

INTÉGRATION PAYTECH

Par

Moussa Ndour

moussa.ndour@intech.sn

+221 77 245 71 99

21 janvier 2021

<https://paytech.sn/>

Ce document est un résumé de la documentation (<https://doc.paytech.sn/>), elle vous aidera à démarrer avec PayTech. Elle vous donne toutes les informations dont vous pourriez avoir besoin pour utiliser nos API.

<https://paytech.sn> : **BackOffice**

<https://sample.paytech.sn> : **Boutique de démo**

<https://doc.paytech.sn> : **Documentation**

1- Clés APIs

Pour utiliser l'API de PayTech API, vous devez vous inscrire sur la plateforme.

Pour créer un compte PayTech rendez à l'adresse <https://paytech.sn/> et cliquer sur le bouton « Espace Client » du menu, la page d'inscription va s'ouvrir, renseigner les informations du formulaire et confirmer la creation de compte.

Ensuite connectez vous dans votre espace client à l'adresse <https://paytech.sn/app>.

Une fois dans le Dashbord, cliquer sur le menu Paramètres, puis sur API pour récupérer vos clés.

Vous avez aussi la possibilité de les régénérer.



The screenshot shows a web interface titled "API CONFIGURATION". At the top left, there is a "API Keys" section with a refresh icon. At the top right, there is a "Save" button with a checkmark. Below this, there are two input fields: "API KEY" and "SECRET KEY". Each field has a dotted line indicating a long string of characters. To the right of each field, there are two icons: a green eye icon (toggle visibility) and a blue copy icon.

Toutes les requêtes envoyées par le site marchand vers notre API doivent comporter obligatoirement la clé API et la clé secrète dans les headers de la requête :

```
# you can pass the correct header for each request in this way
curl "api_endpoint_here"
-H "API_KEY: 1afac858d4fa5ec74e3e3734c3829793eb6bd5f4602c84ac4a506936"
-H "API_SECRET: 96bc36c11560f2151c4b43eee310cefabc2e9e9000f7e315c3ca3"
```

2- Demande de paiement

Le Paiement avec Redirection vous permet de rediriger votre client vers notre plateforme afin qu'il puisse achever le processus de paiement.

La demande de paiement définit une expérience utilisateur cohérente entre les méthodes de paiement, les systèmes de paiement, les plateformes et les commerçants.

Le marchand doit tout d'abord inclure le SDK Paytech dans la page de paiement du marchand comme suit:

```
<head>
  <meta charset="UTF-8">
  <link rel="stylesheet" href="https://paytech.sn/cdn/paytech.min.css">
  <script src="https://paytech.sn/cdn/paytech.min.js"></script>
</head>
```

Le site du marchand devra ensuite définir une URL pour la demande de paiement. Le SDK javascript Paytech se chargera d'appeler cette URL en lui passant les informations de paiement (prix, id_client...). L'URL devra alors attaquer l'API de Paytech pour récupérer le token de Paiement.

Un exemple d'URL de demande de paiement:

```
https://mydomaine.sn/requestPayment
```

Une fois le SDK inclus dans la page, le marchand pourra déclencher la demande de paiement par exemple lors d'un événement onclick:

```
<button onclick="makePayment(this)" data-id-transaction="UYH8HAB54SG">Payer cette transaction</button>
```

Et voici le corps de la fonction « makePayment »:

```

<script>
function makePayment(btn) {
    var idTransaction = pQuery(btn).attr('data-id-transaction');
    (new PayTech({
        idTransaction      : idTransaction,
    })).withOption({
        requestTokenUrl    : 'https://mydomaine.sn/requestPayment',
        method             : 'POST',
        headers            : {},
        presentationMode   : PayTech.OPEN_IN_POPUP,
        didReceiveError: function (error) {
            console.log(error);
        },
        didReceiveNonSuccessResponse: function (jsonResponse) {
            console.log(jsonResponse);
        }
    }).send();
    // .send params are optional
}
</script>

```

Le champs « presentationMode » peut prendre les valeurs:

- PayTech.OPEN_IN_POPUP (par default)
- PayTech.OPEN_IN_NEW_TAB

- PayTech.OPEN_IN_SAME_TAB

Maintenant, le marchand doit a définir la ressource qui va handle le paiement pour l'URL « <https://mydomaine.sn/requestPayment> »:

```
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import java.io.*;
```

```
@WebServlet("/requestPayment")
public class PaymentController extends HttpServlet {
    protected void doPost(HttpServletRequest req, HttpServletResponse
response) throws ServletException, IOException {
        String idTransaction = req.getParameter("idTransaction");
        int amount = 5000;
        String paymentTitle = String.format("Paiment facture N° %s de %s
FCFA", idTransaction, amount);
        String jsonResult = PayTech.sendPaymentRequest(idTransaction, amount,
paymentTitle);
```

```
        PrintWriter out = response.getWriter();
        out.print(jsonResult);
        out.flush();
    }
}
```

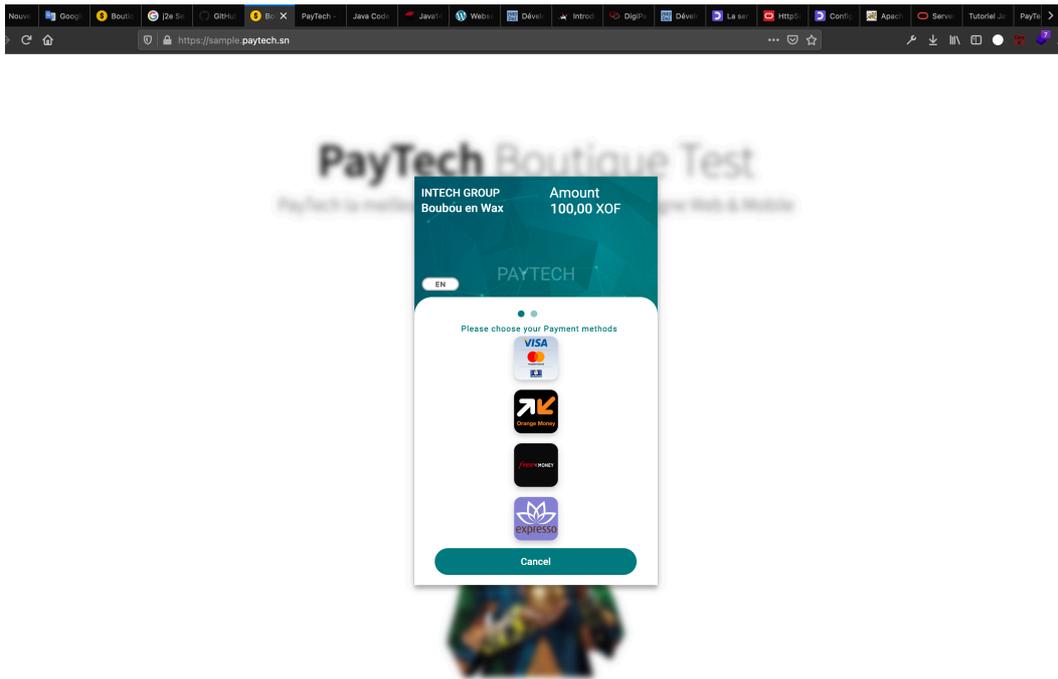
(La classe « PayTech » est défini dans le fichier PayTech.java (<https://doc.paytech.sn/PayTech.zip>) et contient les appels POST vers notre APIs).

Si votre backend ne trouve pas sous Java, vous pouvez consulter la documentation ou lire contenu du fichier PayTech.java pour voir les champs envoyé en POST vers l'APIs de PayTech.

Le marchand doit éditer le fichier PayTech.java pour mettre sa configuration notamment ses 6 champs:

```
private static final String API_KEY = API_KEY_HERE";
private static final String API_SECRET_HERE = API_SECRET_HERE_HERE";
private static final String PAYMENT_IPN_URL = « https://mydomaine.sn/
paymentIpn";
private static final String PAYMENT_SUCCESS_REDIRECT_URL = https://
mydomaine.sn/paymentSuccess";
private static final String PAYMENT_CANCELED_REDIRECT_URL = « https://
mydomaine.sn/paymentCanceled";
private static final boolean IS_PRODUCTION_MODE = true;
```

- API_KEY est la clé api générer plus haut
- API_SECRET_HERE est la clé secret généré plus haut
- PAYMENT_SUCCESS_REDIRECT_URL est la page vers le quelle l'étudiant sera redirigé une fois le paiement terminé
- PAYMENT_CANCELED_REDIRECT_URL est la page vers le quelle l'étudiant sera redirigé s'il annule le paiement (uniquement quand presentationMode = PayTech.OPEN_IN_NEW_TAB ou PayTech.OPEN_IN_SAME_TAB)
- PAYMENT_IPN_URL est l'URL qui sera appelé par PayTech pour signalé que le paiement a bien été validé
- IS_PRODUCTION_MODE si la valeur est false, le client sera débité que de 100 FCFA lors d'un paiement. Si elle est a true c'est le montant réel qui sera débité au client



Page de paiement PayTech

Maintenant quand l'utilisateur clique sur le bouton « Payer » cette URL est appelée par le SDK JS de PayTech, et si tous les paramètres ont été bien renseigné la page de paiement va s'ouvrir:

3- Instant Payment Notification (IPN) URL

Lorsqu'un client valide un paiement, PayTech envoie une notification à votre serveur à l'URL IPN que vous avez indiquée (PAYMENT_IPN_URL dans PayTech.java).

Cette notification comporte toutes les informations pertinentes de paiement de votre client que vous avez transmises lors de la demande de token de paiement.

À la réception d'une notification, votre serveur devra vérifier si la requête provient bien de PayTech en comparant le hash en sha256 de sa clé API et clé secrète avec les deux clés que nous lui avons envoyées dans la requête.

Voici la liste des paramètre qui sera posté vers votre URL IPN:

Paramètres	Description
client_phone	Le numéro de téléphone du client
payment_method	La méthode de paiement ['Carte Bancaire', 'Orange Money', 'Tigo Cash', 'Emoney']
item_name	Nom du produit
item_price	Prix de la commande
ref_command	La référence de commande générée par le site marchand.
command_name	Description de la commande
currency	Le code de la devise du paiement [XOF, EUR, USD....]
env	Environnement ['test', 'prod']
custom_field	Donnée additionnelle envoyée aux serveurs de PayTech lors de la demande de token
token	Le token de paiement
api_key_sha256	La clé API du site marchand hachée avec l'algorithme sha256
api_secret_sha256	La clé secrète du site marchand hachée avec l'algorithme sha256

Le marchand doit définir le controller pour l'URL « <https://mydomaine.sn/paymentipn> »:

```
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import java.io.PrintWriter;
```

```
@WebServlet("/paymentIpn")
public class IpnController extends HttpServlet {
    protected void doPost(HttpServletRequest req, HttpServletResponse
response) throws ServletException, IOException {
        PrintWriter out = response.getWriter();
        try {
            if(PayTech.ipnFromPayTech(req)){
                String idTransaction = req.getParameter("custom_field");
                String amount = req.getParameter("item_price");
                String paymentMethod =
req.getParameter("payment_method");
                //todo here update transaction status to solded
                out.print("IPN OK");
            }
            else{
                out.print("IPN KO NOT FROM PAYTECH");
            }
        } catch (Exception e) {
            e.printStackTrace();
            out.print(e.getMessage());
        }
        out.flush();
    }
}
```

Une fois l'IPN Reçu le marchand doit vérifier qu'il provient bien de PayTech, et alors il peut mettre à jour le status de la transaction a « soldé ».