

# **DOKUMENTACJA**

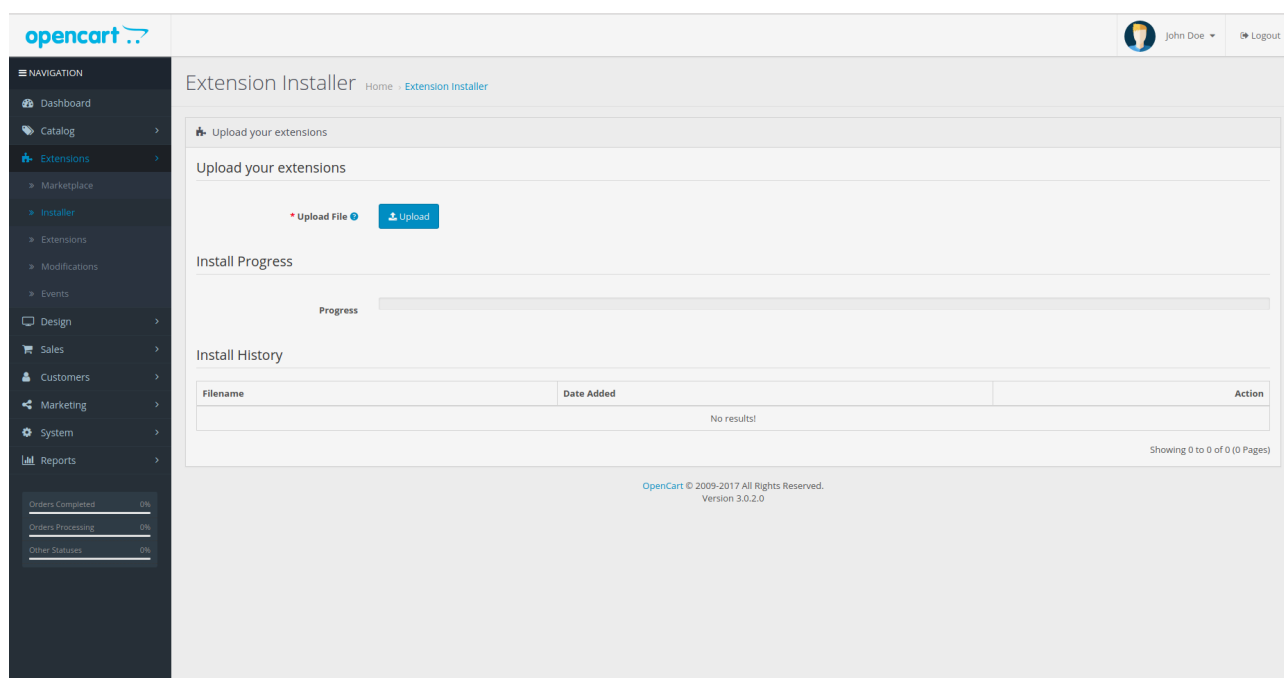
**OpenCart 3.X.X**

# INSTALACJA

## Jak zainstalować plugin OpenCart w wersji 3.X.X?

Instalacja pluginu OpenCart jest bardzo prosta. Opis instalacji krok po kroku:

1. Popieramy plik o nazwie „Homepay OpenCart.zip” ze strony [www.homepay.com](http://www.homepay.com) znajdziemy go po zalogowaniu do serwisu, w zakładce Homepay → Moduły.
2. Po pobraniu pliku rozpakowujemy archiwum zip - znajdziemy w nim plik o nazwie *homepay.ocmod.zip*.
3. Najprostsza metoda instalacji pluginu odbywa się po zalogowaniu jako admin do swojego sklepu. *Znalezieniu i wejściu w zakładkę w panelu głównym „Extensions” następnie w szukamy zakładki Installer po kliknięciu pojawi się nam przycisk o nazwie „Upload” klikamy go.*



4. *Następnym krokiem do ukończenia instalacji jest znalezienie i wybranie naszego pliku homepay.ocmod.zip i załadowaniu go.*

5. Jeśli moduł prawidłowo się zainstaluje ukaże nam się komunikat.

opencart

NAVIGATION

- Dashboard
- Catalog
- Extensions
  - Marketplace
  - Installer
  - Extensions
  - Modifications
  - Events
- Design
- Sales
- Customers
- Marketing
- System
- Reports

Orders Completed 0%  
Orders Processing 0%  
Other Statues 0%

Extension Installer

HomeExtension Installer

Upload your extensions

Upload your extensions

Upload File

Upload

Install Progress

Progress

Success: You have modified extensions!

Install History

Filename	Date Added	Action
homepay.ocmod.zip	16/11/2017	

Showing 1 to 1 of 1 (1 Pages)

OpenCart © 2009-2017 All Rights Reserved.  
Version 3.0.2.0

6. W następnym kroku przechodzimy do zakładki *Extension* → *Extension* w formularzu wyboru *Choose the extension type* wybieramy *Payments*.

opencart

NAVIGATION

- Dashboard
- Catalog
- Extensions
  - Marketplace
  - Installer
  - Extensions
  - Modifications
  - Events
- Design
- Sales
- Customers
- Marketing
- System
- Reports

Orders Completed 0%  
Orders Processing 0%  
Other Statues 0%

Extensions

HomeExtensions

Extension List

Choose the extension type

Payments (52)

Analytics (1)  
Captchas (2)  
Dashboard (8)  
Feeds (3)  
Anti-Fraud (3)  
Menu (0)  
Modules (26)  
Payments (52)  
Reports (13)  
Shipping (13)  
Themes (1)  
Order Totals (1)

Accept payments from around the world  
Offer Express Checkout Shortcut, letting buyers checkout directly from your basket page  
Improve conversion with PayPal One Touch and In-Context checkout

Connect with PayPal



















































page  
Fast configuration and setup  
Achieve PCI DSS v3.0 SAQ-A compliance using Hosted Fields  
Allow customers to vault card details

Connect with Braintree

Payments

Payment Method	Status	Sort Order	Action
Alipay Pay	Disabled		
Alipay Cross-border	Disabled		
Amazon Pay	Disabled		
Authorize.Net (AIM)	Disabled		

7. Wyszukujemy rozszerzenia *Homepay* instalujemy poprzez kliknięcie zielonego przycisku *Install*.

Cheque / Money Order		Disabled			 
Cash On Delivery		Enabled	5		 
Divido		Disabled			 
eWAY Payment		Disabled			 
First Data EMEA Connect (3DSecure enabled)		Disabled			 
First Data EMEA Web Service API		Disabled			 
Free Checkout		Enabled	1		 
G2APay		Disabled			 
Globalpay Redirect		Disabled			 
Globalpay Remote		Disabled			 
Homepay		Disabled			 
Klarna Account		Disabled			 
Klarna Checkout		Disabled			 
Klarna Invoice		Disabled			 
Lay-Buy		Disabled			 
LIQPAY		Disabled			 
NOCHEX		Disabled			 
Paymate		Disabled			 

# KONFIGURACJA

Konfiguracja jest równie prosta jak instalacja. Po zainstalowaniu modułu możemy kliknąć przycisk o nazwie „*Edit*”, jeśli posiadamy już aktywne konto z szybkimi przelewami w serwisie *Homepay*. To po prostu uzupełniamy wszystkie trzy pola wymaganymi danymi oraz zmieniamy status z *Disabled* na **Enabled** i klikamy w prawym górnym rogu przycisk „*save*”. Jeśli podane dane są prawidłowe moduł już powinien działać. ;)

The screenshot shows the OpenCart admin dashboard with the 'Homepay' extension configuration page. The left sidebar contains a navigation menu with options like Dashboard, Catalog, Extensions, Marketplace, Installer, Modifications, Events, Design, Sales, Customers, Marketing, System, and Reports. The main content area is titled 'Homepay' and includes a breadcrumb trail 'Home > Extensions > Homepay'. Below this is a form titled 'Edit Homepay' with several fields: 'Info' (Plugin version: 1.0.0, Register Now), a status dropdown (currently 'Disabled'), 'Total' (Total), 'Geo Zone' (All Zones), and three required fields: 'User ID', 'Public key', and 'Private key'. At the bottom, there are status bars for 'Orders Completed', 'Orders Processing', and 'Order Statuses', all showing 0%. The footer indicates 'OpenCart © 2009-2017 All Rights Reserved. Version 3.0.2.0'.

## PLUGIN OPENCART

### Problemy związane z napisaniem plugin-u do OpenCart:

- nie pełna dokumentacja tzn. wyjaśnione tylko podstawowe zagadnienia potrzebne do napisania plugin-u
- słabo opisane poszczególne funkcje, które potrzebne są do napisania plugin-u
- należy pamiętać o czyszczeniu cache za każdym razem gdy zmieniamy coś w skrypcie wpływającym na wyświetlanie się informacji.

### Czego potrzebujemy do napisania działającego plugin-u:

Do napisania plugin-u będziemy potrzebować odpowiedniej podstawowej struktury plików. Mamy tu do czynienia z MVCL czyli Model View Controller Language. Struktura jest dość prosta oraz intuicyjna. Na początek tworzymy folder o nazwie „upload” w tym folderze umieszczamy dwa kolejne katalogi o nazwie „admin” oraz „catalog”, w których to kolejno będziemy tworzyć strukturę MVCL. W każdym folderze musimy umieścić kolejne podkatalogi o nazwach: controller, view, model, language. W katalogach *controller* oraz *model* musimy utworzyć foldery i plik PHP w następującej postaci *extension/payment/homepay.php*. W folderze *view* tworzymy dwa foldery o

nazwach *image* oraz *template* w katalogu *template* tworzymy analogicznie jak poprzednio katalogi oraz plik *extension/payment/homepay.twig*, a w folderze *image* tworzymy katalog *payment*, w którym to umieszczamy logo homepay o wymiarach 77x35 px oraz nazwie *homepay.png*. W folderze o nazwie *admin* umieszczamy skrypty w odpowiednich katalogach odpowiadające za konfigurację, zapisanie oraz wyświetlenie naszego pluginu w oknie dla administratora dla sklepu opartego o OpenCart. W folderze *catalog* tworzy analogiczna strukturę folderów do tej z folderu *admin*. W tym katalogu umieszczamy całą logikę oraz front połączenia się z serwisem homepay.

## STRUKTURA SKRYPTU

### **Nazwy z opisem funkcji użytych w folderze admin/controller i pliku „homepay.php”:**

*public function index* – funkcja wczytuje język oraz odpowiada za logikę pól formularza konfiguracyjnego.

*protected function validate* – funkcja odpowiada za walidację pól formularza.

*public function install* – funkcja instalująca.

*public function uninstall* – funkcja odinstalowująca.

### **Nazwy z opisem funkcji użytych w admin/model i pliku „homepay.php”:**

*public function createDatabaseTables* – funkcja tworzy bazę z podstawowymi danymi

*public function dropDatabaseTables* – funkcja usuwa bazę.

### **Skrypt w admin/view i pliku „homepay.twig”:**

*Skrypt widoku formularza konfiguracyjnego.*

### **Nazwy z opisem funkcji użytych w folderze catalog/controller i pliku „homepay.php”:**

*public function loadConfig* – skrypty z dokumentacji homepay’a dostosowany do OpenCart.

*public function index* – wymagana funkcja przekazuje parametry z formularza zapłaty do controllera.

### **Nazwy z opisem funkcji użytych w folderze catalog/controller i pliku „homepay\_notify.php”:**

*Skrypt dostępny w dokumentacji homepay’a.*

### **Skrypt w catalog/view i pliku „homepay.twig”:**

*Skrypt widoku formularza zapłaty.*

### **Nazwy z opisem funkcji użytych w folderze catalog/model i pliku „homepay.php”:**

*public function getMethod* – funkcja pobiera podstawowe dane z bazy danych.