

Amazon Pay - Intégration



Contenu

[Création d'un compte marchand AmazonPay](#)
[Comment proposer AmazonPay à vos clients ?](#)
 [En mode Shortcut](#)
 [Les web services en mode Web](#)
 [Rediriger l'acheteur après l'authentification sur Amazon Pay](#)
 [En mode Direct](#)
[Comment réaliser des tests ?](#)
[Les codes de retour](#)
[Les liens web](#)
[Pages associées](#)

Plus d'information

- [Détail de la création de compte Amazon Pay et Payline](#)
- [Amazon Pay - Intégration](#)

Création d'un compte marchand AmazonPay

Pour mettre en place AmazonPay sur Payline, vous devez créer un compte AmazonPay dans SellerCentral et récupérer les informations ID Marchand, Jeton d'autorisation NWS et Identifiant Client nécessaires pour configurer votre contrat Payline.

Le Card_Code de Monext Online est AMAZON_PAY

Informations

Statut

Type de moyen de paiement : **AMAZON_PAY**

Point de vente associé : **Demo Payline**

N° contrat *

Libellé

Devise *
Euro (978)

Description

Configuration spécifique au moyen de paiement

Type du bouton *
Pwa

Couleur du bouton *
LightGray

Taille du bouton *
medium

ID commerçant *

ID de l'application *

Token MWS *

Liste blanche des pays de livraison

Région *
DE

Pour créer un compte AmazonPay en homologation et production

Cette procédure est présentée à [titre indicatif](#), veuillez vous reporter à la documentation AmazonPay pour tout complément ou changement.

1. Lancer le script : OnBoardingMonext contenant l'id PSP Payline.
2. Suivre les étapes d'inscription et fournir les informations, coordonnées commerciales et bancaires puis récupérer l'**ID Marchand** le menu Intégration > Clé d'accès NWS.
3. Associer le compte marchand avec le compte PSP Payline : dans le lien sellercentral, sélectionner le menu [Paramètres > Autorisations utilisateur](#).
4. Autoriser un développeur avec le nom Monext et le l'identifiant : **1310-1336-9584**, puis récupérer le **Jeton d'autorisation MWS**. Voir l'identifiant de production.
5. Inscrire une nouvelle application à l'aide du menu de sellercentral, choisir le menu [Connectez-vous](#) avec Amazon et cliquer sur [Inscrivez une nouvelle application](#).
 - ⚠ Vous devez whitelister les URL d'appel à Amazon. Les URL seront statiques.
 - Pour le shortcut, vous devez indiquer l'URL d'appel depuis le bouton Amazon Pay ;
 - Pour le mode à redirection (page web de paiement V2 de Payline), vous devez supprimer le token de l'URL et whitelister URL avec le template dans l'interface Seller Central à chaque modification du template (voir l'encart information pour plus de détail).
6. Déployer l'encart Paramètres Web et récupérer l'**Identifiant client**.
7. Remplir impérativement les champs indiqués comme facultatifs. Les paramètres GET ne sont pas autorisés dans les URL de retour au panier :
 - Origines JavaScript autorisées: il s'agit des URLs des sites du commerçant ;
 - URL de renvoi autorisées: il s'agit des URL de retour au panier.



[Veillez suivre la procédure détaillée sur ce lien.](#)

Pour créer un moyen de paiement dans le centre administration Payline

Vous devez vous rapprocher de votre responsable de compte partenaire pour obtenir les informations nécessaires permettant de configurer votre compte de production sur Payline (contrat vad, ID marchand, Jeton d'autorisation MWS, Identifiant client).

Ensuite veuillez adresser un email à notre équipe support avec les informations ci dessous, afin qu'ils puissent procéder au paramétrage dans votre compte marchand Payline en indiquant l'ID Marchand et le nom de l'alias du contrat partenaire à créer. Vous devez réaliser des transactions 'pilote' pour valider le bon fonctionnement en production.

Contactez le [support Monext Online](#).

Ce numéro de contrat (alias) vous permettra d'effectuer les paiements.

Paramètre Payline	Description
ID commerçant	ID marchand récupéré précédemment sur sellercentral
ID de l'application	Identifiant client de l'application du marchand récupéré précédemment sur sellercentral
Token MWS	Jeton d'autorisation MWS associé au compte Monext récupéré précédemment sur sellercentral
Type de bouton	Aspect souhaité pour le bouton Amazon Pay
Couleur du bouton	Valeurs possibles : Gold (par défaut), LightGray, DarkGray
Taille du bouton	Valeurs possibles : small, medium (valeur par défaut), large, x-large
Liste blanche des pays de livraison	Cette liste est utilisée pour contrôler l'adresse de livraison. (vide) : Aucun contrôle sur l'adresse de livraison LOT 2 codes pays suivants normes ISO 3166 séparés par des "," (Ex.: "FR;IT;ES") PAYLINE contrôle l'adresse de livraison. Si elle est dans un pays hors liste blanche alors un message d'erreur demande à l'acheteur de changer son adresse de livraison.
Région	Il s'agit de la région Amazon de laquelle dépend le compte marchand • UK Grande Bretagne LOT 2 • DE Europe (valeur par défaut) • US Etats Unis LOT 2 • JP Japon LOT 2



Consignes AmazonPay sur les boutons

<https://pay.amazon.com/fr/merchant/tools>

Type du bouton

LwA	blocked URL
Login	blocked URL
PwA (valeur par défaut)	blocked URL
Pay	blocked URL
A	blocked URL

Taille du bouton

<i>small</i>
<i>medium (valeur par défaut)</i>
<i>large</i>
<i>x-large</i>

Couleur du bouton

Gold (valeur par défaut)	blocked URL	
LightGray	blocked URL	
DarkGray	blocked URL	

Validez votre intégration

Cette étape obligatoire vous permet de contrôler que votre intégration est correcte.

- Obtenez une transaction de paiement accordée avec le numéro de contrat Amazon Payline.
- Consultez le centre d'administration commerçant et validez le bon enregistrement de vos transactions de paiement.

Comment proposer AmazonPay à vos clients ?

Vous pouvez afficher le bouton AmazonPay sur le panier, ou dans la liste des moyens de paiement.

En mode Shortcut

Avec l'[API Web Payment](#) de Payline, vous pouvez intégrer le [mode Shortcut](#) :

Les étapes	Description
1. Le marchand initie le paiement	Appeler le service doWebPayment avec un <code>contractNumber</code> AmazonPay ;
2. Le marchand affiche le bouton AmazonPay sur le panier en mode Widget	Ajouter les scripts suivants dans le head de la page : <pre><script src="\$environment/scripts/widget-min.js" charset="utf-8"></script> <link rel="stylesheet" href="\$environment/styles/widget-min.css" charset="utf-8"></pre> Ajouter le code suivant dans le body de la page : <pre><div id="PaylineWidget" data-token="\$token" data-template="shortcut" data-event-didshowstate="widgetStateCallback" data-embeddedredirectionallowed="true"></pre> Si vous masquez les moyens de paiement du compte Amazon: <pre>.pl-amazon-wallet-container { display : none; }</pre> ⚠ Les URL de retour au panier ne doivent pas contenir de paramètres GET.

3. Le marchand affiche le carnet d'adresses et les moyens de paiement d'Amazon	Ajouter les scripts suivants dans le head de la page : <pre><script src="\$environment/scripts/widget-min.js" charset="utf-8"></script> <link rel="stylesheet" href="\$environment/styles/widget-min.css" charset="utf-8"></pre> Ajouter le code suivant dans le body de la page : <pre><div id="PaylineWidget" data-token="\$token" data-template="shortcut" data-event-didshowstate="widgetStateCallback" data-embeddedredirectionallowed="true"></pre> Si vous masquez les moyens de paiement du compte Amazon: <pre>.pl-amazon-wallet-container { display : none; }</pre>
4. Le marchand récupère les informations acheteurs et adresses	Utiliser la fonction : <code>Payline.Api.getBuyerShortcut();</code>
5. Le marchand doit récupérer les informations mises à jour si le consommateur change son adresse de livraison ou de facturation ou son moyen de paiement	Lorsque le consommateur change d'adresse, vous êtes prévenu via la callback widgetStateCallback et l'état <code>PAYMENT_TRANSITIONAL_SHORTCUT</code>. Il suffit alors de faire un nouvel appel à <code>Payline.Api.getBuyerShortcut()</code> A la validation du mode de livraison (clic sur le bouton Poursuivre) : mettre à jour les informations de livraison, de facturation et le montant de la commande <code>Payline.Api.updateWebpaymentData();</code>
6. Le marchand valide le paiement (clic sur le bouton payer)	Appeler la fonction <code>Payline.Api.finalizeShortcut();</code>
7. Le marchand doit contrôler le résultat du paiement	Utiliser le <code>getWebPaymentDetails</code> pour contrôler le montant, adresses et mode de livraison et les informations du consommateurs.

Les web services en mode Web

Le mode d'intégration est disponible avec l'API WebPayment : services [doWebPayment](#) et [getWebPaymentDetails](#).

- **Interfaces de paiement** : API WebPayment, Cente d'administration, TPEV, Batch.
- **Paiement à la commande** : en utilisant le service [doWebPayment](#) avec `payment.mode` à CPT et `code Action` à 101.
- **Paiement à l'expédition** : en utilisant les services [doWebPayment](#) et [doCapture](#) avec le `code Mode` à CPT et `code Action` à 100.
- **Demande d'annulation d'une autorisation non encaissée** : utilisez la fonction `doReset` permet au marchand de demander l'annulation totale ou partielle de la commande.
- **Demande de remboursement** d'une transaction remise : utilisez la fonction [doRefund](#) pour rembourser la commande.
- **Mode de paiement** : comptant, différé, récurrent, n fois avec le `code Mode` = CPT.
- **Module fraude** : règles sur la transaction, l'acheteur et sur le moyen de paiement.
- **3DSV2** : Suivre l'intégration des pages web : [3DSV2 avec Interface PageWeb](#).

Rediriger l'acheteur après l'authentification sur Amazon Pay

L'acheteur revient sur la page où se trouvait le widget au départ.

Pour rediriger l'acheteur sur la page de votre choix après son authentification chez Amazon, vous devez alimenter le paramètre 'data-partnerreturnurl' en [mode Widget](#), en ajoutant cet attribut dans la balise `<div>` avec le token session.

Vous pouvez également récupérer le token qui se trouve dans le hash # qu'a placé amazonPay dans l'url :

- Soit le token est présent dans l'url de manière classique [mon-marchand.com/checkout?paylinetoken=XXXXX](#)
- Soit en le récupérant dans le hash # AmazonPay avec ce bout de javascript :

Exemple de la fonction à reprendre..

```
function _getURLParameterFromHash(name, source) {
    return decodeURIComponent((new RegExp('[?]&|#]' + name + '=' +
        '([^\&]+?)(&|#|;|$)').exec(source) || [, ""])[1].replace(/\+/g,
        '%20')) || null;
}

var state = _getURLParameterFromHash('state', location.hash);
if (state != null) {
    var split = state.split('&');
    if (split != null) {
        for (var i = 0; i < split.length; i++) {
            const param = split[i];
```

```

        if (param.indexOf('token') !== -1 || param.indexOf('payline-token') !== -1) {
            var splittedParam = param.split("=");
            urlToken = splittedParam[1];
        }
    }
    if (urlToken) {
        element.setAttribute('data-token', urlToken);
    }
}
}
}

```

Récupérer l'adresse de livraison

⚠ Il est indispensable pour le commerçant de récupérer l'adresse de livraison via les fonctions de l'API widget Payline pour assurer l'envoi à la bonne adresse et afin d'obtenir la garantie de paiement Amazon Pay : voir le processus de paiement [developer.amazon.com.fr](https://developer.amazon.com/fr).

Mode Widget

- surcharger le bouton Payer en créant un bouton Continuer : `InitShortcut()` ;
- récupérer l'adresse via la fonction `getBuyerShortcut()` ;
- proposer de sélectionner le mode de livraison en fonction de l'adresse ;
- mettre à jour l'adresse utilisée avec `updateWebpaymentData()` et `finalizeShortcut()` puis proposer le bouton Payer ;
- afficher le résultat du paiement.

Mode Shortcut

- sur le bouton Payer prévoir un callback widget Payline ;
- le commerçant récupère l'adresse de livraison via les fonction du widget Amazon pour déterminer les modes de livraison disponibles ;
- sur le bouton Payer géré par le commerçant, mettre à jour le montant en fonction du mode de livraison ;
- La callback passée déclenchera le clique sur le bouton caché Continuer
- exécuter la fonction `iniShortcut(data-shortcutfinalizeauto = TRUE)` ;
- afficher le résultat du paiement.

En mode Direct

Avec l'API [Direct Payment](#) de Payline, vous pouvez intégrer le [mode direct](#) :

Les fonctionnalités de paiement disponibles :

- Le paiement à la commande en utilisant les services [doAuthorization](#) et/ou [doCapture](#) ;
- La fonction [getWebPaymentDetail](#) renvoie l'échéancier sélectionné par l'acheteur.
- La demande de remboursement partiel et multiple d'une transaction remise avec le service [doRefund](#).

Les spécificités AmazonPay :

- Le nom et prénom sont regroupés dans la balise `buyer.lastname` ;
- Le service [doCapture](#) et [doRefund](#) répondent un code retour `ON_HOLD_PARNER`. Une notification sera envoyée vers le marchand dès la mise à jour ;
- Adresse de livraison :
 - Une adresse de livraison est rattachée à un ou plusieurs moyens de paiement. Un moyen de paiement est rattaché à une adresse de facturation.
 - Pour les cas où le commerçant souhaite garder son adresse de livraison au lieu de l'adresse de livraison du consommateur d'Amazon, il suffit de réaliser un appel de la fonction **`Payline.Api.updateWebPaymentData()`** avant l'appel à **`Payline.Api.finalizeShortcut()`**.
Par exemple si l'adresse de livraison n'est pas l'adresse domicile du consommateur : point relais, adresse du magasin...
- Référence commande :
 - De la même manière avec la fonction **`Payline.Api.updateWebPaymentData`** vous pouvez mettre à jour la référence commande à la fin du parcours si la référence commande n'est pas définie en début de parcours.
- Authentification forte :
 - Vous devez récupérer les informations en post paiement selon la procédure décrite par Amazon Pay : Consulter la procédure AmazonPay : <https://developer.amazon.com/fr/docs/eu/amazon-pay-onetime/sca-upgrade.html>

Comment réaliser des tests ?

Toutes les types de cartes acceptées dans le wallet AmazonPay seront acceptés pour un paiement, [consulter la documentation d'AmazonPay pour tout complément information](#).

Note : Il est indispensable de réaliser des tests transactionnels sur l'environnement de production avec des comptes Amazon après la validation de la mise en œuvre sur l'environnement de test.

Les codes de retour

Monext Online vous informe du résultat d'un paiement via le ShortMessage, selon le mode d'intégration proposé par le moyen de paiement.

- pour l'[API WebPayment](#) avec le service [getWebPaymentDetails](#),
- pour l'[API DirectPayment](#) en réponse du service [doAuthorization](#) de manière synchrone ou [getTransactionDetails](#) en réponse asynchrone.

Les états retournés :

- Le paiement est accepté avec l'état ACCEPTED et le code retour 00000.
- Le paiement est refusé avec l'état REFUSED. Le code varie en fonction du motif de refus (Par exemple : 01xxx pour une raison bancaire ou 04xxx pour une suspicion de fraude).

La gestion des états et des codes retour sont listés [ici](#).

Les codes les plus utilisés sont :

Code Payline	ShortMessage	LongMessage
00000	ACCEPTED	Operation Successfull
02000	ONHOLD_PARTNER	Transaction in progress, please wait for payment status
02008	CANCELLED	Transaction canceled by user => redirection sur abort ou timeout url
02016	ONHOLD_PARTNER	Transaction hold on partner, please wait for payment method return
02008	CANCELLED	Transaction canceled by user
02020	REFUSED	Transaction refused by partner
02101	ERROR	System internal error (frontend)
02102	ERROR	Acquirer server communication error
02324	REFUSED	The session expired before the consumer has finished the transaction
02106	ERROR	Payment partner error
02012	ERROR	Unmatched partner return code
02305	ERROR	Invalid field format
02308	ERROR	Invalid value for
02310	ERROR	Requested function not available

Les liens web

Voici les liens utiles :

- Garantie Amazon de A-à-Z pour les marchands : <https://pay.amazon.com/fr/help/201212410>
- Garantie Amazon de A-à-Z pour les acheteurs : <https://pay.amazon.com/fr/help/201751470>
- Opposition FAQ : <https://pay.amazon.com/fr/help/201749690>
- Chargeback best practices : <https://pay.amazon.com/fr/help/201749670>

Pages associées

- [Amazon Pay](#)
- [Amazon Pay - Intégration](#)
- [Google Pay - Intégration](#)
- [Google Pay™ payment method](#)

- [Lydia](#)
- [MB Way](#)
- [Paylib payment method](#)
- [PayPal](#)
- [PayPal - Configuration](#)
- [PayPal - Intégration](#)