89f3",
  "reference": "Exemple MKP BURO 2",
  "description": "Commande fournitures",
  "alias": null,
  "awaiting_validation": false,
  "buyer": [
    {
      "reference": "ZhIsYM",
      "title": null,
      "type": "PRIVATE",
      "first_name": null,
      "last_name": null,
      "legal_name": null,
      "phone_number": "33 (0)1 11 22 33 44",
      "email": "jean.dupont@acheteur.fr",
      "address": null
    }
  ],
  "shipping": [
    {
      "shipping_method": "RELAY_POINT",
      "delivery_company_name": null,
      "shipping_speed": null,
      "shipping_delay": null,
      "type": null,
      "first_name": null,
      "last_name": null,
      "legal_name": null,
      "phone_number": null,
      "address": {
        "street_number": "37",
        "street": "rue Marcel Philippe",
        "district": "Wallis-et-Futuna",
        "zipcode": "59259",
        "city": "Dupont",
        "state": "Meurthe-et-Moselle",
        "country": "FR"
      }
    }
  ],
  "payment_config": "SINGLE",
  "amount": 1150,
  "initial_amount": 1150,
  "currency": "EUR",
  "status": "PENDING",
  "webhook_result": null,
  "url_return": "https://URLreturn.com",
  "links": {
    "items": {
      "href": "https://secure.lyra.com/marketplace/orders/7fac13b0-7ab9-4382-9073-11ddb38d4427/items" },
    "transactions": {
      "href": "https://secure.lyra.com/marketplace/orders/7fac13b0-7ab9-4382-9073-11ddb38d4427/transactions" },
    "refunds": {
      "href": "https://secure.lyra.com/marketplace/orders/7fac13b0-7ab9-4382-9073-11ddb38d4427/refunds" }
    },
  "vads_transaction_id": 600001,
  "vads_transaction_date": "20190527093234",
  "expected_capture_date": "20190530093235",
  "capture_delay": 3
}
```



**Important :** L'Order peut être validé tant que la date d'expiration de la demande d'autorisation n'est pas dépassée. Si cette date est dépassée alors le paiement prend le statut définitif **EXPIRED**.



## 10.7. Paiement par Titres-Restaurant Dématérialisés

L'application Marketplace permet le paiement partiel ou en intégralité des items du panier au moyen de Titres-Restaurant dématérialisés (TRD).

Cette fonctionnalité ne concerne que les marketplaces éligibles TRD.

Elle nécessite au préalable de renseigner les contrats TRD des marchands dans la marketplace.

### 1. Configuration des marchands

Les marchands sont configurés via le Back Office Marketplace ou lors de la procédure d'enrôlement. Le numéro de contrat TRD du marchand lui est affecté en renseignant l'attribut **trd\_contract**.

Exemple de POST REGISTRATION (extrait) :

```
{
  "marketplace": "d085b6bb-4b8e-4f23-b84a-c9e5fafaf3dd",
  "name": "Acme Corporation",
  "person_type": "PM",
  "trading_name": "Acme Corp.",
  "nature": "SARL",
  "reference": "registration002",
  "trd_contract": "1234567",
  "iban": "FR76000000000000000000",
  "vat_number": "FRXX123456789",
  "legal_form": 5710,
  "external_ref": "000123",
  ...
}
```

### 2. Préparation de la commande

Pour chaque *item* éligible à un paiement TRD, indiquez le montant maximum payable en TRD au moment de créer l'**order** dans la marketplace. Ce montant est renseigné par l'attribut **max\_trd\_amount** et il ne représente pas obligatoirement l'intégralité du montant de l'*item*.

Exemple de POST ORDER avec des items éligibles TRD :

```
{
  "marketplace": "d085b6bb-4b8e-4f23-b84a-c9e5fafaf3dd",
  "reference": "cmd00054",
  "description": "Commande éligible paiement TRD",
  "currency": "EUR",
  "url_return": "https://lyra.com/doc/fr/",
  "language": "fr",
  "capture_delay": 3,
  "items": [
    {
      "seller": "fa740b68-d873-4eab-9639-88a8ac413e42",
      "reference": "demoeYDPnZuC",
      "description": "Bon menu",
      "amount": 1500,
      "type": "FOOD_AND_GROCERY",
      "is_commission": false,
      "max_trd_amount": 1000
    },
    {
      "seller": "fa740b68-d873-4eab-9639-88a8ac413e42",
      "reference": "azraty",
      "description": "Bon vin",
      "amount": 2000,
      "type": "FOOD_AND_GROCERY",
      "is_commission": false
    },
    {
      "seller": "4d20a9d4-0526-4474-b452-e936dc25418d",
      "reference": "pemo9knR",
      "description": "Bon café",
      "amount": 500,
      "type": "FOOD_AND_GROCERY",
      "is_commission": false,
      "max_trd_amount": 500
    }
  ],
  "buyer": {

```

```

    "type": "PRIVATE",
    "email": "paulette17@goncalves.org",
    "phone_number": "+33 (0)5 11 08 12 32",
    "reference": "ZhIsYM"
  },
  "shipping":
  {
    "address":
    {
      "zipcode": "59259",
      "street_number": "37",
      "country": "FR",
      "state": "Meurthe-et-Moselle",
      "street": "rue Marcel Philippe",
      "district": "Wallis-et-Futuna",
      "city": "Dupont"
    },
    "shipping_method": "ETICKET",
    "delay": 2
  }
}

```

### Exemple de POST ORDER avec des items mixtes TRD et non TRD :

```

{
  "marketplace": "d085b6bb-4b8e-4f23-b84a-c9e5fafaf3dd",
  "reference": "cmd00077",
  "description": "Commande éligible paiement TRD et NON TRD",
  "currency": "EUR",
  "url_return": "https://lyra.com/doc/fr/",
  "language": "fr",
  "items": [
    {
      "seller": "fa740b68-d873-4eab-9639-88a8ac413e42",
      "reference": "jambonbeurre",
      "description": "Restaurant",
      "amount": 1550,
      "quantity": 1,
      "max_trd_amount": 1550
    },
    {
      "seller": "fa740b68-d873-4eab-9639-88a8ac413e42",
      "reference": "mug",
      "description": "MugDuRestaurant",
      "amount": 950,
      "quantity": 1
    },
    {
      "seller": "fa740b68-d873-4eab-9639-88a8ac413e42",
      "reference": "commisison",
      "description": "commission",
      "amount": 160,
      "quantity": 1,
      "is_commission": true
    }
  ],
  "buyer": {
    "type": "PRIVATE",
    "email": "paulette17@goncalves.org",
    "phone_number": "+33 (0)5 11 08 12 32",
    "reference": "ZhIsYM"
  },
  "shipping":
  {
    "address":
    {
      "zipcode": "59259",
      "street_number": "37",
      "country": "FR",
      "state": "Meurthe-et-Moselle",
      "street": "rue Marcel Philippe",
      "district": "Wallis-et-Futuna",
      "city": "Dupont"
    },
    "shipping_method": "ETICKET",
    "delay": 2
  }
}

```

Au moment du **GET** sur l'**order** pour figer la commande, le montant total de la commande est calculé ainsi que le montant éligible au paiement TRD.

Un seul marchand par commande peut être payé par TRD.

Pour ce faire, l'application utilise l'algorithme de sélection suivant :

- calcul du montant total payable en TRD par marchand
- sélection du marchand qui a le plus haut montant payable en TRD

Si le montant éligible est identique entre plusieurs marchands, celui avec le plus petit numéro de contrat sera sélectionné.

Le numéro de contrat du marchand sélectionné est ajouté dans la commande pour être envoyé à la plateforme de paiement ainsi que le montant éligible.

### 3. Paiement

Le paiement est géré par la plateforme Lyra Collect.

- Paiement total en TRD

Le montant intégral de la commande est payé à l'aide du TRD. Une seule transaction est générée dans la marketplace.

- Paiement partiel en TRD

Le montant éligible TRD est inférieur au montant total de la commande, l'acheteur utilise un Titre-Restaurant pour payer la partie éligible finançable par TRD puis complète le restant-dû avec un autre moyen de paiement.

On parle de paiement en cascade car il y a 2 moyens de paiement associés donc 2 transactions distinctes sont générées dans la marketplace.

#### REMARQUE

Même si sa commande est éligible à un paiement TRD, l'acheteur peut décider de ne pas payer par Titre-Restaurant Dématérialisé.

Il est possible d'annuler une commande comprenant des transactions CONECS. En revanche, la modification d'une transaction CONECS n'est pas possible.

## 11. ANALYSER LE RÉSULTAT DU PAIEMENT

Pour traiter le résultat du paiement, la Marketplace sera notifiée sur une url fournie à Lyra Collect, attribut : *webhook\_url*.

Un paramètre fourni dans cette notification permettra à la Marketplace de faire un GET ORDER afin de récupérer l'état de l'ORDER et donc savoir si le paiement a réussi.

Exemple de notification (webhook) :

```
{ 'order': '515abac9-6cb2-4e21-8a25-b08d7e41e43c' }
```

Exemple d'appel GET ORDER

GET <https://secure.lyra.com/marketplace/orders/515abac9-6cb2-4e21-8a25-b08d7e41e43c/>

Exemple de retour GET ORDER

```
{
  "uuid": "515abac9-6cb2-4e21-8a25-b08d7e41e43c",
  "href": "https://secure.lyra.com/marketplace/orders/515abac9-6cb2-4e21-8a25-b08d7e41e43c",
  "created_at": "2015-03-19T16:30:14.434Z",
  "updated_at": "2015-03-19T16:30:14.434Z",
  "marketplace": "9537e049-8862-400a-ae8d-da2ec9ca6051",
  "reference": "cmd00052",
  "description": "Commande",
  "buyer": [{
    "reference": "nope775",
    "title": "M.",
    "type": "PRIVATE",
    "first_name": "Simone",
    "last_name": "Legendre",
    "phone_number": "02 13 06 95 27",
    "email": "ndurand@tiscali.fr",
    "address": {
      "street_number": "29",
      "street": "rue Besnard",
      "district": "Île-de-France",
      "zipcode": "83819",
      "city": "Roux",
      "state": "Hautes-Pyrénées",
      "country": "FR"
    }
  ]},
  "shipping": [ {
    "delivery_company_name": "DHL",
    "address": {
      "street number": "493",
      "street": "avenue Duhamel",
      "district": "Saint-Martin",
      "zipcode": "33980",
      "city": "Hamon",
      "state": "Charente-Maritime",
      "country": "FR"},
    "shipping_speed": "EXPRESS",
    "shipping_method": "ETICKET",
    "type": "PRIVATE",
    "first_name": "Luc",
    "last_name": "Leveque",
    "phone_number": "+33 (0)1 46 05 15 89"
  } ],
  "amount": null,
  "currency": "EUR",
  "status": "CREATED",
  "webhook_result": null,
  "url_return": "http://www.lyra-sms.com/",
  "links": {
    "items": {
      "href": "https://secure.lyra.com/marketplace/orders/515abac9-6cb2-4e21-8a25-b08d7e41e43c/items",
      "execute": {
        "href": "https://secure.lyra.com/marketplace/orders/515abac9-6cb2-4e21-8a25-b08d7e41e43c/execute" }
    }
  },
  "vads_transaction_id": 500003,
  "vads_transaction_date": ""
}
```

## 12. GESTION DES ALIAS

---

L'API Marketplace utilise deux notions :

- les **alias**
- les **tokens**

Un **alias** est un objet qui permet de stocker les données d'un moyen de paiement.

Un **token** est un objet qui est associé à un alias. Il contient notamment les informations de l'acheteur.

Il y a 2 façons de créer un alias (et donc un token) :

1. durant le paiement (voir chapitre Paiement avec création d'un alias à la page 26). L'alias et le token sont donc associés à un Order,
2. sans passer par la création d'un Order (voir chapitre suivant).

### 12.1. Créer un token

---

Il est possible de demander l'enregistrement du moyen de paiement (et donc la création d'un token) sans qu'il ne soit lié à une commande.

Pour cela, il faut appeler l'API TOKEN pour créer un token, avec une requête proche de celle utilisée pour une création de paiement.

Pour créer un token en utilisant la page de paiement :

Exemple d'appel POST TOKEN:

```
POST https://secure.lyra.com/marketplace/tokens/
```

```
{
  "marketplace": "9537e049-8862-400a-ae8d-da2ec9ca6051",
  "url_return": "http://www.my-website.com/",
  "buyer": {
    "type": "PRIVATE",
    "first_name": "Jean",
    "last_name": "Dupond",
    "email": "jean.dupond@lyra.fr",
    "phone_number": "1234",
    "reference": "Acheteur_1",
    "address": {
      "zipcode": "59259",
      "street_number": "37",
      "country": "FR",
      "street": "rue Marcel Philippe",
      "city": "Vignoux"
    }
  }
}
```

Exemple de retour POST TOKEN :

```
{
  "token": "d3329266-c8d6-421c-8d2a-10a8ffbcaef6",
  "payment_url": "https://secure.lyra.com/vads-payment/
exec.refresh.a;jsessionid=335D2aDb5eF8356Aed2cf3dF.vadpayment02inte011bg?
cacheId=913355311811276000040"
}
```

Une fois le token créé, il faut rediriger l'acheteur vers l'URL transmise dans **payment\_url** pour procéder à l'enregistrement du moyen de paiement (et donc à la création d'alias).

### **Pour créer un token en utilisant le formulaire embarqué :**

#### **Exemple d'appel POST TOKEN:**

```
POST https://secure.lyra.com/marketplace/tokens/embedded/
```

```
{
  "marketplace": "9537e049-8862-400a-ae8d-da2ec9ca6051",
  "url_return": "http://www.my-website.com/",
  "buyer": {
    "type": "PRIVATE",
    "first_name": "Jean",
    "last_name": "Dupond",
    "email": "jean.dupond@lyra.fr",
    "phone_number": "1234",
    "reference": "Acheteur_1",
    "address": {
      "zipcode": "59259",
      "street_number": "37",
      "country": "FR",
      "street": "rue Marcel Philippe",
      "city": "Vignoux"
    }
  }
}
```

#### **Exemple de retour POST TOKEN :**

```
{
  "token": "d3329266-c8d6-421c-8d2a-10a8ffbcaef6",
  "form_token": "fa0yfV2FQuR3aak1SwsgcuZg195eyJhbW91bnQiOjEwOTAwLCJjdXJyZW5jeSI6IkVVUuIsIm1vZGUuOiJURVNUiIiwidmVyc2l2b3I6Mywib3JkZXJJZCI6IjVjNDk1N2E5LWNkOGEtNgJhZC1hMGNiLTgxMTZiMWE3ZDdmNiIsInNob3BOYW11IjoiTHlyYSBTTVMiLCJicmFuZFByaW9yaXR5IjpbIkJBTknPT1RBQ1QiL CJDQiIsIkUtQ0FSVEVCTEVVR5IsIk1BU1RFUkNBUkQiLCVGVQVVMVCJ9fX0a702"
}
```

Une fois le token créé, le **form\_token** doit être inséré dans le formulaire (attribut **kr-form-token**) pour procéder à l'enregistrement du moyen de paiement (et donc à la création d'alias).

## **12.2. Analyser le résultat d'une demande de token**

Comme pour un paiement, la Marketplace sera notifiée sur l'URL **webhook\_url** de l'évolution d'une demande de token.

#### **Exemple de notification (webhook):**

```
{"token": "d3329266-c8d6-421c-8d2a-10a8ffbcaef6"}
```

L'identifiant ainsi transmis permettra à la Marketplace de faire un GET TOKEN afin de savoir si la création de l'alias a réussi.

#### **Remarque :**

Lors d'un paiement avec création d'alias, vous recevrez donc deux notifications : une pour la création de l'Order et une pour la création de l'alias (il faudra utiliser l'ID du token retourné dans la notification pour retrouver l'alias correspondant).

### Exemple d'appel GET TOKEN :

```
GET https://secure.lyra.com/marketplace/tokens/d3329266-c8d6-421c-8d2a-10a8ffbcaef6
```

### Exemple de retour GET TOKEN :

```
{
  "uuid": "d3329266-c8d6-421c-8d2a-10a8ffbcaef6",
  "created_at": "2018-11-20T12:53:51.547541Z",
  "updated_at": "2018-11-20T12:53:52.513541Z",
  "marketplace": "9537e049-8862-400a-ae8d-da2ec9ca6051",
  "buyer": {
    "reference": "Acheteur_1",
    "title": "MR",
    "type": "PRIVATE",
    "first_name": "Jean",
    "last_name": "Dupond",
    "phone_number": "012345678",
    "email": "jean.dupond@lyra.fr",
    "address": {
      "street_number": "37",
      "street": "rue Marcel Philippe",
      "district": null,
      "zipcode": "59259",
      "city": "Martin",
      "state": null,
      "country": "FR"
    }
  },
  "language": "fr",
  "url_return": "http://www.my-website.com/",
  "status": "SUCCEDEED",
  "alias": "1144951ea7ab42989c97159b3dfc0382",
  "alias_to_update": null,
  "payment_url": "https://secure.lyra.com/vads-payment/
exec.refresh.a;jsessionid=Cdb37C1CcfC5eA2BE82bDCA6?cacheId=913355311811206000040"
}
```

## 12.3. Modifier un alias

Pour mettre à jour un moyen de paiement, vous devez procéder à une nouvelle création de token en ajoutant l'alias existant dans l'attribut **alias\_to\_update** de la requête POST TOKEN.

### Mettre à jour un alias en utilisant la page de paiement :

#### Exemple d'appel POST TOKEN:

```
POST https://secure.lyra.com/marketplace/tokens/
```

```
{
  "marketplace": "9537e049-8862-400a-ae8d-da2ec9ca6051",
  "alias_to_update": "1144951ea7ab42989c97159b3dfc0382",
  "url_return": "http://www.my-website.com/",
  "buyer": {
    "type": "PRIVATE",
    "first_name": "Jean",
    "last_name": "Dupond",
    "email": "jean.dupond@lyra.fr",
    "phone_number": "1234",
    "reference": "Acheteur_1",
    "address": {
      "zipcode": "59259",
      "street_number": "37",
      "country": "FR",
      "street": "rue Marcel Philippe",
      "city": "Vignoux"
    }
  }
}
```

### Exemple de retour POST TOKEN :

```
{
  "token": "62a52e50-ce29-409a-9cec-9ea6ee36ab41",
  "payment_url": "https://secure.lyra.com/vads-payment/
exec.refresh.a;jsessionid=335D2aDb5eF8356Aed2cf3dF.vadpayment02inte01lbg?
cacheId=913355311811276000040"
}
```

Pour procéder à la mise à jour du moyen de paiement, il faut rediriger l'acheteur vers l'URL transmise dans **payment\_url**.

Vous recevrez une notification webhook à la fin de l'opération.

### Pour créer un token en utilisant le formulaire embarqué :

#### Exemple d'appel POST TOKEN:

```
POST https://secure.lyra.com/marketplace/tokens/embedded/
```

```
{
  "marketplace": "9537e049-8862-400a-ae8d-da2ec9ca6051",
  "alias_to_update": "1144951ea7ab42989c97159b3dfc0382",
  "url_return": "http://www.my-website.com/",
  "buyer": {
    "type": "PRIVATE",
    "first_name": "Jean",
    "last_name": "Dupond",
    "email": "jean.dupond@lyra.fr",
    "phone_number": "1234",
    "reference": "Acheteur_1",
    "address": {
      "zipcode": "59259",
      "street_number": "37",
      "country": "FR",
      "street": "rue Marcel Philippe",
      "city": "Vignoux"
    }
  }
}
```

### Exemple de retour POST TOKEN :

```
{
  "token": "62a52e50-ce29-409a-9cec-9ea6ee36ab41",
  "form_token": "fa0yfV2FQuR3aak1SwsgcuZg195eyJhbW91bnQiOjEwOTAwLCJjdXJyZW5jeSI6IkVVUuIIsIm
1yZGUuOiJURVNUIiwidmVyc2lvbiI6Mywib3JkZXJZCI6IjVjNDk1N2E5LWNkOGEtNGJhZC1hMGNiLTgxMT
ZiMWE3ZDdmNiIsInNob3BOYWllIjoithlyYSBTTVMiLCJicmFuZFByaW9yaXR5IjpbIkJBTkNPTlRBQ1QiL
CjDQiIsIkUtQ0FSVEVCTEVVRSIsIk1BU1RFUkNBUCQiLCVGVQVVMVCJ9fx0a702"
}
```

Pour procéder à la mise à jour du moyen de paiement, le **form\_token** doit être inséré dans le formulaire (attribut **kr-form-token**).



## 12.4. Récupérer le détail d'un alias

---

Une fois qu'un alias a été créé, il est possible d'obtenir ses informations en appelant le service de détail d'un token.

Exemple d'appel GET ALIAS :

```
GET https://secure.lyra.com/marketplace/marketplaces/9537e049-8862-400a-ae8d-da2ec9ca6051/alias/1144951ea7ab42989c97159b3dfc0382
```

Exemple de retour GET ALIAS :

```
{
  "brand": "CB",
  "expiry_month": "6",
  "expiry_year": "2023",
  "number": "497010XXXXXX0000",
  "first_name": "Jean",
  "last_name": "Dupond",
  "email": "jean.dupond@lyra.fr",
  "creation_date": "2019-01-21T16:01:19Z"
}
```

Autre exemple avec un alias de type IBAN :

```
{
  "brand": "SDD",
  "expiry_month": "5",
  "expiry_year": "2023",
  "number": "FR7630002005701234567890158_CRLYFRPPXXX",
  "first_name": "Jean",
  "last_name": "Dupond",
  "email": "jean.dupond@lyra.fr",
  "creation_date": "2020-05-12T14:24:13Z"
}
```

## 13. TRAITER LE RETOUR VERS LA MARKETPLACE

---

Afin de rediriger l'acheteur vers la Marketplace, l'attribut **url\_return** de la ressource ORDER doit être renseigné.

**Remarque** : le retour vers la Marketplace doit uniquement vous permettre d'afficher un contexte visuel à l'acheteur. N'utilisez pas les données reçues pour effectuer le traitement en base de données, ni pour contrôler le statut du paiement.

## 14. MODIFIER UNE COMMANDE

Il est possible de modifier un Order tant qu'il possède un statut transitoire (**CREATED** ou **PENDING**).

La transaction initiale ne doit donc pas être remise en banque.

Dans le cas d'un Order au statut **CREATED**, toutes les valeurs sont modifiables, sans restriction (voir l'étape 2 du chapitre Réaliser un paiement à la page 19).

Dans le cas d'un Order au statut **PENDING**, la commande PUT est rejetée si les deux valeurs suivantes sont identiques à celles du dernier enregistrement :

- **amount**, déterminé à partir de la somme des montants des items.  
La requête de modification annule et remplace l'ensemble des items de l'Order initial.
- **expected\_capture\_date** en UTC au format YYYYMMDDHHMMSS.  
La comparaison des `expected_capture_date` ne tient compte que de la date.  
Par exemple, 20200101100000 (1er janvier 2020 à 10h UTC) est équivalent à 20200101180000 (1er janvier 2020 à 18h UTC).

Par ailleurs

- le montant ne peut être supérieur au montant initial de la commande (cf. valeur enregistrée à la clé **initial\_amount**),
- les items doivent être obligatoirement transmis dans la requête PUT (?expand=items).

Exemple d'appel PUT ORDER :

```
PUT https://secure.lyra.com/marketplace/orders/515abac9-6cb2-4e21-8a25-b08d7e41e43c?
expand=items
```

```
{
  "marketplace": "ebfb36ab-2d30-4326-adb9-e16b0c9a89f3",
  "reference": "Exemple MKP BURO",
  "description": "Commande fournitures",
  "currency": "EUR",
  "url_return": "https://URLreturn.com",
  "expected_capture_date": "202006205352",
  "items": [
    {
      "seller": "d0f80202-0676-4d8d-9247-f455f30aec1b",
      "reference": "commburo",
      "description": "commission BURO",
      "amount": 150,
      "is_commission": true
    },
    {
      "seller": "5d0ef88c-3345-4b33-948e-80e23d553b73",
      "reference": "STV1",
      "description": "Stylos verts",
      "amount": 1000,
      "is_commission": false
    },
    {
      "seller": "d0f80202-0676-4d8d-9247-f455f30aec1b",
      "reference": "liv",
      "description": "Livraison",
      "amount": 500,
      "is_commission": false
    }
  ],
  "buyer": {
    "type": "PRIVATE",
    "email": "jean.dupont@acheteur.fr",
    "phone_number": "+33 (0)1 11 22 33 44",
    "reference": "ZhIsYM"
  },
  "shipping": {
    "address": {
      "zipcode": "59123",
```

```
"street_number": "37",
"country": "FR",
"street": "rue Marcel Philippe",
"city": "Nantes"
},
"shipping_method": "RELAY_POINT"
}
}
```

## 14.1. Mise à jour d'une commande réglée en plusieurs fois

---

Il est possible de modifier les transactions après le paiement et tant que les transactions en question sont en attente (c'est-à-dire qu'elles doivent se trouver à l'état PENDING).

Seules deux valeurs sont modifiables :

- le montant de la transaction (attribut **amount**) ;
- la date de capture de la transaction (attribut **expected\_capture\_date**).

Notes :

- Tout comme avec une commande mono-transactionnelle, il n'est pas possible d'accroître le montant d'une transaction au-delà de son montant initial, tel qu'enregistré dans l'attribut en lecture seule **initial\_amount**.
- Les items doivent être obligatoirement transmis dans la requête PUT (?expand=items).
- Il est possible de supprimer une (ou plusieurs) transaction(s) (cf. ci-dessous), mais il n'est actuellement pas possible d'en ajouter.

### Procédure :

La modification d'une ou de plusieurs transactions passe par la même commande de mise à jour d'une commande (PUT ORDER), mais avec deux différences :

1. Il faut ajouter la liste complète des transactions de la commande dans le corps de la requête PUT ORDER dans une liste de **transactions**, comprenant chacune les attributs **trans\_uid**, **expected\_capture\_date** et **amount** éventuellement modifiés.

Par exemple :

```
{
  "marketplace": "2434c0a2-9d46-4e96-9553-1536c898625b",
  "reference": "MyMultitransactionOrder01",
  "description": "Mise à jour de la commande.",
  "currency": "EUR",
  "url_return": "https://www.my-website.com/",
  "language": "en",
  "items": [
    {
      "seller": "4d20a9d4-0526-4474-b452-e936dc25418d",
      "reference": "0000001",
      "description": "TV Ecran plat",
      "amount": 390000,
      "quantity": 1,
      "is_commission": false
    },
    {
      "seller": "72ccc2ff-b455-4653-847e-deb6fee99f8d",
      "reference": "0000002",
      "description": "Commission",
      "amount": 14755,
      "quantity": 1,
      "is_commission": true
    }
  ],
  "transactions": [
    {
      "trans_uid": "c383739d12a6489badc3bb6847db84cc",
      "payment_scheme": "CB",

```

```

    "amount": 14755,
    "expected_capture_date": "201909185743"
  },
  {
    "trans_uuid": "d0c4d34249d540af87ff8df3a2fa314a",
    "payment_scheme": "CB",
    "amount": 190000,
    "expected_capture_date": "201905175352"
  },
  {
    "trans_uuid": "8c5a6788b3334d368185b0a567dd7bcd",
    "payment_scheme": "CB",
    "amount": 200000,
    "expected_capture_date": "201904205352"
  }
],
"buyer": {
  ...
}
}

```

**2.** Il est impératif d'ajouter "**transactions**" dans l'attribut url « **expand** ».

Vous devez donc envoyer une ressource PUT ORDER sur

**[https://secure.lyra.com/marketplace/orders/{order\\_uuid}?expand=items,transactions](https://secure.lyra.com/marketplace/orders/{order_uuid}?expand=items,transactions)**.

Si la valeur "**transactions**" n'est pas indiquée, aucune modification de transaction ne pourra être prise en compte.

Lors de la mise à jour, l'application vérifie la cohérence des montants entre les items et les transactions : en toute logique, le total des montants de l'item doit être identique au montant total des transactions. Et par conséquent, une réduction des montants sur les items doit mécaniquement s'accompagner d'une réduction des montants sur les transactions.

*Cas particulier d'une suppression de transactions*

Il se peut que la réduction des montants sur la partie items soit supérieure au montant d'une des transactions. Si au lieu de répartir la réduction sur l'ensemble des transactions, vous choisissez de supprimer une transaction, il suffit de l'omettre dans la liste des transactions envoyées en requête de modification. Elle est alors interprétée comme annulée, et son statut passe à CANCELLED.

Autres remarques :

- L'application renvoie une erreur si la mise à jour ne comporte aucune modification (ni des transactions, ni des items, ni des valeurs de la commande).
- La nouvelle valeur de **expected\_capture\_date** ne doit pas être antérieure à la date et heure de modification.

## 15. ANNULER UNE COMMANDE

---

L'annulation de la commande est possible sur les commandes avant paiement et après paiement tant que ce dernier n'a pas été remisé en banque, c'est à dire lorsqu'elle se trouve à l'état:

- **CREATED** et *avant l'exécution de la commande*,
- **PENDING**.

Si vous essayez d'annuler le paiement entre le moment de l'exécution de la commande (alors encore à l'état CREATED) et la fin du processus du paiement (i.e. le passage de CREATED à PENDING), vous recevrez l'erreur suivante :

```
{'Order': "This order has been executed and is not cancelable until the end of the payment process"}
```

Exemple d'appel DELETE ORDER :

```
DELETE https://secure.lyra.com/marketplace/orders/60c9dbf5-ff99-40fb-9fb6-a709005359f8
```

En cas de succès, le serveur répond par un code **HTTP 204 (NO CONTENT)**.

## 16. REMBOURSER UN PAIEMENT

---

Pour rembourser un paiement, vous devez indiquer sur quelle commande (ORDER) porte le remboursement et quel vendeur (SELLER) va assumer le coût du remboursement.

Pour demander le remboursement d'un paiement, vous suivez les 2 étapes suivantes :

1. Créer une demande de remboursement
2. Suivre l'évolution de la demande

### 16.1. Créer la demande de remboursement

---

Cette étape permet de créer la demande de remboursement en faisant un POST sur la ressource REFUND

```
POST https://secure.lyra.com/marketplace/refunds/
```

A la création d'une demande de remboursement, les éléments suivants sont contrôlés par l'API et font l'objet d'un code de retour d'erreur 400 s'ils ne sont pas vérifiés :

- La commande à rembourser doit être dans l'état SUCCEEDED.
- Le vendeur qui rembourse doit être un des vendeurs de la commande remboursée.
- Le remboursement ne doit cibler qu'un seul vendeur. Si plusieurs vendeurs doivent rembourser une partie d'une même commande, plusieurs demandes de remboursements distinctes devront être créées : une pour chaque vendeur. La seule exception étant si le deuxième vendeur de la demande est le gestionnaire de la place de marché.

**Remarque** : le montant remboursé ne fait pas l'objet de contrôle à cette étape. Le contrôle a lieu ultérieurement, par les chaînes de traitement planifiées, ce qui se traduit par l'évolution du statut de la demande de traitement.

## Exemple d'appel POST REFUND

Dans cet exemple de contenu JSON à poster, une demande de remboursement de 130€ est formulée, dont 120€ à la charge du vendeur et 10€ à la charge du gestionnaire de la place de marché:

```
POST https://secure.lyra.com/marketplace/refunds/
```

```
{
  "order": "9537e049-8862-400a-ae8d-da2ec9ca6051",
  "reference": "remb000045",
  "description": "Remboursement Mme Lafont 001",
  "currency": "EUR",
  "items": [
    {
      "seller": "dfc42a76-10b5-421a-91cd-c288c8265c92",
      "reference": "remb000045a",
      "description": "Phoneshop",
      "amount": 12000
    },
    {
      "seller": "975e2a43-7e72-438c-a2b2-b61347aa160c",
      "reference": "remb000045b",
      "description": "Gestionnaire",
      "amount": 1000
    }
  ]
}
```

## Exemple réponse du POST REFUND

L'attribut **amount** en retour est calculé automatiquement à partir des montants postés de chacune des parts.

```
{
  "uuid": "16ad9da8-b9cb-11e4-97c6-b1229586dec7",
  "href": "https://secure.lyra.com/marketplace/refunds/16ad9da8-b9cb-11e4-97c6-b1229586dec7",
  "created_at": "2018-06-08T12:36:56.681073Z",
  "updated_at": "2018-06-08T12:39:46.859402Z",
  "order": "9537e049-8862-400a-ae8d-da2ec9ca6051",
  "reference": "remb000045",
  "description": "Remboursement Mme Lafont 001",
  "amount": 13000,
  "currency": "EUR",
  "status": "CREATED",
  "items": [
    {
      "seller": "dfc42a76-10b5-421a-91cd-c288c8265c92",
      "reference": "remb000045a",
      "description": "Phoneshop",
      "amount": 12000,
      "is_commission": false
    },
    {
      "seller": "975e2a43-7e72-438c-a2b2-b61347aa160c",
      "reference": "remb000045b",
      "description": "Gestionnaire",
      "amount": 1000,
      "is_commission": false
    }
  ]
}
```



## 16.2. Suivre l'évolution de la demande de remboursement

---

Les remboursements sont exécutés en fonction des soldes disponibles du marchand et de la Marketplace. Par exemple, un remboursement de 100 euros avec 80 euros à la charge du marchand et 20 euros à la charge de la Marketplace ne sera exécuté que si :

- le solde disponible du sous-marchand est supérieur ou égal à 80 euros
- et si le solde disponible de la Marketplace est supérieur de 20 euros.

Pour suivre l'évolution de la demande, consultez le statut du REFUND par un GET :

```
GET https://secure.lyra.com/marketplace/refunds/{identifiant}/
```

Une demande de remboursement peut passer par les états suivants au cours de son cycle de vie :

- **CREATED** : Demande de remboursement enregistrée par l'API, en attente de traitement
- **PENDING** : Demande adressée aux chaînes de traitement, en cours de traitement
- **SUCCEEDED** : Traitement du remboursement effectué avec succès
- **FAILED** : Demande de remboursement rejetée, par exemple en raison d'un solde insuffisant sur le compte du vendeur
- **CANCELLED** : Demande de remboursement annulée

### Exemple réponse du GET REFUND


```
{
  "uuid": "16ad9da8-b9cb-11e4-97c6-b1229586dec7",
  "href": "https://secure.lyra.com/marketplace/refunds/16ad9da8-b9cb-11e4-97c6-b1229586dec7",
  "created_at": "2018-06-08T12:36:56.681073Z",
  "updated_at": "2018-06-08T12:51:21.241448Z",
  "order": "9537e049-8862-400a-ae8d-da2ec9ca6051",
  "reference": "remb000045",
  "description": "Remboursement Mme Lafont 001",
  "amount": 13000,
  "currency": "EUR",
  "status": "PENDING",
  "items":
  [
    {
      "seller": "dfc42a76-10b5-421a-91cd-c288c8265c92",
      "reference": "remb000045a",
      "description": "Phoneshop",
      "amount": 12000,
      "is_commission": false
    },
    {
      "seller": "975e2a43-7e72-438c-a2b2-b61347aa160c",
      "reference": "remb000045b",
      "description": "Gestionnaire",
      "amount": 1000
      "is_commission": false
    }
  ]
}
```

## 16.3. Modifier une demande de remboursement

Un remboursement en cours peut être modifié dans la limite de son montant initial tant qu'il n'a pas été exécuté (statut CREATED ou PENDING).

La modification s'effectue à l'aide de l'uuid du remboursement en question, à travers la ressource:

```
PUT https://secure.lyra.com/marketplace/refunds/{uuid}
```

 **Remarque :** Si le montant du remboursement est inchangé, le serveur renvoie une erreur HTTP 400.

### Exemple d'appel PUT REFUND

```
PUT https://secure.lyra.com/marketplace/refunds/16ad9da8-b9cb-11e4-97c6-b1229586dec7
```

```
{
  "order": "9537e049-8862-400a-ae8d-da2ec9ca6051",
  "reference": "remb000045",
  "description": "Remboursement Mme Lafont 001",
  "currency": "EUR",
  "items": [{
    "seller": "dfc42a76-10b5-421a-91cd-c288c8265c92",

    "reference": "remb000045a",
    "description": "Phoneshop",
    "amount": 9450,
    "is_commission": false
  },
  {
    "seller": "975e2a43-7e72-438c-a2b2-b61347aa160c",

    "reference": "remb000045b",
    "description": "Gestionnaire",
    "amount": 900,
    "is_commission": false
  }
]}
```

### Exemple réponse du PUT REFUND

```
{
  "uuid": "83402cc0-d969-443b-a72a-0f2fe9557879",
  "href": "https://secure.lyra.com/marketplace/refunds/83402cc0-d969-443b-a72a-0f2fe9557879",
  "created_at": "2019-06-11T09:31:59.659687Z",
  "updated_at": "2019-06-11T09:52:51.959406Z",
  "order": "67ca248e-99cd-4345-9795-5873bb6fd8f2",
  "reference": "remb000045",
  "description": "Remboursement Mme Lafont 001",
  "amount": 10905,
  "currency": "EUR",
  "status": "PENDING",
  "transaction": null,
  "items": [{
    "seller": "dfc42a76-10b5-421a-91cd-c288c8265c92",
    "refund": "83402cc0-d969-443b-a72a-0f2fe9557879",
    "item": "9f7c6d31-defb-4299-a3db-71bb214cae9e",
    "reference": "remb000045a",
    "description": "Phoneshop",
    "amount": 9450,
    "is_commission": false
  },
  {
    "seller": "975e2a43-7e72-438c-a2b2-b61347aa160c",
    "refund": "83402cc0-d969-443b-a72a-0f2fe9557879",
    "item": "7b99691b-0492-42ff-a819-302ac178fc7e",
    "reference": "remb000045b",
    "description": "Gestionnaire",
    "amount": 900,
    "is_commission": false
  }
]}
```

## 16.4. Annuler une demande de remboursement

---

Un remboursement en cours peut être annulé tant qu'il n'a pas été exécuté (statut CREATED ou PENDING).

```
DELETE https://secure.lyra.com/marketplace/refunds/{uuid}
```

Par exemple :

```
DELETE https://secure.lyra.com/marketplace/refunds/83402cc0-d969-443b-a72a-0f2fe9557879
```

En cas de succès, le serveur répond par un code **HTTP 204 (NO CONTENT)**.

Si le statut du remboursement ne permet pas l'annulation, le serveur retourne une erreur **HTTP 405**.

## 17. DÉCLENCHEMENT MANUEL DU PAIEMENT DES ITEMS

Dans la cinématique normale, le paiement d'un item de commission intervient automatiquement après l'expiration du délai de blocage des fonds du vendeur lié à l'item.

Il peut arriver que la marketplace souhaite retenir ce paiement afin de contrôler le moment de son déclenchement.

C'est notamment le cas d'une prestation qui s'étale dans le temps. Dans ce cas le client a effectué son paiement avant le début de la prestation, mais le vendeur ne doit recevoir sa contrepartie que suite à la vérification par la marketplace du bon déroulement de la prestation.

Il appartient alors à la marketplace :

- d'indiquer lors de la création de la commande le ou les items concernés par le blocage du paiement, en valorisant l'attribut **hold\_payment** à **true**:

```
"items": [
  {
    "seller": "4d20a9d4-0526-4474-b452-e936dc25418d",
    "reference": "croisiere12345",
    "description": "Croisiere en bateau",
    "amount": 245000,
    "quantity": 1,
    "is_commission": false,
    "hold_payment": true
  },
  {
    "seller": "4d20a9d4-0526-4474-b452-e936dc25418d",
    "reference": "cartemaritime",
    "description": "Carte maritime - La Martinique",
    "amount": 1250, "quantity": 1,
    "is_commission": false
  },
  {
    "seller": "72ccc2ff-b455-4653-847e-deb6fee99f8d",
    "reference": "croisiere12345com",
    "description": "Commission sur croisiere en bateau",
    "amount": 16450,
    "quantity": 1,
    "is_commission": true
  }
]
```

- et, au moment voulu, de libérer le transfert en effectuant une requête POST (sans corps de requête) à l'adresse :

```
POST /items/{id_item}/activate
```

En cas de succès de l'opération, la requête est redirigée (status\_code = 302) vers le détail de l'item, qui indique alors **hold\_payment** : **false**.



### Remarque :

- Le blocage ne peut concerner les items de panier d'un vendeur de type Marketplace (seller avec "is\_marketPlace = true").
- Une fois le paiement de l'item activé, il n'est plus possible de le bloquer à nouveau .
- L'attribut **hold\_payment** n'est pas obligatoire. Il est même préférable de l'omettre (ou bien de le valoriser à "**hold\_payment** : **null**") pour tout item dont le transfert doit être géré automatiquement, afin de distinguer les items ayant fait l'objet d'un blocage des autres.

## 18. CONSULTER LES CASHOUTS DE LA MARKETPLACE

### 18.1. Liste des cashouts

La liste des cashouts d'une marketplace est consultable sur l'URL :

```
GET https://secure.lyra.com/marketplace/cashouts/
```

Les résultats sont retournés sous la forme:

```
{
  "count": 3,
  "next": null,
  "previous": null,
  "results": [
    {
      "href": "/marketplace/cashouts/749e668e-526d-41ec-8e7c-e5afc3d89ddc",
      "uuid": "749e668e-526d-41ec-8e7c-e5afc3d89ddc",
      "seller": "b646ea68-a145-4a5a-8a6f-55e8f68643dd",
      "seller_external_ref": "ref039412",
      "ref": "demoeYDPnZuC",
      "status": "CREATED",
      "amount": 1900,
      "currency": "EUR",
      "captured_at": "2019-04-04",
      "capture_label": null
    },
    {
      "href": "/marketplace/cashouts/b92d904a-8613-4368-98b7-96d75a623d97",
      "uuid": "b92d904a-8613-4368-98b7-96d75a623d97",
      "seller": "f8dcc611-bbaa-411a-8f28-ea2d6e4f49a8",
      "seller_external_ref": "ref09523",
      "ref": "demoeYDPnZuC",
      "status": "CREATED",
      "amount": 1900,
      "currency": "EUR",
      "captured_at": "2019-03-03",
      "capture_label": null
    },
    {
      "href": "/marketplace/cashouts/f9132b0b-8d42-4409-b3e1-c1c6d711688b",
      "uuid": "f9132b0b-8d42-4409-b3e1-c1c6d711688b",
      "seller": "4d20a9d4-0526-4474-b452-e936dc25418d",
      "seller_external_ref": "ref012345",
      "ref": "demoeYDPnZuC",
      "status": "CREATED",
      "amount": 1450,
      "currency": "EUR",
      "captured_at": "2019-01-01",
      "capture_label": null
    }
  ]
}
```

Ces résultats sont triés par date de capture descendante (attribut **captured\_at**).

Ils sont affichés de façon paginée. Il y a 100 cashouts par page.

Les liens "**next**" et "**previous**", quand ils sont valorisés, permettent de naviguer d'une page à une autre.

Il est également possible de filtrer les résultats de cette liste par leur date de capture.

Pour ce faire, il faut spécifier les paramètres **capture\_start\_date** et/ou **capture\_end\_date** de la façon suivante :

```
GET https://.../marketplace/cashouts/?capture_start_date=2019-02-01&capture_end_date=2019-03-31
```

```
GET https://.../marketplace/cashouts/?capture_start_date=2019-02-01
```

```
GET https://.../marketplace/cashouts/?capture_end_date=2019-01-01
```

Si le paramètre **capture\_start\_date** est renseigné mais pas **capture\_end\_date**, celui-ci est fixé à la date du jour (« aujourd'hui »).

Si à l'inverse, c'est le paramètre **capture\_start\_date** qui est omis mais que **capture\_end\_date** est précisé, la sélection prendra les cashouts depuis le début jusqu'à la date donnée.



**Remarque :** Les cashouts dont l'attribut "captured\_at" n'est pas valorisé sont exclus des résultats.

## 18.2. Détails des cashouts

---

La liste des cashouts fournit l'URL pour accéder aux détails de chacun des cashouts, par exemple:

```
"/marketplace/cashouts/f9132b0b-8d42-4409-b3e1-c1c6d711688b".
```

Un GET sur l'URL permet de récupérer les informations sur les virements et les remboursements associés au cashout (avec les détails des commandes et items) :

```
GET https://.../marketplace/cashouts/f9132b0b-8d42-4409-b3e1-c1c6d711688b
```

```
{
  "href": "/marketplace/cashouts/f9132b0b-8d42-4409-b3e1-c1c6d711688b",
  "uuid": "f9132b0b-8d42-4409-b3e1-c1c6d711688b",
  "seller": "4d20a9d4-0526-4474-b452-e936dc25418d",
  "seller_external_ref": "ref012345",
  "ref": "demoeYDPnZuC",
  "status": "CREATED",
  "amount": 1450,
  "currency": "EUR",
  "captured_at": "2019-01-01",
  "capture_label": null,
  "transfers": [
    {
      "uuid": "0a5a8c7e-9c8b-4f16-8a29-0018c5aa20ef",
      "created_at": "2019-04-12T12:45:54.705798Z",
      "updated_at": "2019-05-06T12:15:09.942029Z",
      "item": {
        "uuid": "e8950426-f13c-4b18-8d27-17f2e6bbca8b",
        "ref": "demoeYDPnZuC",
        "desc": "Repas"
      }
    },
    {
      "order": {
        "uuid": "0bfa5ebf-cd09-4fd8-bb78-465d17854b55",
        "ref": "TestMKP",
        "desc": " Test marketplace "
      },
      "amount": 1900,
      "currency": "EUR"
    }
  ],
  "refunds": [
    {
      "uuid": "2760e1dd-252b-4109-816c-efbeafa2eaa2",
      "created_at": "2019-04-10T14:06:34.076796Z",
      "updated_at": "2019-04-10T14:06:35.719923Z",
      "order": {
        "uuid": "06f0fbd9-4a1f-4828-a014-aafda50df703",
        "ref": "TestMKP",
        "desc": "Test marketplace"
      },
      "item": {
        "uuid": "00f5a201-3d0c-49e7-9d05-0e706d04a385",
        "ref": "demoeYDPnZuC",
        "desc": "Repas"
      }
    },
    {
      "amount": 450,
      "currency": "EUR"
    }
  ]
}
```

## 19. DICTIONNAIRE DE DONNÉES

### 19.1. Objet Address

Tous les attributs de cet objet sont obligatoires si au minimum un attribut est renseigné. Donc si aucun attribut n'est renseigné, aucun attribut n'est obligatoire.

Nom	Description	Format	Obligatoire	Exemple
street_number	Numéro de la rue	an..5		12
street	Nom de la rue	ans..255		rue de la bohème
district	Complément d'adresse	ans..127		quartier 15ème
zipcode	Code postal	an..64		75015
city	Ville	an.128		Paris
state	Etat	ans..127		Paris
country	Pays	a2		FR pour France GP pour Guadeloupe PF pour Polynésie Française

### 19.2. Objet Alias

Nom	Description	Format	Exemple
brand	Marque de la carte	string	CB
expiry_month	Mois d'expiration de la carte entre 1 et 12	n..2	3
expiry_year	Année d'expiration sur 4 chiffres	n4	2023
number	Numéro de carte masqué	string	497010XXXXXX0000
first_name	Prénom de l'acheteur	string	Jean
last_name	Nom de l'acheteur	string	Dupond
email	E-mail de l'acheteur	string	jean.dupond@lyra.com
creation_date	Date de création de l'alias	cf. exemple	2020-04-21T16:01:19Z

## 19.3. Objet Buyer

Nom	Description	Format	Obligatoire	Exemple
reference	référence de l'acheteur pour la Marketplace	an..63	X	000012
legal_name	nom de la société	an..63		Dupond & Cie
title	civilité de l'acheteur	an..63		Mme
type	type d'acheteur	enum	X	PRIVATE   COMPANY
first_name	prénom de l'acheteur	an..63		Stéphanie
last_name	nom de l'acheteur	an..63		DUPOND
phone_number	numéro de téléphone de l'acheteur	an..32	X	0612324565
email	adresse e-mail de l'acheteur	ans..150	X	ste.dupond@site.com
address	adresse postale de l'acheteur	objet Address		

## 19.4. Objet Item

Nom	Description	Format	Obligatoire	Exemple
uuid	Identifiant unique de la prestation <i>Généré par Lyra Collect</i>	an..36	N/A	3ea5e574-e198-55d4-ba23-f9405ec4226f
href	URL d'accès à cette ressource <i>Généré par Lyra Collect</i>	cf. exemple	N/A	<a href="https://secure.lyra.com/marketplace/items/3ea5e574-e198-55d4-ba23-f9405ec4226f">https://secure.lyra.com/marketplace/items/3ea5e574-e198-55d4-ba23-f9405ec4226f</a>
created_at	Date et heure de création de la ressource <i>Généré par Lyra Collect</i>	cf. exemple	N/A	2015-01-17T09:39:54.948Z
updated_at	Date et heure de la dernière modification de la ressource <i>Généré par Lyra Collect</i>	cf. exemple	N/A	2015-01-17T09:39:54.948Z
seller	UUID du prestataire	an..36	X	1ea5e574-e198-55d4-ba23-f9405ec4226a
order	UUID de la commande	an..36		8ea5e574-e198-55d4-ba23-f9405ec4226c
reference	Référence de la prestation	an..32+"-"	X	0000000181
description	Description de la prestation	an..255		Prestation blablabla
type	définit le type de l'article contenu dans le panier	enum		voir tableau ci-après.
amount	Si <b>is_commission</b> est valorisé à false : Montant de la prestation. Si <b>is_commission</b> est valorisé à true : Montant de la commission définie au niveau de la commande.	n..12	X	50000
commission_amount	Montant de la commission définie sur l'item.	n..12		500



Nom	Description	Format	Obligatoire	Exemple
is_commission	indique que cet item est une commission	enum		<i>true   false</i> <i>false</i> par défaut
status	Statut de la prestation <i>Généré par Lyra Collect</i>	an..10	N/A	<i>SUCCEDED</i>

## Valeurs de l'attribut type

Valeur	Description
FOOD_AND_GROCERY	Produits alimentaires et d'épicerie
AUTOMOTIVE	Automobile / Moto
ENTERTAINMENT	Divertissement / Culture
HOME_AND_GARDEN	Maison et jardin
HOME_APPLIANCE	Équipement de la maison
AUCTION_AND_GROUP_BUYING	Ventes aux enchères et achats groupés
FLOWERS_AND_GIFTS	Fleurs et cadeaux
COMPUTER_AND_SOFTWARE	Ordinateurs et logiciels
HEALTH_AND_BEAUTY	Santé et beauté
SERVICE_FOR_INDIVIDUAL	Services à la personne
SERVICE_FOR_BUSINESS	Services aux entreprises
SPORTS	Sports
CLOTHING_AND_ACCESSORIES	Vêtements et accessoires
TRAVEL	Voyage
HOME_AUDIO_PHOTO_VIDEO	Son, image et vidéo
TELEPHONY	Téléphonie

## 19.5. Objet Marketplace

Nom	Description	Format	Exemple
uuid	Identifiant unique de la Marketplace <i>Généré par Lyra Collect</i>	an..36	<i>6ea5e574-e198-55d4-ba23-f9405ec4226b</i>
href	URL d'accès à cette ressource <i>Généré par Lyra Collect</i>	cf. exemple	<i>https://secure.lyra.com/marketplace/marketplaces/6ea5e574-e198-55d4-ba23-f9405ec4226b</i>
created_at	Date et heure de création de la ressource <i>Généré par Lyra Collect</i>	cf. exemple	<i>2015-01-17T09:39:54.948Z</i>
updated_at	Date et heure de la dernière modification de la ressource <i>Généré par Lyra Collect</i>	cf. exemple	<i>2015-01-17T09:39:54.948Z</i>
reference	Nom de la boutique Lyra Collect créée pour la Marketplace	an..255	<i>Ex :Nom de la boutique Marketplace</i>
description	Nom de la boutique Lyra Collect créée pour la Marketplace	an..255	<i>Ex :Nom de la boutique Marketplace</i>
bic	Code banque du compte bancaire de la Marketplace	an..11	<i>CMCIC1234</i>
iban	Numéro de compte du compte bancaire de la Marketplace	an..34	<i>FR1212341234123412341234</i>
vads_key	Identifiant de la boutique Lyra Collect de la Marketplace <i>Généré par Lyra Collect</i>	an..8	<i>12345678</i>
vads_cert	Certificat de production Lyra Collect de la Marketplace <i>Généré par Lyra Collect</i>	an..16	<i>123456789123456789</i>
webhook_url	URL de notification de la Marketplace	an..1024	<i>http://marketplace.com/page_traitement_notification.php</i>
status	Statut de la ressource Marketplace <i>Généré par Lyra Collect</i>	an..10	<i>ACTIVE</i>
links	Liens vers les ressources Sellers et Orders de la Marketplace <i>Généré par Lyra Collect</i>	cf. exemple	<pre>{   "sellers": {     "href":     "https://secure.lyra.com/marketplace/marketplaces/6ea5e574-e198-55d4-ba23-f9405ec4226b/sellers"   },   "orders": {     "href":     "https://secure.lyra.com/marketplace/marketplaces/6ea5e574-e198-55d4-ba23-f9405ec4226b/orders"   } }</pre>

## 19.6. Objet Order

Nom	Description	Format	Obligatoire	Exemple
uuid	Identifiant unique de la commande <i>Généré par Lyra Collect</i>	an..36	N/A	8ea5e574-e198-55d4-ba23-f9405ec4226c
href	URL d'accès à cette ressource <i>Généré par Lyra Collect</i>	cf. exemple	N/A	<a href="https://secure.lyra.com/marketplace/orders/8ea5e574-e198-55d4-ba23-f9405ec4226c">https://secure.lyra.com/marketplace/orders/8ea5e574-e198-55d4-ba23-f9405ec4226c</a>
created_at	Date et heure de création de la ressource <i>Généré par Lyra Collect</i>	cf. exemple	N/A	2015-01-17T09:39:54.948Z
updated_at	Date et heure de la dernière modification de la ressource <i>Généré par Lyra Collect</i>	cf. exemple	N/A	2015-01-17T09:39:54.948Z
marketplace	UUID de la Marketplace	an..36	X	6ea5e574-e198-55d4-ba23-f9405ec4226b
reference	Référence de la commande	an..32+"-"	X	0000123
description	Description de la commande	an..255		Commande blablabla
vads_transaction_id	Identifiant de la transaction Lyra Collect	n..6	N/A	379 <i>Remarque : Champ déprécié à cet endroit.</i>
vads_transaction_date	Date et heure de la transaction Lyra Collect au format YYYYMMDDHHMMSS	an..14	N/A	20150320094512 <i>Remarque : Champ déprécié à cet endroit.</i>
buyer	Informations sur l'acheteur	Objet Buyer	X	
alias	Identifiant du token dans la plateforme	ans..64		1144951ea7ab42989c97159b3dfc0382
shipping	Informations sur la livraison	Objet Shipping		
amount	Montant total de la commande, exprimé dans la plus petite unité de la devise.	n..12	N/A	50000 pour 500 EUR.
currency	Devise de la commande (ISO 4217)	a3	X	EUR
language	Langue d'affichage des pages de paiement (ISO 639-1)	a2		FR
capture_delay	Indique le délai en nombre de jours avant remise en banque	Entier positif compris entre 0 et 6	Non. Seulement disponible en création (POST)	3
expected_capture_date	Date de remise en banque souhaitée, dans le fuseau UTC, au format YYYYMMDDHHMMSS.	Date	Non. Seulement disponible en mise à jour (PUT)	20180716083000
status	Statut de la commande <i>Généré par Lyra Collect</i>	an..10	N/A	PENDING
webhook_result	Résultat de l'envoi de la notification à la Marketplace <i>Généré par Lyra Collect</i>	n..3	N/A	200
url_return	URL de retour de l'acheteur sur le site de la Marketplace	an..512		<a href="http://www.sitemarketplace.com/url_de_retour.html">http://www.sitemarketplace.com/url_de_retour.html</a>

Nom	Description	Format	Obligatoire	Exemple
links	Liens vers les ressources Items <i>Générés par Lyra Collect</i>	cf. exemple	N/A	<pre>{   "items": {     "href":       "https://secure.lyra.com/       marketplace/orders/8ea5e574-       e198-55d4-       ba23-f9405ec4226c/items"   } }</pre>
form_token	Token du formulaire embarqué	an		Exemple: ffa0yfv2FQuR3aak1SwsgcuZg195eyJhbW91bnQi OjEwOTAwLCJjdXJyZW5jeSI6IkVVUilsm1vZG UiOiJURVNUliwidmVyc2lvcil6Mywib3JkZlJlZC I6jVjNDk1N2E5LWNkOGEtNGJhZC1hMGNiLT gxMTZiMWE3ZDdmNiIsInNob3BOYW1lIjoiTlI yYSBTTVMiLCJicmFuZFByaW9yaXR5IjpbIkJB TkNPTIRBQ1QiLCJkQilskUtQ0FSVEVCTEVV RSIsIk1BU1RFUKNBukQjLCVGVQVVMVCj9fX 0a702

## 19.7. Objet Refund

Nom	Description	Format	Obligatoire
<b>order</b>	Identifiant de la commande à rembourser	an..36	X
<b>reference</b>	Référence technique du remboursement	an..32	X
<b>status</b>	Statut du remboursement	an..10	X
<b>description</b>	Description du remboursement	ans..255	
<b>currency</b>	Devise de la commande (ISO 4217) Seul la valeur <b>EUR</b> est autorisée	a3	X
<b>items</b>	Liste des parts du remboursement à effectuer, par vendeur	enum	X
<b>items.seller</b>	Identifiant du vendeur devant assumer cette part du remboursement	an..36	X
<b>items.reference</b>	Référence technique	an..32	X
<b>items.description</b>	Description	ans..255	
<b>items.amount</b>	Montant de la part à rembourser par le vendeur	n..12	X
<b>items.item</b>	Référence de l'item sur lequel porte le remboursement au sein de l'order d'origine	an..50	

## 19.8. Objet Seller

Nom	Description	Format	Exemple
uuid	Identifiant unique du prestataire <i>Généré par Lyra Collect</i>	an..36	<i>1ea5e574-e198-55d4-ba23-f9405ec4226a</i>
href	URL d'accès à cette ressource <i>Généré par Lyra Collect</i>	cf. exemple	<i>https://secure.lyra.com/marketplace/sellers/1ea5e574-e198-55d4-ba23-f9405ec4226a</i>
created_at	Date et heure de création de la ressource <i>Généré par Lyra Collect</i>	cf. exemple	<i>2015-01-17T09:39:54.948Z</i>
updated_at	Date et heure de la dernière modification de la ressource <i>Généré par Lyra Collect</i>	cf. exemple	<i>2015-01-17T09:39:54.948Z</i>
marketplace	UUID de la Marketplace	an..36	<i>6ea5e574-e198-55d4-ba23-f9405ec4226b</i>
reference	Identifiant de la boutique Lyra Collect du prestataire	an..255	<i>87654321</i>
description	Nom de la boutique Lyra Collect du prestataire	an..255	<i>Boutique du prestataire</i>
bic	Code banque du compte bancaire du prestataire	an..11	<i>CMCIC1234</i>
iban	Numéro de compte du compte bancaire du prestataire	an..34	<i>FR1212341234123412341234</i>
is_marketplace_seller	ce champ permet d'identifier le seller de type market place. Dans la commande les items de type commission (is_commission = true) doivent être rattachés à ce seller. Ce seller est créé automatiquement lors de la création de la marketplace.	booléen	<i>true   false</i>
status	Statut de la ressource Marketplace <i>Généré par Lyra Collect</i>	an.10	<i>ACTIVE</i>
links	Liens vers les ressources Items du prestataire <i>Générés par Lyra Collect</i>	cf. exemple	<pre>{   "items": {     "href":     "https://secure.lyra.com/marketplace/sellers/1ea5e574-e198-55d4-ba23-f9405ec4226a/items"   }, }</pre>

## 19.9. Objet Shipping

---

Nom	Description	Format	Obligatoire	Exemple
delivery_company_na	nom du transporteur	ans..128		
shipping_speed	mode de livraison	enum		<i>STANDARD   EXPRESS   PRIORITY</i>
shipping_method	méthode de livraison	enum	X	<i>RECLAIM_IN_SHOP   RELAY_POINT   RECLAIM_IN_STATION   PACKAGE_DELIVERY_COMPANY   ETICKET</i>
shipping_delay	rapidité liée au mode de livraison lorsque shipping_speed = PRIORITY.	enum		<i>INFERIOR_EQUALS   SUPERIOR   IMMEDIATE   ALWAYS</i>
type	type de destinataire	enum		<i>PRIVATE   COMPANY</i>
legal_name	nom de la société	an..63		<i>Dupond &amp; Cie</i>
name	nom de la personne livrée			
first_name	prénom de l'acheteur	an..63		<i>Stéphanie</i>
last_name	nom de l'acheteur	an..63		<i>DUPOND</i>
phone_number	numéro de téléphone de l'acheteur	an..32		<i>0612324565</i>
address	adresse postale de l'acheteur	objet Address		

## 19.10. Objet Token

Nom	Description	Format	Obligatoire	Exemple
uuid	Identifiant unique de la commande <i>Généré par Lyra Collect</i>	an..36	N/A	<i>c730def9-e171-4f4f-92de-b7f09ed74b8d</i>
created_at	Date et heure de création de la ressource <i>Généré par Lyra Collect</i>	cf. exemple	N/A	<i>2018-11-27T13:53:48.461191Z</i>
updated_at	Date et heure de la dernière modification de la ressource <i>Généré par Lyra Collect</i>	cf. exemple	N/A	<i>2018-11-27T13:53:48.461191Z</i>
marketplace	UUID de la Marketplace	an..36	N/A	<i>2434c0a2-9d46-4e96-9553-1536c898625b</i>
buyer	Informations sur l'acheteur	Objet Buyer	X	
language	Langue d'affichage des pages de paiement (ISO 639-1)	a2		<i>FR</i>
url_return	URL de retour de l'acheteur sur le site de la Marketplace	an..512		<i>http://www.my-website.com</i>
status	Statut de la demande <i>Généré par Lyra Collect</i>	an..10	N/A	<i>CREATED</i>
alias	Identifiant du token dans la plateforme	ans..64		<i>1144951ea7ab42989c97159b3dfc0382</i>
alias_to_update	Identifiant du token à modifier dans la plateforme	ans..64		<i>1144951ea7ab42989c97159b3dfc0382</i>
payment_url	URL à donner au client pour la création du token	ans..255		

## 19.11. Objet Transaction

Nom	Description	Format	Obligatoire	Exemple
trans_uuid	Identifiant de la transaction	ans..32	X	<i>03934e6df4ec4f8ea0fe491e95fbc619</i>
order	Identifiant de la commande associée		X	<i>33ec0e5a-f25c-4bb1-8115-e54c70e30f16</i>
sequence_number	Indice de la transaction, notamment utile dans le cas des paiement en N fois.	n..12	X	1 pour la première transaction, 2 pour la seconde, etc.
paid_at	Date du paiement.	Date	Non	<i>2020-06-15T13:16:22Z</i>
auto_code	Code retour autorisation.	a..2	Non	<i>"0", "00", "51"</i>
payment_scheme	Description du moyen de paiement employé.	an..16	N/A	<i>CB</i>



Nom	Description	Format	Obligatoire	Exemple
<b>amount</b>	Montant de la transaction, exprimé dans la plus petite unité de la devise.	n..12	X	10450 pour 104,50 EUR
<b>initial_amount</b>	Montant de la transaction enregistré lors de la demande d'autorisation, exprimé dans la plus petite unité de la devise.	n..12	Non	10450 pour 104,50 EUR
<b>expected_capture_date</b>	Date de remise en banque souhaitée, dans le fuseau UTC, au format YYYYMMDDHHMMSS.	Date	X	20200615151622
<b>status</b>	Statut de la transaction. <i>Généré par Lyra Collect</i>	an..10	Non	TO_VALIDATE
<b>links</b>	Liens vers les ressources transactions. <i>Généré par Lyra Collect</i>	cf. exemple	Non	"links": { "transfers": { "href": "https://secure.lyra.com/marketplace/transactions/03934e6df4ec4f8ea0fe491e957bc619/transfers" } }
<b>created_at</b>	Date et heure de création de la ressource.	cf. exemple	Non	2020-06-15T13:16:49.223042Z
<b>updated_at</b>	Date et heure de la dernière modification de la ressource.	cf. exemple	Non	