


# Index

- [Create merchant account.](#)
- [Import Paymentwall plugins.](#)
- [Integrate payment widget.](#)
- [Integrate Brick \( built-in UI \).](#)
- [Integrate Brick \( code only \).](#)

## Create merchant account


If you do not have a merchant account registered at Paymentwall, please sign up by completing the steps below.

- If you don't have a Paymentwall Merchant Account yet, please [Sign up](#).
- Fill out the Sign Up Form




### Personal

- Get instant access to your payment history
- Receive the latest updates on your favorite services
- Easily move to a business account to start accepting payments



### Business

- Instantly start accepting payments online in 200+ countries
- Earn money by being a Paymentwall merchant, advertiser or affiliate
- Gain access to fraud management, analytics and advanced reporting tools



## PW Account

Show

[SIGN UP](#)

[Already have an account? Sign In](#)

[Secure Area](#)

[Terms of Service](#) | [Privacy Policy](#)

- Please enter your Company Name and website URL

# Set up your Paymentwall account to buy or sell online!



## Personal

- Get instant access to your payment history
- Receive the latest updates on your favorite services
- Easily move to a business account to start accepting payments



## Business

- Instantly start accepting payments online in 200+ countries
- Earn money by being a Paymentwall merchant, advertiser or affiliate
- Gain access to fraud management, analytics and advanced reporting tools



## PW Account

Only one more step to register for your PW Account

CONTINUE


Secure Area

[Terms of Service](#) | [Privacy Policy](#)

- Press the Continue button. Your Paymentwall Merchant Account has been created

If you already have a Merchant Account registered at Paymentwall.

- After you login, please go to **My Projects** tab. You will see your first project already created. Please note the Project Key and Secret Key. You will need them later to finish the project setup.

 **Paymentwall**

PERSONAL **BUSINESS**

Feed

Analytics >

Payouts

**My Projects**

Marketing tools >

Research and Refund

Test App

SUBMIT FOR REVIEW

Web

Project Key 614fdc7e3f87108595ae0459d029e822

Secret Key ..... Show

Settings Payment Systems Products Widgets

BRICK

MOBIAMO


PW LOCAL

SPIDERPIPE

MINT

+ ADD N

- Press the **Settings** button. Please select "Digital Goods" (or other API) under "Your API".
- Set the **Pingback URL** to: [http://\[your-domain\].com/paymentwall\\_pingback](http://[your-domain].com/paymentwall_pingback)
- Set the **Signature Version** to 3.
- Click **Save Changes** .


**Paymentwall**

PERSONAL
BUSINESS

Feed
Analytics >
Payouts
My Projects
Marketing tools >
Research and Refund

Settings | Regions | Offers | Payment systems | Products | Widgets

Your API

☒ Digital Goods ?
☐ Virtual Currency ?
☐ Cart ?
☐ DRM ?

Widget signature is required

☐ Off ?

Evaluation mode

☐ Off ?

Pingback type

☒ URL
☐ Email

Pingback URL

[Test pingback](#)

Please include "http://" in the URL

Pingback signature version

Custom Pingback parameters

[Add a custom parameter](#) | [Get help on available parameters](#)

Notify email

optional email to subscribe for daily statistics, which are sent at midnight CDT. You can set up it later at Project settings section.

CANCEL
SAVE CHANGES

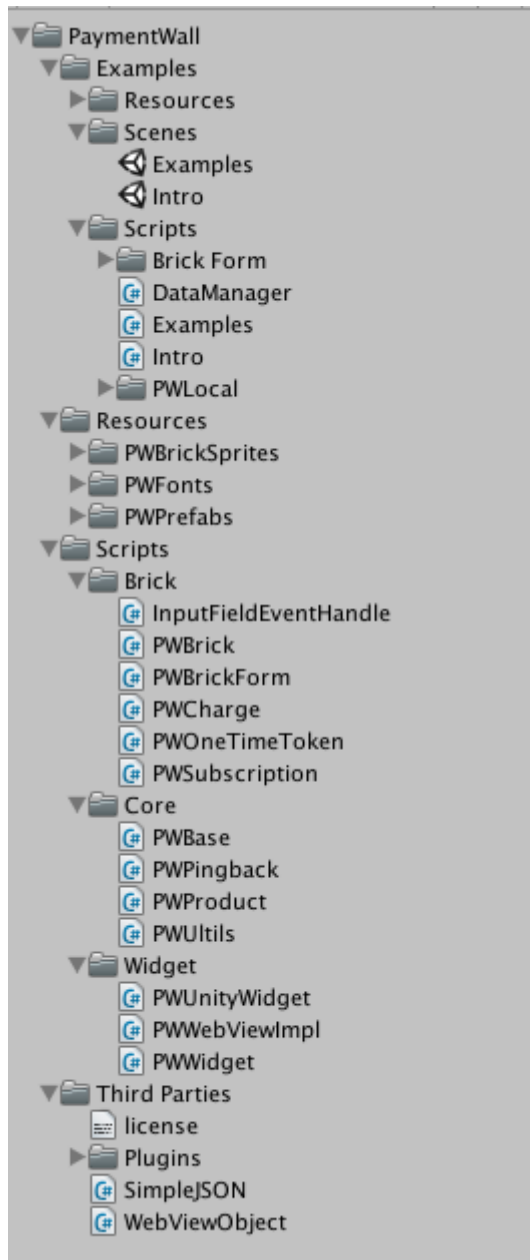
- Press the Widgets button, Add New Widget, scroll down and select Paymentwall Multi widget.

## Import Paymentwall plugins

- Go to [Assets Store](#) and install the Paymentwall plugin.
- Import all files from Paymentwall plugin.



- Folder structure of Paymentwall Plugin :
  - **\*\*Examples** : contains example scenes and example scripts.( Please check this scenes for details )
  - **\*\*Resources** : contains sprites and prefab of Paymentwall's SDK.
  - **\*\*Scripts** : contains base scripts of Paymentwall's SDK.
  - **\*\*Third Parties** : contains third-party code ( JSON parse, webview renderer )



## Integrate payment widget

- Create new Unity MonoBehaviour class, add following code to it. Modify API type and other custom parameters as you like.

```
using Paymentwall;
```

```
public class TestPlugin : MonoBehaviour {  
  
    // Use this for initialization  
    void Start () {  
        PWBase.SetApiType(Paymentwall_Base.API_VC);  
        PWBase.SetAppKey("9fa5245253f742e50e705f65b9b02d66"); // available in your  
Paymentwall merchant area  
        PWBase.SetSecretKey("3918c61450ab7aaa62760036f03eb947"); // available in  
your Paymentwall merchant area  
    }  
}
```

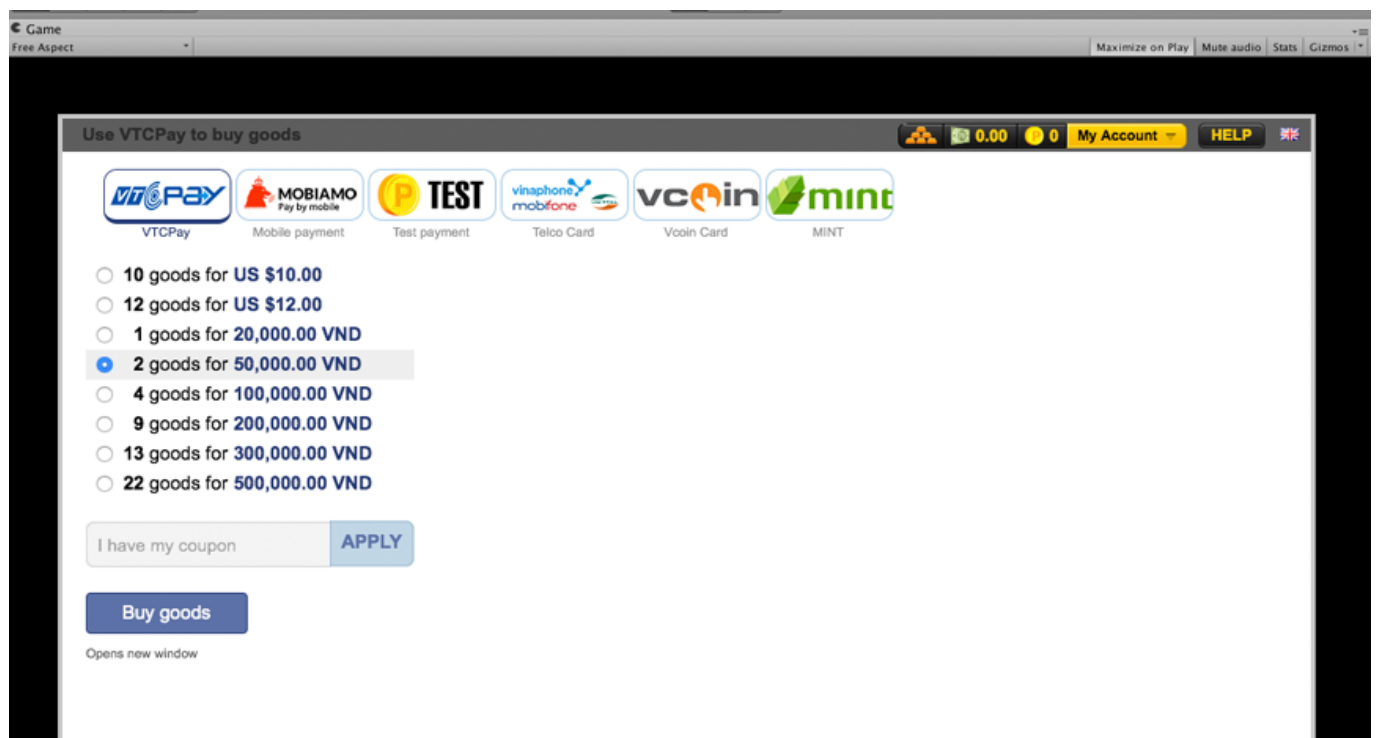
```

List<PWProduct> productList = new List<PWProduct>();
PWProduct product = new PWProduct(
    "product301", // id of the product in your system
    9.99f, // price
    "USD", // currency code
    "Gold Membership", // product name
    PWProduct.TYPE_SUBSCRIPTION, // this is a time-based product; for
one-time products, use Paymentwall_Product.TYPE_FIXED and omit the following 3
parameters
    1, // time duration
    PWProduct.PERIOD_TYPE_YEAR, // year
    true // recurring
);
productList.Add(product);
PWWidget widget = new PWWidget(
    "user40012", // id of the end-user who's making the payment
    "p1_1", // widget code, e.g. p1; can be picked inside of your merchant
account
    productList,
    new Dictionary<string, string>() {{"email", "user@hostname.com"}} //
additional parameters
);

PWUnityWidget unity = new PWUnityWidget (widget);
StartCoroutine (unity.callWidgetWebView (gameObject,canvas)); // call this
function to display widget
}
}

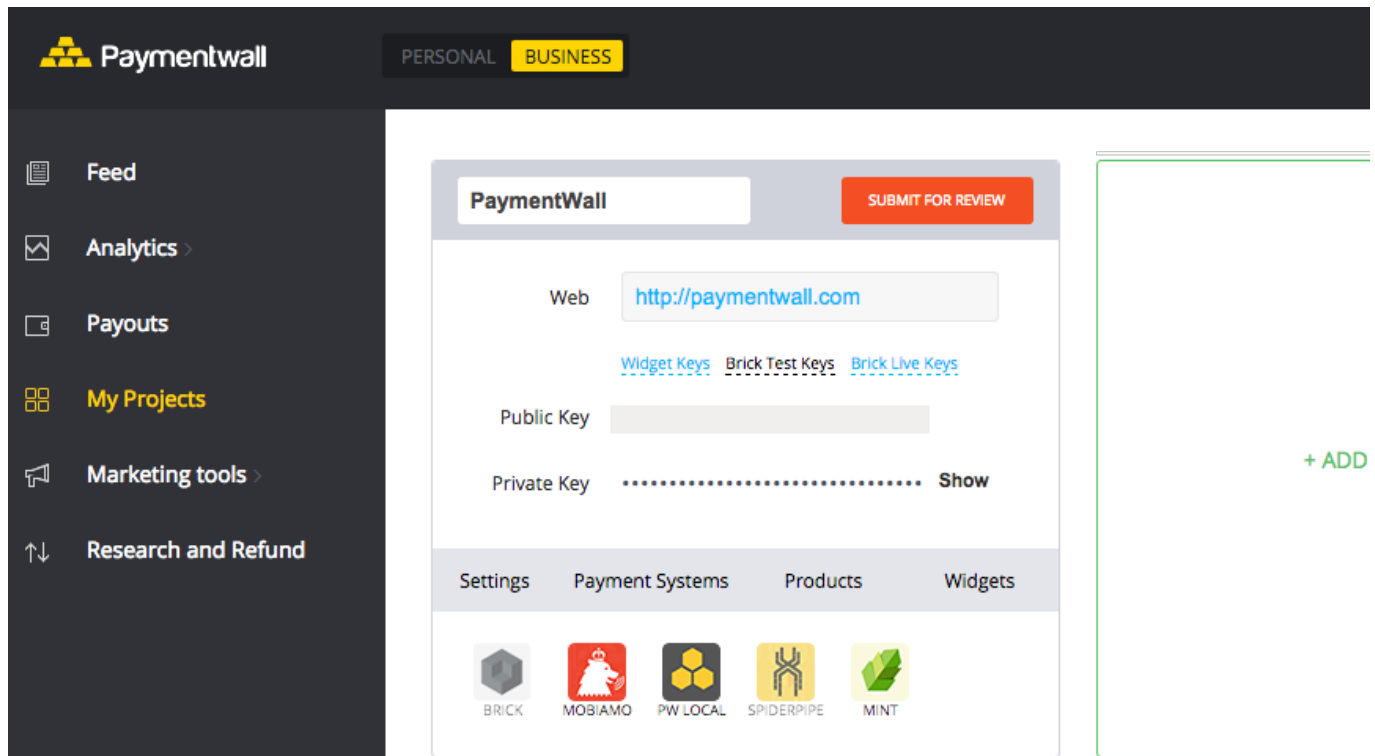
```

- You can modify size and behaviour of payment's widget in **PWUnityWidget.cs**
- Run Unity Editor ( only support on OSX ) or build it to mobile and see your results.



# Integrate Brick ( built-in UI )

- Grab your public key and private key inside your merchant account.



- Create new Unity MonoBehaviour class, add following namespace.

```
using Paymentwall;
```

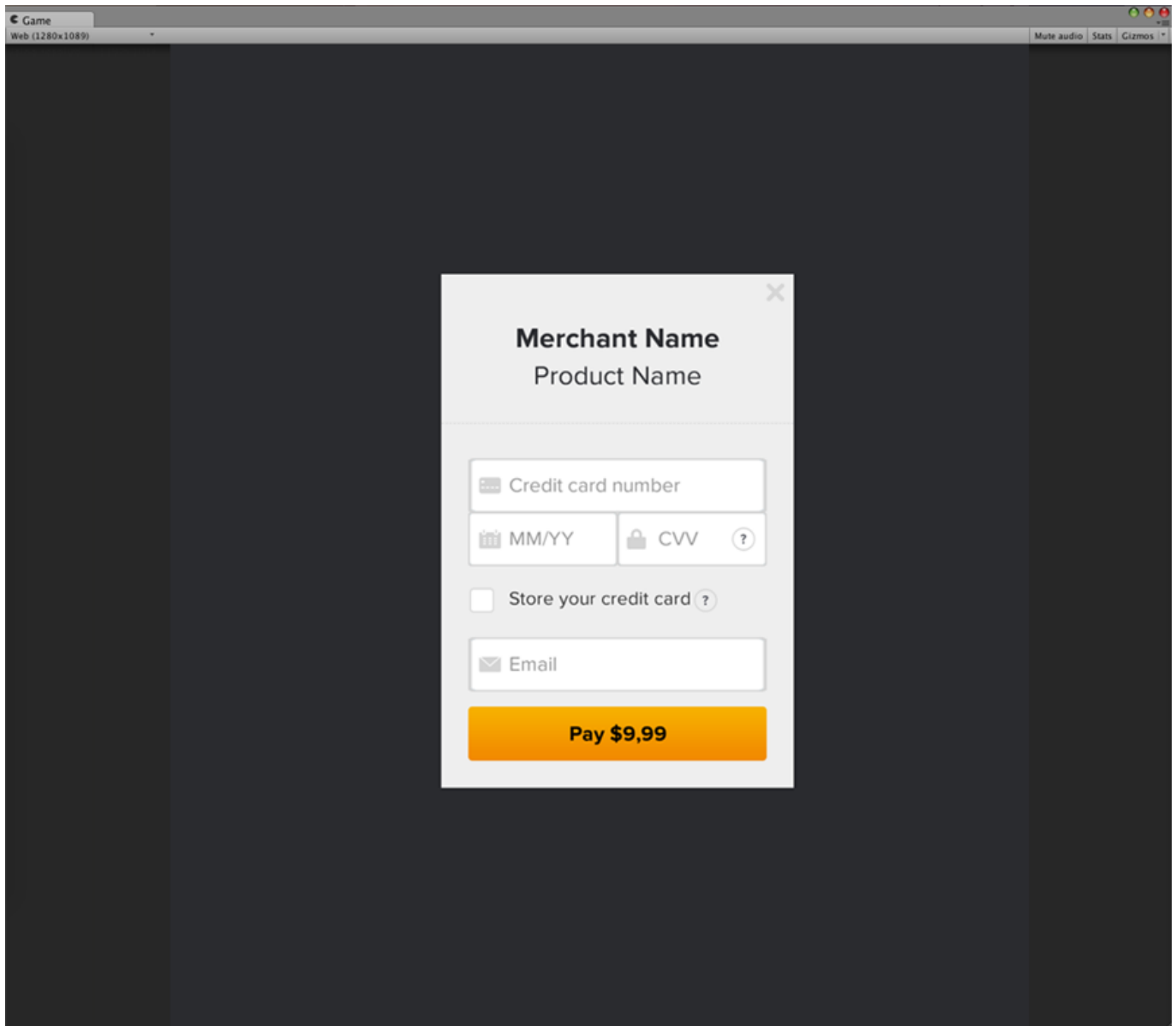
- Remember to switch SDK to Live Mode.

```
PWBase.SetApiMode(API_MODE);
```

- Add following code and modify parameters based on your demand.

```
PWBrick brick = new PWBrick (3.0f, "USD", "Your Awsome Game", "Small pack");  
PWBase.SetAppKey ("YOUR PUBLIC KEY");  
PWBase.SetSecretKey ("YOUR PRIVATE KEY");  
brick.ShowPaymentForm ();
```

- Result



## Integrate Brick ( code only )

- Or you can use your custom UI, and request token and charge credit card with this code.

```
PWBase.SetAppKey ("YOUR PUBLIC KEY");
PWBase.SetSecretKey ("YOUR PRIVATE KEY");

PWOneTimeToken tokenModel = new PWOneTimeToken ();
tokenModel.Create (PWBase.GetAppKey (),
    "4000000000000002", // card number
    "12", // expire month
    "16", // expire year
    "123"); // CVV number
PWCharge charge = gameObject.AddComponent<PWCharge> ();
yield return StartCoroutine(charge.create (tokenModel,
    "test@email.com", // Paymentwall will sent receipt to this
    address.
    "USD", // Currency code
    "9.99", // Amount of charge
    "fingerprint", // Fingerprint
```



```
"this is description")); // Description of payment
```

### ***Submit the project for approval***

Once all the settings have been properly configured, go back to your Paymentwall Merchant Area -> My Projects and submit the project for approval by pressing the Submit For Review button.

Your project will be approved shortly once we review the integration and make sure that everything is running smoothly for your users.

If you have any questions, please feel free to contact us at [devsupport@paymentwall.com](mailto:devsupport@paymentwall.com)