



PHP API Developer Guide

This guide illustrates all the steps needed to add WAC's in-app payment functionality to your PHP app.

Step 1: Run the Sample App

See how in-app payment would work in your app -- install the SDK and try out its sample application.

Step 2: Create Your WAC Account

If you do not already have a WAC account, start by creating one.

Step 3: Create Your API Keys

Define your application, planned operator markets, and pricing. Then we can provide your API keys.

Step 4: Manage Your App

Customize in-app icons and localize purchase item names for different markets / countries.

Step 5: Update Your App

Add the WAC in-app payments functionality to your app.

Step 6: Test Your App

See if it works, test your in-app payments using WAC's test environment.

Step 7: Certify Your App

Tell WAC how to pay you, answer the compliance questions from your operators, and have WAC handle your publishing credentials.

Step 8: Set Your API Keys Live

Activate your API keys for the markets where you want your in-app purchases to be real monetary transactions.

Step 9: Push App to Markets

Make your app available to customers with WAC's in-app payment service inside.

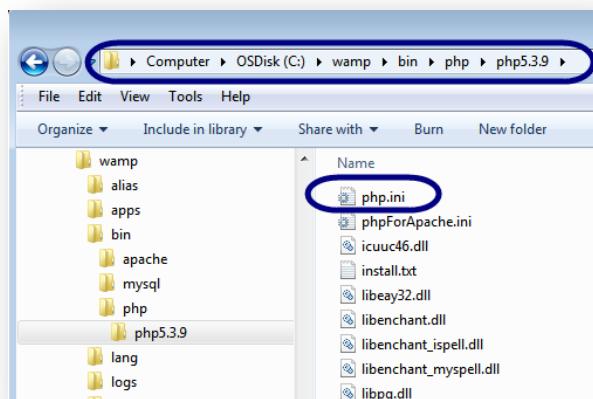
Step 1: Run the Sample App

WAC has included a sample app with its SDK so you can see WAC's in-app payments in action. And we recommend you start off by doing just that -- download WAC's SDK first off and run the sample app. Experience WAC in-app payments for yourself and experience what your customers will experience.

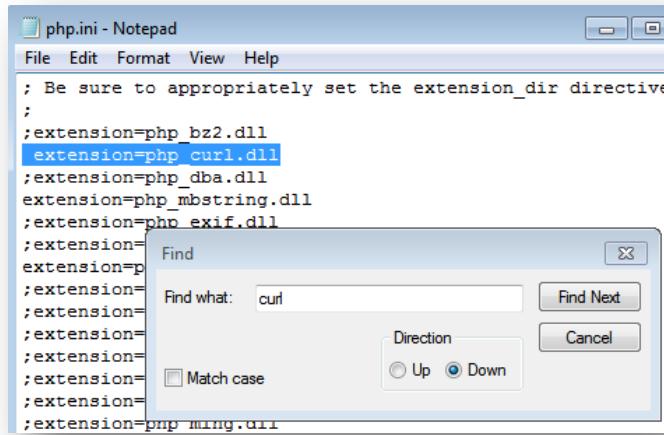
- A) Set Up Your Environment**
- B) Set Up WAC's Payment SDK**
- C) Run the Sample App**

A) Set Up Your Environment

1. You will need to set up the following on your system:
 - [WampServer 2c or greater](#) (*Windows only*)
 - PHP Version 5.1.2 or greater (*included when installing WampServer 2.2c on Windows*)
 - Apache 2.2 or greater (*included when installing WampServer 2.2c on Windows*)
 - The following enabled in php.ini: libcurl3 with cURL, SQLite, mcrypt
2. WampServer requires port 80, therefore ensure that no other process is using this port. (Note that both Skype and IIS (Internet Information Services) use port 80 by default, these must be halted or configured for a different port before running WampServer.)
3. WampServer users enable php.ini extensions as follows:
 - a. Ensure WampServer is shut down.
 - b. Open **WAMP\bin\php\<your php version>\php.ini**.



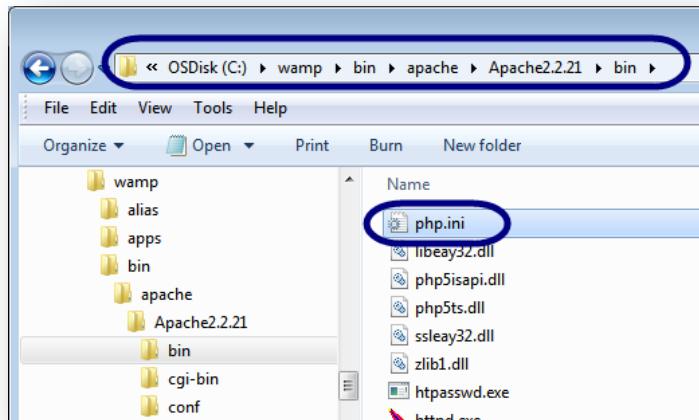
- c. Ensure that this line is not commented out: **extension=php_curl.dll**.



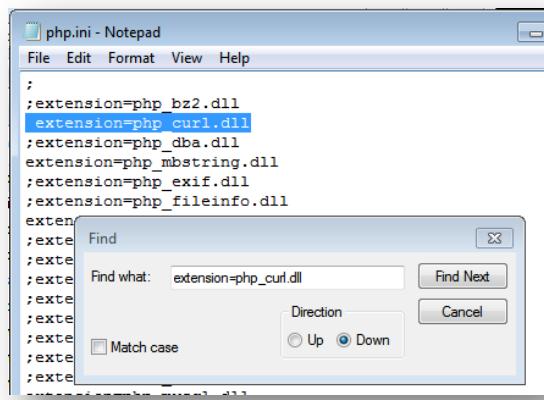
- d. Do the same for the following:

- **extension=php_sqlite.dll**
- **mcrypt.algorithms_dir=**
- **mcrypt.modes_dir=**
- If you are using a version of PHP earlier than 5.1.2, do the same for **extension=php_mhash.dll**

- e. Open **WAMP\bin\Apache\<your apache version>\php.ini**.



f. Here also, ensure **extension=php_curl.dll** is not commented out.



g. Do the same for the following:

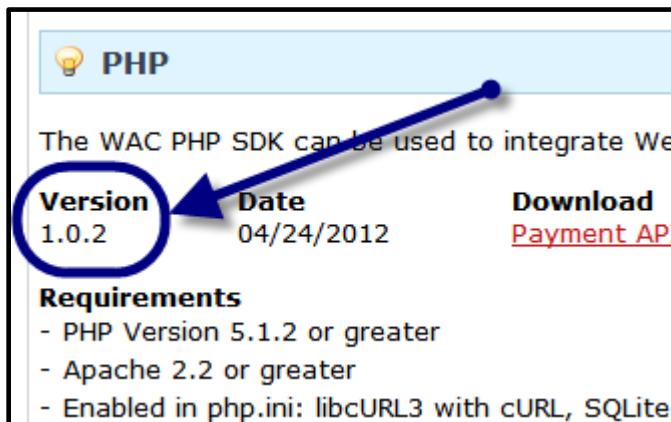
- **extension=php_sqlite.dll**
- **mcrypt.algorithms_dir=**
- **mcrypt.modes_dir=**
- If you are using a version of PHP earlier than 5.1.2, do the same for **extension=php_mhash.dll**

h. Restart WampServer.

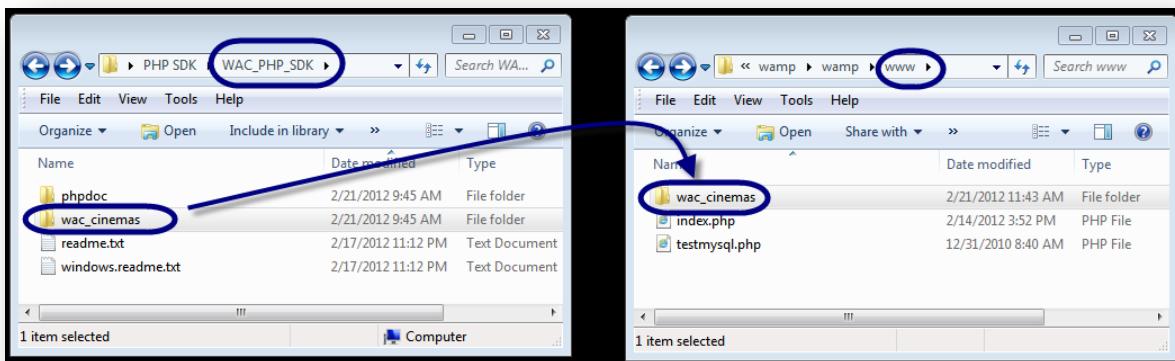
B) Set Up WAC's PHP SDK

- 1) Download and extract WAC's PHP SDK: <https://www.wacapps.net/sdks>.

Note: This guide's instructions are for **version 1.0.2 of WAC's PHP SDK** – make certain that this guide is correct for your version of the SDK. The most current developer guide is always available at <https://www.wacapps.net/developer-guide>.



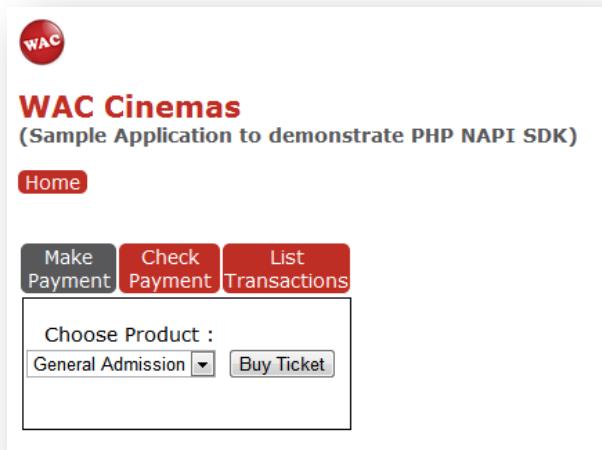
- 2) Copy the SDK folder **wac_cinemas** to the root directory (\www) of your Webserver.



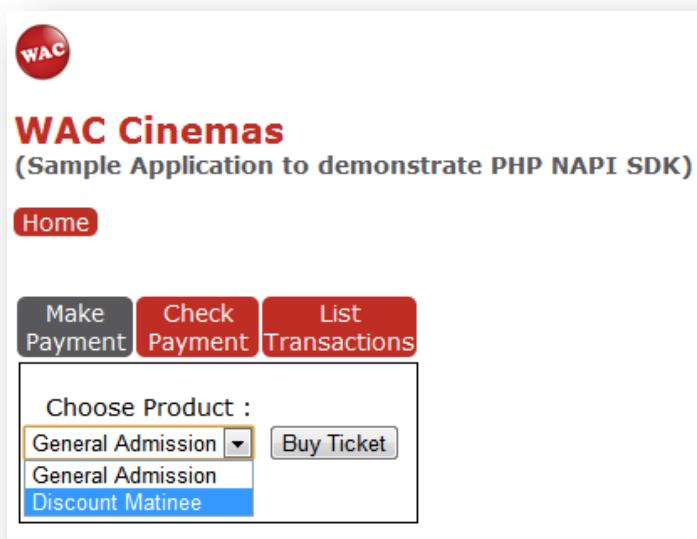
C) Run the Sample App

1. Ensure that the device you run the app on is set to your correct local time: If WAC's payment gateway receives calls from a system whose time varies more than 3 minutes from the app's local time, it will send back an error "Signature validation failed and/or not authorized."
2. Open http://localhost/wac_cinemas to open the sample app.

Note: The sample application redirect URL is provisioned as http://localhost/wac_cinemas/wac_billing_callback.php. As a result, you must use this URL (http://localhost/wac_cinemas) to run the sample application. This is specified in `wac_cinemas/app_constants.php`.

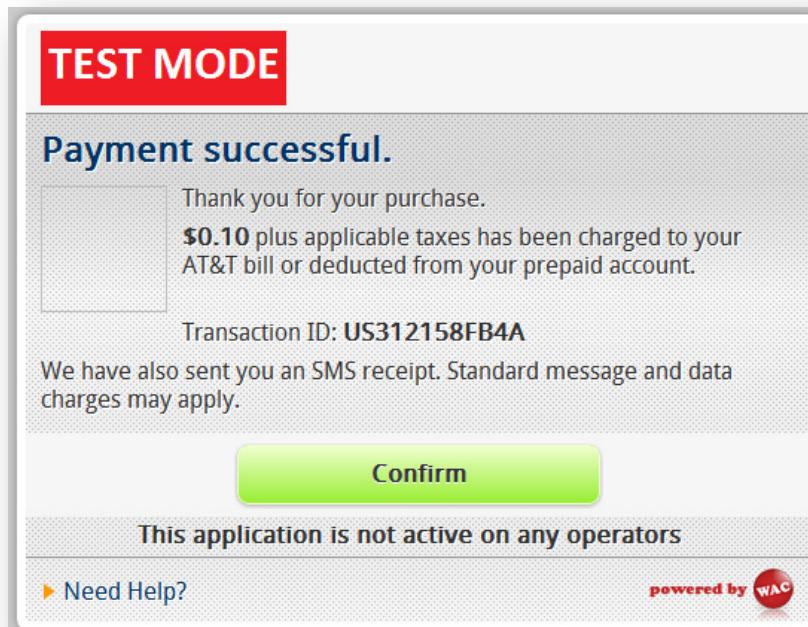


3. Try out the payment functionality by buying a mock movie ticket: Choose a ticket and click **Buy Ticket**.



4. The purchase screen opens.

- a. **Customers on their operator's signal** will be identified automatically and taken to the purchase confirmation screen. Clicking **Confirm** completes the purchases and the charge is made to the customer's mobile operator bill.



- b. **Customers not on their operator's signal** (such as when using WIFI instead), will be asked to submit their location and number for identification.

The screenshot shows a form titled 'We cannot detect your phone number'. It asks to select a country (United States) and enter a mobile phone number (+19724891737). A note says the number will not be passed on. By clicking 'Submit', a text message with a PIN code will be sent. Buttons for 'Submit' and 'Cancel' are at the bottom, along with a 'Need Help?' link and a 'powered by WAC' logo.

For your test of this sample app, WAC has set up these test numbers so that you can see each different purchase scenario. Use these while trying this sample app:

Scenario	What to Enter	Message
Successful payment	+19724891737	<i>Payment successful.</i>
Spend limit error	+19724891648	<i>You have reached your operator spend limit. Please contact your operator for help.</i>
Payment failed	+19724892325	<i>Payment failed, please try again later. You will not be charged for this transaction.</i>
Operator not supported	Any operator number not live on WAC	<i>Operator not supported.</i>

- c. After entering their phone number, in the real world the customer would be sent an SMS PIN so they could verify their identity by entering it on the next screen.

For your test of any number above with this sample app, enter PIN **4681** and click **Buy**.

TEST MODE

WAC Cinemas - PHP

You are about to buy a **Discount Matinee** from **WAC**. \$0.10 plus applicable taxes will be charged to your AT&T bill or deducted from your prepaid account.

A text message with a temporary pin has been sent to **+19724891737**. Please enter your PIN below.

Enter PIN:

By clicking **Buy** you accept our [Terms & Conditions](#).

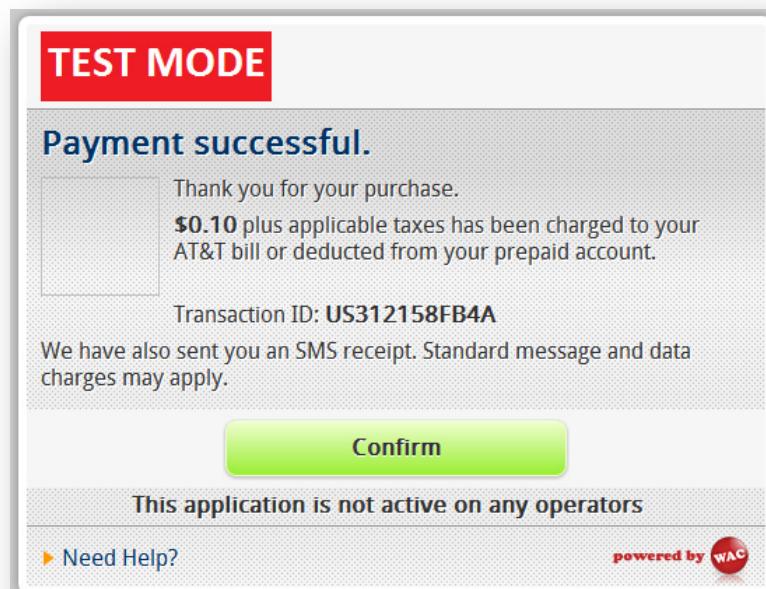
Buy **Cancel**

This application is not active on any operators

► Need Help? 

- d. The confirmation screen appears.

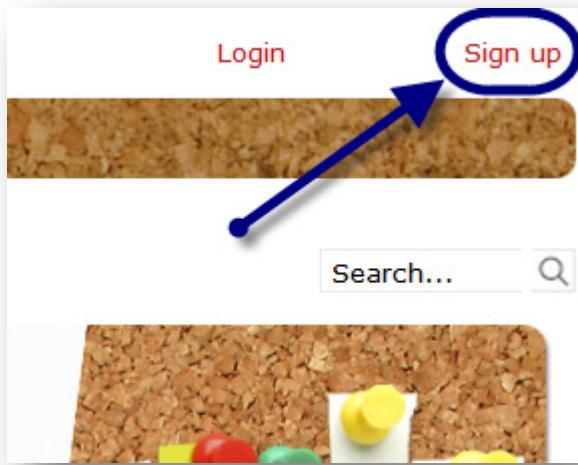
When the **Confirm** button is clicked, the customer's operator bill is charged and the purchase is completed.



Step 2: Create Your WAC Account

If you do not already have a WAC account, create one now to tell WAC who you are and how to deposit the proceeds from your in-app sales.

- 1) Go to <http://www.wacapps.net/> and click **Sign up**.



- 2) Complete the Sign up screen and click **Sign me up**.

The screenshot shows the 'Sign up' form. It includes fields for First Name (Walter), Last Name (Sobchak), Email Address (walter.s.sobchak@mailinator.com), Nickname (WalterS), Company Name (Sobchak Security), Password and Retype Password (both containing dots), and a Security Image field with the code 'E 2 6 T 6 4 K' and a 'Refresh image' button. Below the form is a note about agreeing to terms and conditions, a 'Sign me up!' button, and a 'View WAC Terms & Conditions' link.

Sign up

Please ensure all field's with an * are filled out.

*First Name

*Last Name

*Email Address

*Nickname

*Company Name

*Password

*Retype Password

Security Image

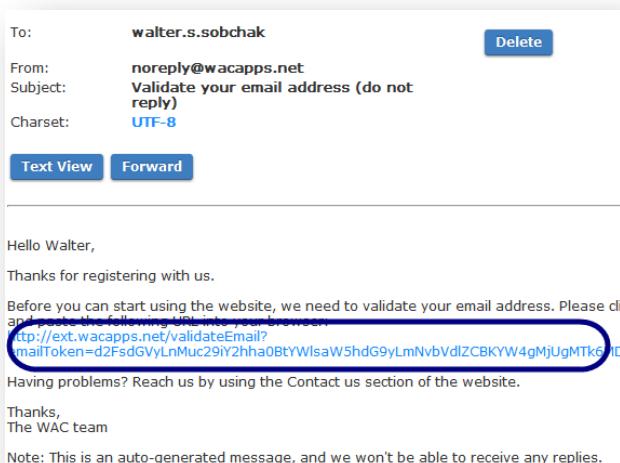
*Image Text

*By signing up you agree to be bound by the terms of this site.
View [WAC Terms & Conditions](#)

Field	Description
Email Address	We will use this email to send critical updates, be certain to use an account you check frequently.
Nickname	This will identify you when you post to forums and must be unique.
Company Name	Just enter your name if you do not represent a company.
Password	Must be composed of between 6 and 20 letters and numbers.

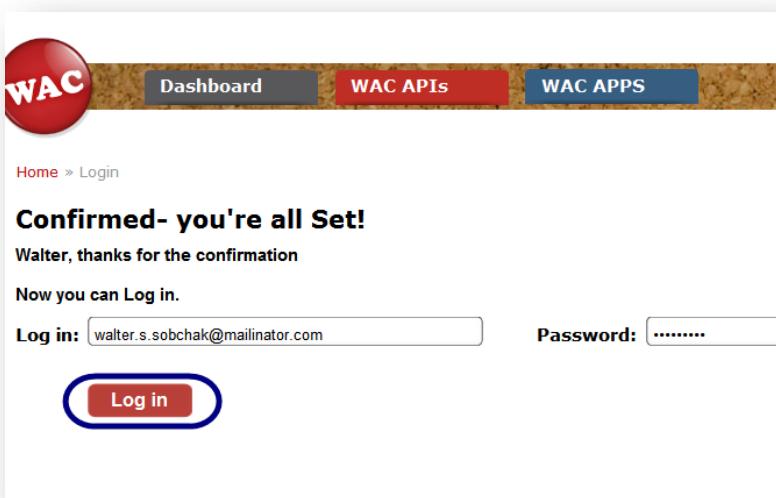
- 3) You should immediately receive an email asking you to validate your registration.

Click the validation link.

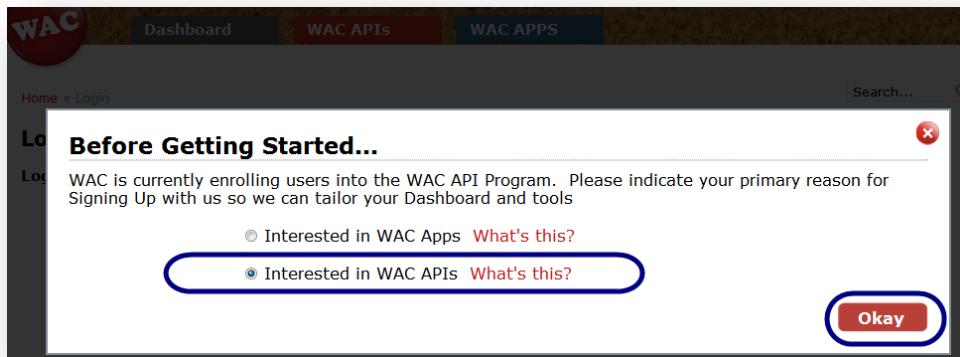


- 4) Your browser opens and you see your account creation is confirmed.

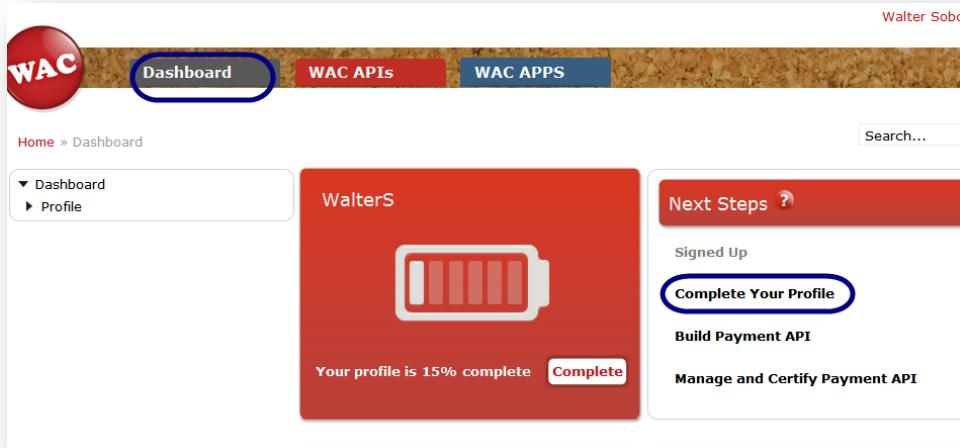
Log in with the credentials you just created.



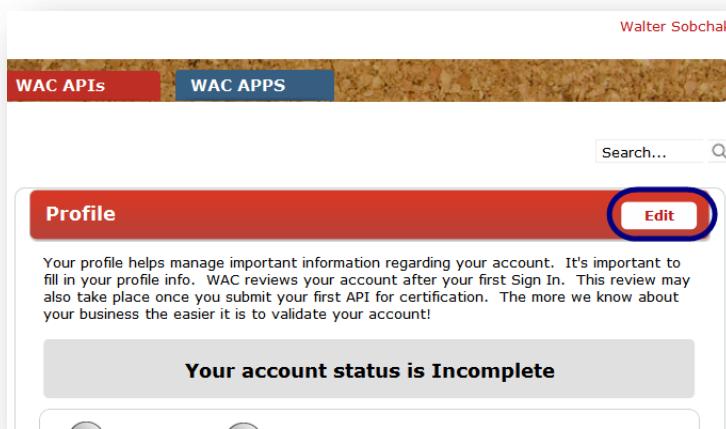
- 5) Choose the **WAC APIs** option and click **Okay**.



- 6) On the Dashboard screen, click **Complete Your Profile**.



- 7) On the Profile screen, click **Edit**.



- 8) Enter your information as shown and click **Save**.

Profile

Your profile helps manage important information regarding your account. It's important to fill in your profile info. WAC reviews your account after your first Sign In. This review may also take place once you submit your first API for certification. The more we know about your business the easier it is to validate your account!

Your account status is Incomplete

Basics

- Profile Status:** 15%
- First Name:** Walter
- Last Name:** Sobchak
- Nickname:** WalterSs
- Email:** walter.s.sobchaak@mailinator.com

Company Info

- Company Name:** Sobchak Security
- Company Address:**
 - 2556 East 3rd Street
 - Long Beach, CA 90804
 -
 -
 -
- Country:** USA

Support

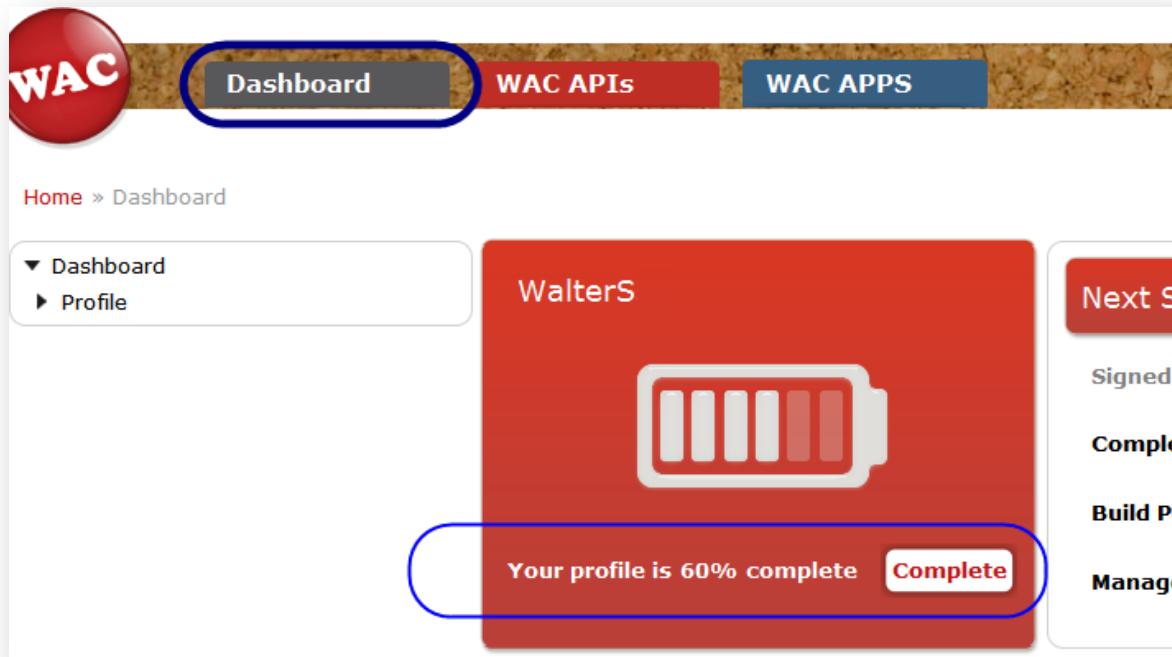
- Support Email:** walter.s.sobchaak@mailinator.com
- Support URL:** http://www.sobchaaksecurity.com
- Support Number:** +1 9252018501 Ext

Contact Info

- Main Contact:**
 - Walter Sobchak
 - Email: walter.s.sobchaak@mailinator.com
 - Number: +1 9252018501 Ext
- Marketing Contact:**
 - Walter Sobchak
 - Email: walter.s.sobchaak@mailinator.com
 - Number: +1 9252018501 Ext
- Technical Contact:**
 - Walter Sobchak
 - Email: walter.s.sobchaak@mailinator.com
 - Number: +1 9252018501 Ext

9) Click **Dashboard**.

Your profile is not yet complete because your banking details that tell WAC how to pay you for your app sales are not yet entered. You do not need to do this yet, leave your profile now and move on to [Step 3: Create Your API Keys](#).

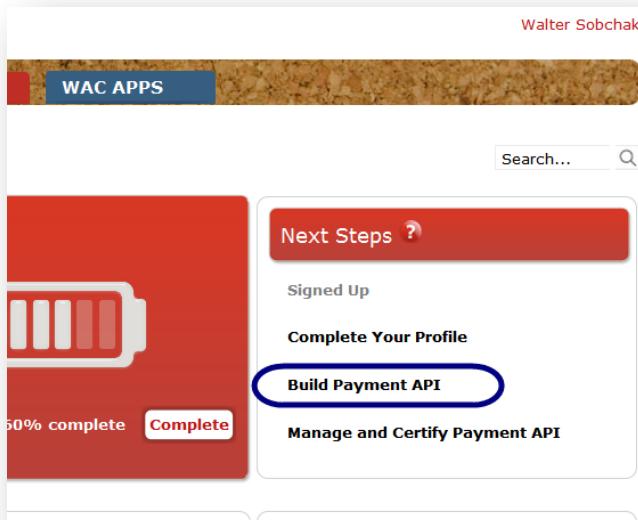


Step 3: Create Your API Keys

Now tell WAC what you will sell inside your app, the operator markets to target, and how much to charge in each market.

After you enter this information, WAC can provide you the API keys that your app can use to conduct in-app payments with WAC's servers.

- 1) Click **Build Payment API**.



- 2) Complete the **App Details** screen and click **Next**.

The screenshot shows the 'App Details' step of the WAC APPS setup process. The steps are: App Details, Create Products, Select Markets, Price Products, Review, and Create Keys. The 'App Details' step is highlighted with a red circle. The form asks for application details: Application Name (WACTrek), Version number (1.0), Application platform (Android 2.3), and Description (Space shooter game.). It also has a 'Web Based App?' field with radio buttons for 'Yes' and 'No' (set to 'No'). At the bottom are 'Cancel', 'Save & Exit', and 'Next' buttons, with 'Next' being circled in blue.

Field	Description
Application Name	Enter the name customers will see when they view their purchase history.
Version Number	Enter no more than five digits (numbers or “.” only) to identify this payment API to you and to customers viewing their purchase history. Note: You can use the same version number and application name for multiple in-app payment APIs. However, choose a version number that makes it easy for you and for customers viewing their purchase history to tell which item was purchased.
Application Platform	Indicate which platform the app runs on.
Description	This description (1) helps the WAC Compliance Team understand your app’s functionality and (2) helps differentiate between apps with the same name.
Web Based App?	If this is a Web-based app, choose Yes . Otherwise, choose No .
Use default WAC Redirect URI?	If you indicated this is a Web-based app above, this option appears: <ul style="list-style-type: none"> Choose Yes to enter your own redirect URI. Choose No to have WAC assign one to you at the end of this registration.
Enter your Redirect URI	If you elected to enter your own redirect URI, enter it here.

- 3) On the **Create Products** screen, you define each item customers can purchase in your app:

For each item customers can purchase in your app, enter a product item name (*25 characters max*) and click **Create**.

Create names for each of the items you want to charge for in your application and we'll assign Product Codes which will need to be incorporated into your App

Default I...	Product Item	Product Code	
	WACTrek - Level 5	wac-c1bb555f-f4fa-4672-a3ed-0f2ebd...	X
	WACTrek - Level 4	wac-c6764a9b-60a9-482c-b9b6-9313...	X

Page 1 of 1 Displaying 1 - 2 of 2

Back Save & Exit Next

- 4) When finished, click **Next**. (You can always return to edit and add additional items.)

Create names for each of the items you want to charge for in your application and we'll assign Product Codes which will need to be incorporated into your App

Default I...	Product Item	Product Code	
	WACTrek - Level 6	wac-a254773b-5b5e-4b44-a6b6-5086...	X
	WACTrek - Level 5	wac-c1bb555f-f4fa-4672-a3ed-0f2ebd...	X
	WACTrek - Level 4	wac-c6764a9b-60a9-482c-b9b6-9313...	X

Page 1 of 1 Displaying 1 - 3 of 3

Back Save & Exit Next

- 5) On the **Select Market screen**, mark the operator checkbox for each operator in each country with whom you will market your app.

View	Country	Actions
	Bulgaria	
	Germany	
	O2 DE Telekom DE	
	India	
	Korea (South)	
	Philippines	
	Spain	
	USA	

- 6) Click **Next** when finished.

View	Country	Actions
	Bulgaria	
	Germany	
	India	
	Korea (South)	
	Philippines	
	Spain	
	USA	

Back Save & Exit Next

- 7) On the **Price Products** screen , you are ready to enter prices for each in-app sale item you defined.

Note: You can edit these prices later, including after the app is pushed live for market purchase.

Click the **Set prices** button for the first item.

The screenshot shows a progress bar at the top with steps: App Details, Create Products, Select Markets, Price Products (highlighted in red), Review, and Create Keys. Below the bar, a message says: "Now that you've determined your Markets and Operators you can configure prices for each of your Product Items. Select the Set Prices button to use the pricing tool. You will have the chance to change these prices before you set your Payment API live". A table lists three products:

Product Item	Product Code	Price Sta...	Actions
WACTrek - Level 6	wac-3104d272-c4b4-4bed-9b4b...	No	Set prices
WACTrek - Level 5	wac-8c4c246b-788a-43bf-973e...	No	Set prices
WACTrek - Level 4	wac-2aefc50b-57ef-4fda-b1e4...	No	Set prices

At the bottom, there are navigation buttons: Back, Exit, and Next.

- 8) Enter the sales price of the in-app product.

The screenshot shows the 'Price Your Product Item' screen. It includes a note: "Create Your Product Item and set your price & we'll automatically suggest a price for the markets you are creating your Payment API for. If you want to customize click change. [How is this calculated?](#)".

Product Item: WACTrek - Level 6
Product Code: wac-3104d272-c4b4-4bed-9b4b-91f4a201da1b

1 Your Currency: USD - United States D^c 2 Your Price: .99 Suggest Price

Country	Operator	Pricing Type	Price	Actions
Korea (South)	Kt	Fixed	1100 KRW	change
Germany	Telekom DE	Fixed	0.75 EUR	change

3 Cancel Save & Exit 4 Okay

Step	Action
1	Enter your own currency and the item's desired price in this currency.
2	Click Suggest Price .
3	<p>WAC suggests the equivalent in the currency of each market you selected. WAC calculates this suggestion by (a) converting your base currency to the operator's currency using daily exchange rates and then (b) rounding it up to the nearest whole value.</p> <p>Note: If the Pricing Type field is Fixed, you can only select one of the available price points, which have been defined by the operator. If the Pricing Type field is Flexible, you may enter any price point up to the maximum transaction limit. All price points for all operators can be found in the Operator Pricing Guide at https://www.wacapps.net/pricing.</p> <p>To enter a price other than the suggested price, click the change link and you will see a drop down menu that shows all the available price points. Choose one of these price points.</p>  <p>Note: The mobile operator for each market determines whether these item prices include applicable value added or sales tax or other such taxes and how they are added to the customers total at checkout (see the WAC Operator Pricing Guide for all operator VAT/TAX models). In addition, local consumer laws may require disclosure of price variations. In the next step, this guide will provide an example of where and how to post this notice for customers.</p>
4	Click Okay .

9) You are returned to the **Price Products screen**.

Set the price for each remaining item and click **Next** when finished.

Now that you've determined your Markets and Operators you can configure prices for each of your Product Items. Select the Set Prices button to use the pricing tool. You will have the chance to change these prices before you set your Payment API live

Product Item	Product Code	Price Sta...	Actions
WACTrek - Level 6	wac-3104d272-c4b4-4bed-9b4b...	Yes	Change prices X
WACTrek - Level 5	wac-8c4c246b-788a-43bf-973e...	No	Set prices X
WACTrek - Level 4	wac-2aefc50b-57ef-4fda-b1e4-...	No	Set prices X

Page 1 of 1 Displaying 1 - 3 of 3

Back Exit Next

10) When finished setting prices, click **Next**.

Now that you've determined your Markets and Operators you can configure prices for each of your Product Items. Select the Set Prices button to use the pricing tool. You will have the chance to change these prices before you set your Payment API live

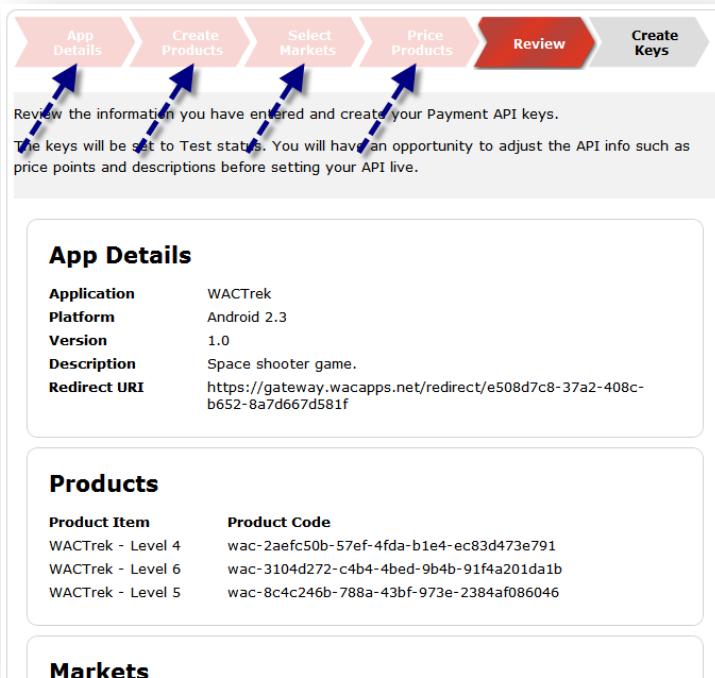
Product Item	Product Code	Price Sta...	Actions
WACTrek - Level 4	wac-2aefc50b-57ef-4fda-b1e4-...	Yes	Change prices X
WACTrek - Level 6	wac-3104d272-c4b4-4bed-9b4b...	Yes	Change prices X
WACTrek - Level 5	wac-8c4c246b-788a-43bf-973e...	Yes	Change prices X

Page 1 of 1 Displaying 1 - 3 of 3

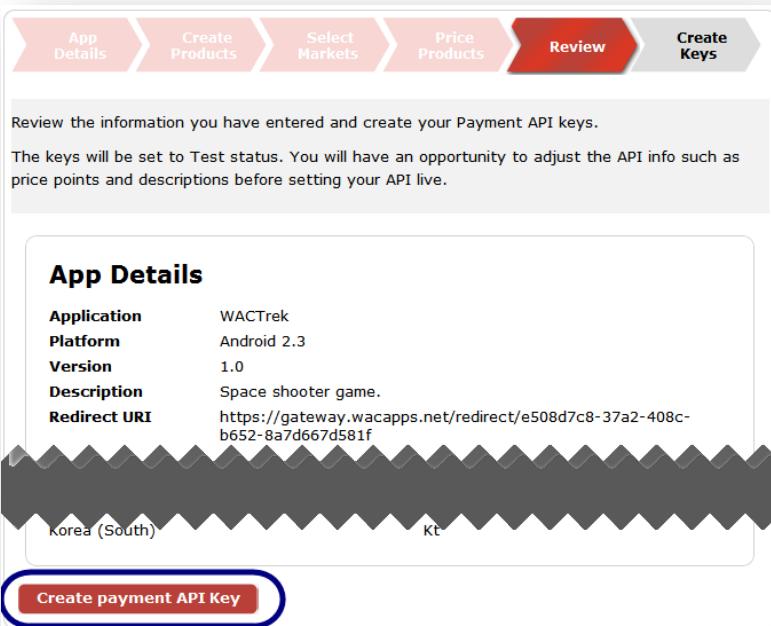
Back Exit **Next**

- 11) On the **Review screen**, verify (a) your app details are correct and (b) you have entered all products you currently know you want to sell in your app (you can return later to add more).

If you need to make edits, click to open and edit the relevant page.



- 12) When you are satisfied that your app details and the products in it are correct, click **Create Payment API Key**.



13) Your API keys are created and provided on the **Create Keys screen**.

Congratulations! You have created a Payment API. Your Payment API Key, Shared Secret Key and Application ID are needed for integration into your App. [Click here](#) for information or visit the [Tutorial](#)

Please note, these keys can only be used for this Payment API and can't be used within multiple platforms and versions of the same App. If you want to sell the same Product items in other App's, you can use our clone function with the [Manage Payment API tool](#)

Application ID	wac-1b084ae1-63fb-45d7-937f-928b3cb53b2e
Client ID	wac-b7e3ce1bf0babae328705b158e9632ad5da23d87
Shared Secret Key	f99c8c290b880b1c4cbc96dedba61c9cdb750860

After your keys are created, you can see them any time on the **App Details** screen:

This screenshot shows the 'Tools' section of the WAC APIs interface. At the top, there's a navigation bar with 'Dashboard', 'WAC APIs' (which is circled in blue and has a large orange number '1' over it), and 'WAC APPS'. Below the navigation is a breadcrumb trail: Home > WAC APIs > Payment API > Tools > Manage Payment APIs. A large red 'Tools' title is centered. On the left, a sidebar menu under 'WAC APIs / Payment API' includes 'Getting Started', 'Documentation', 'Resources', 'Tools' (which is circled in blue and has a large orange number '2' over it), 'Create Payment API Key' (circled in blue), 'Manage Payment APIs' (circled in blue), and 'Customer Management'. To the right is a table titled 'Application' with one entry: 'WACTrek' under 'Application' and 'Android 2.' under 'Platform'. A large orange number '3' is placed over the table.

This screenshot shows the 'App Details' screen for the application 'WACTrek'. The left sidebar lists 'AC APIs', 'Payment API' (with a large orange number '4' over it), 'Getting Started', 'Documentation', 'Resources', 'Tools', 'Create Payment API Key', 'Manage Payment APIs' (circled in blue), and 'Customer Management'. The main area is titled 'App Details' and contains a table with two rows: 'Application Name' (WACTrek) and 'Platform' (Android 2.3). A large orange number '5' is placed over the 'App Details' title.

Step 4: Manage App

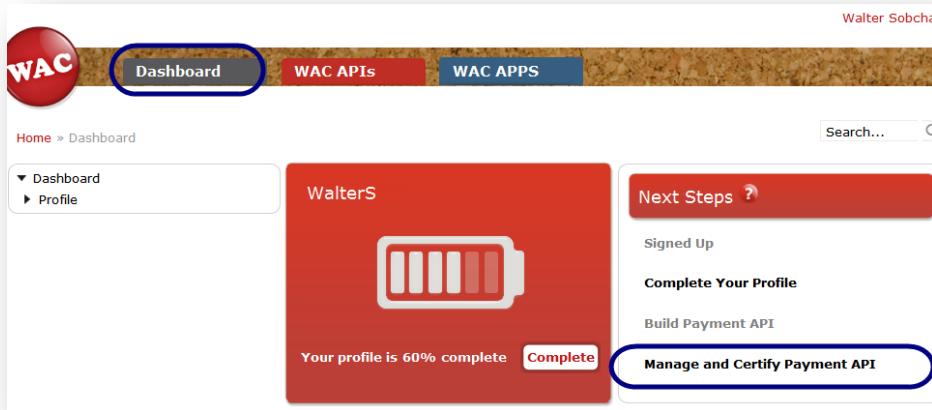
You are ready to add the product icons and descriptions your customers will select to purchase your app's products.

A) Add Icons & Descriptions

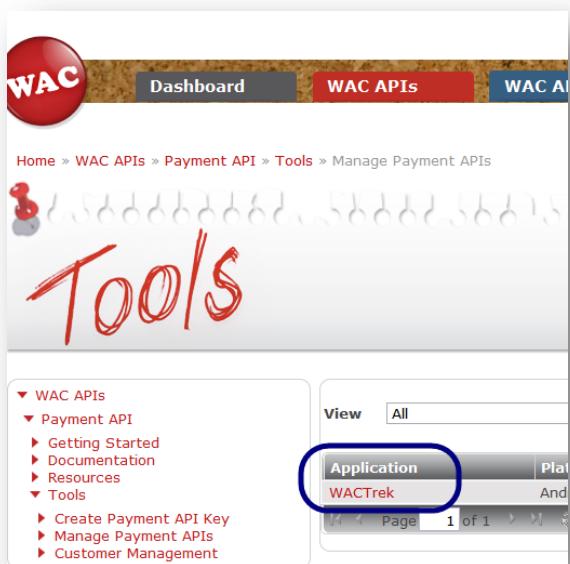
B) Localize Product Names for Each Market

A) Add Icons & Descriptions

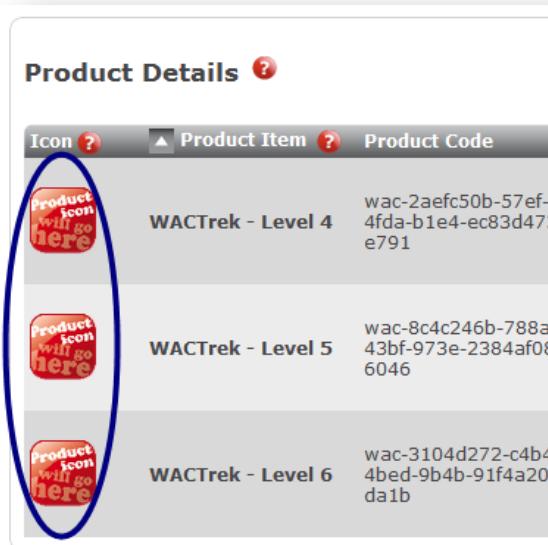
- 1) Click **Dashboard** and then **Manage and Certify Payment API**.



- 2) Open the app you just registered.



- 3) WAC provided default icons while you were defining your app's product items. However, you must replace these with your own icons before your app can be accepted for market.



* The following are requirements and guidelines for the icons you upload to your in-app purchase items.

WAC Icon Guidelines

Requirements

- **Size:** 200 x 200 pixels
- **Format:** PNG

An Easy-to-Manage Name

Make it easy to keep track if you're publishing to multiple markets.

<app name>_<item name>_<market>.PNG

An Effective Design

Icons are small, so make them easy to understand -- simple and distinct gets noticed.

Effective Icons	Ineffective Icons
<ul style="list-style-type: none"> • Distinct centered image(s) • Distinct text (if any) • Distinct colors 	<ul style="list-style-type: none"> • Cluttered images • Small or competing text • Colors that blend together 

- 4) When you are ready to update your product icons, start by clicking edit for the first one.

Product Details					Add Product Item
Icon ?	▲ Product Item ?	Product Code	Price Status	Actions	
	WACTrek - Level 4	wac-2aefc50b-57ef-4fda-b1e4-ec83d473e791	Yes	Edit Localize Delete	
	WACTrek - Level 5	wac-8c4c246b-788a-43bf-973e-2384af086046	Yes	Edit Localize Delete	
	WACTrek - Level 6	wac-3104d272-c4b4-4bed-9b4b-91f4a201da1b	Yes	Edit Localize Delete	

- 5) Use the **change links** to update this app product's description and icon.

Note: The icon must be 200 x 200 pixels and in PNG format.

Edit Product Item

You can edit the name of your product item, upload an icon, and change the price. You can also localize product name on a per operator basis

Product Name	WACTrek - Level 4	Change
Product Description	No description added	Change
Product Code	wac-2aefc50b-57ef-4fda-b1e4-ec83d473e791	
Product Icon Change		

- 6) When ready, click **Okay**.

Edit Product Item

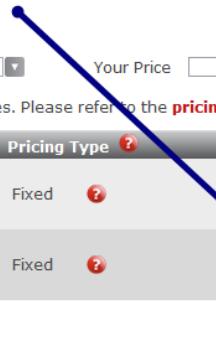
You can edit the name of your product item, upload an icon, and change the price. You can also localize the icon and product name on a per operator basis

Product Name	WACTrek - Level 4	Change
Product Description	With level 4 your ship gets Neutron Torpedoes -- enemies beware!	Save
Product Code	wac-2aefc50b-57ef-4fda-b1e4-ec83d473e791	
Product Icon Change		
Your Currency <input type="text"/> Your Price <input type="text"/> Suggest Price		

* Prices may be exclusive or inclusive of sales and/or other taxes. Please refer to the [pricing guide](#) for more details.

Country	Operator	Localized Name	Pricing Type ?	Price	Actions
Korea (South)	Kt	WACTrek - Level 4	Fixed ?	1100 KRW	change
Germany	Telekom DE	WACTrek - Level 4	Fixed ?	0.75 EGP	change

Cancel **Okay**



- 7) You see that the icon is updated. Update the rest of your product icons and descriptions.

Product Details ?

Add Product Item

Icon ?	Product Item ?	Product Code	Price Status	Actions
	WACTrek - Level 4	wac-2aefc50b-57ef-4fda-b1e4-ec83d473e791	Yes	Edit Localize Delete
	WACTrek - Level 5	wac-8c4c246b-788a-43bf-973e-2384af086046	Yes	Edit Localize Delete
	WACTrek - Level 6	wac-3104d272-c4b4-4bed-9b4b-91f4a201da1b	Yes	Edit Localize Delete

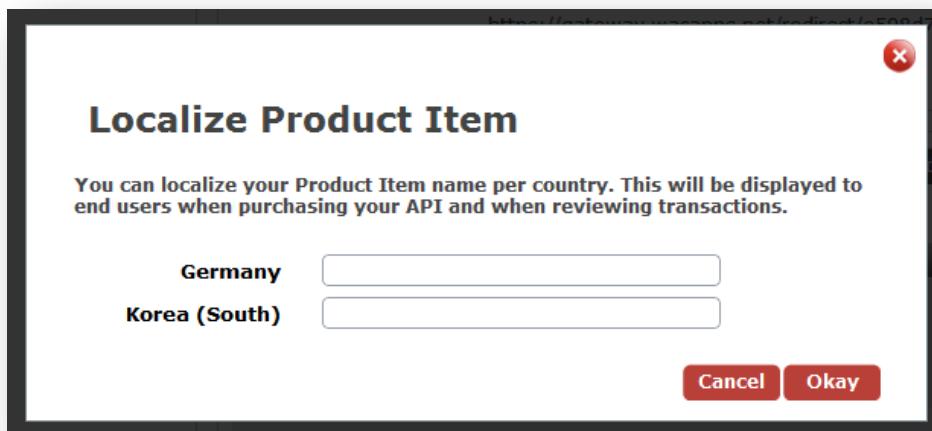
B) Localize Product Names for Each Market

You can elect to have each product's name appear in the language of the purchaser's market. This section shows how.

- 1) Click **Localize**.

Icon ?	Product Item ?	Product Code	Price Status	Actions
	WACTrek - Level 4	wac-2aefc50b-57ef-4fda-b1e4-ec83d473e791	Yes	Edit Localize (circled) Delete
	WACTrek - Level 5	wac-8c4c246b-788a-43bf-973e-2384af086046	Yes	Edit Localize Delete
	WACTrek - Level 6	wac-3104d272-c4b4-4bed-9b4b-91f4a201da1b	Yes	Edit Localize Delete

- 2) Each market you chose previously appears on the screen. Enter the product name customers in those markets will see and click **Okay**. Then repeat for your other app products.



Step 5: Update Your App

Now that you have told WAC's payment gateway what to do when customers make purchases inside your app, it's time to add the actual WAC in-app payment functionality to your app.

This section explains how to update your app and illustrates each step by looking at the working code of the SDK sample app. (*Reminder: You installed and ran WAC's PHP sample app in [Step 1: Run the Sample App](#) – if you skipped that step, go complete it before continuing.*)

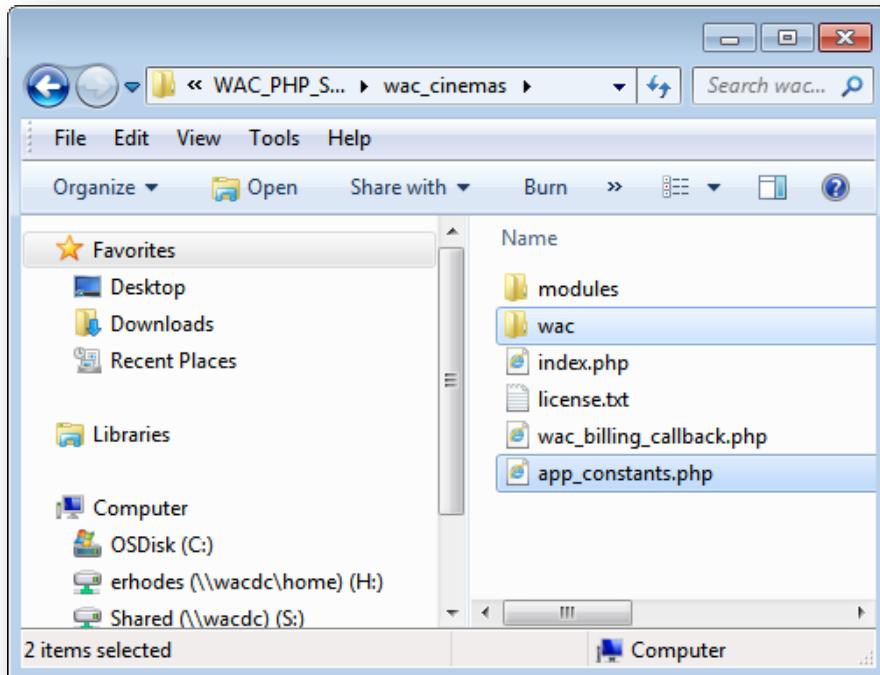
A) Enable In-App Payments in Your Code

B) Add Your “Pay By” Icons

C) Add a Tax Notice for Your Customers

A) Enable In-App Payments in Your Code

- 1) Make certain that you have already completed [Step 1:](#).
- 2) From the SDK's **wac_cinemas** folder, copy the following to your app's folder:
 - **wac** (*This is Wac's SDK API library.*)
 - **app_constants.php** (*This is where your app looks for your unique WAC API keys.*)



- 3) Update the **app_constants.php** file in your app folder your WAC credentials (visible on the [Error! Reference source not found.](#)).

```
<?php

/**
 * Copyright (c) 2011, WAC Application Services Ltd. All Rights Reserved.
 * The use of this SDK is subject to the terms and conditions in license.txt
 */

// Developer Consumer Key and Consumer Secret

$consumerKey = 'wac_b4493a4d42db5de3cf86d41b71d1a1dda5553c51';
$consumerSecret = 'cd0bc75f7bb05db1b4b7760a5bddb3f5760b44c';

// Application ID

$appID = 'wae_84c5ffaf_5fdf_4920_be73_1b855bf0501d';

// Application User/ Developer Name

$appUserName = 'developer1';

// Application Callback URL

$redirectURI = "http://localhost/wac_cinemas/wac_billing_callback.php";
?>
```

- 4) Update your app's code to use WAC in-app payment functionality. To illustrate how, this table describes how each function is implemented in the WAC SDK sample app, primarily its **index.php** file.

- **Setup**
- **Payment Initialization**
- **Billing Redirect / Callback**
- **Utilities**

Setup

Action	Example in SDK Sample App
Include the WAC PHP payment library	<pre>// Include SDK files in application require_once 'wac/wac.php';</pre>
Initiate WacPaymentService to get a handle to access WAC PHP payment library API's.	<pre>// Create a WacPaymentService instance \$consumer = new WacPaymentService();</pre>
Initialize WacPaymentService in your application.	<pre>/** * Initializes the application credentials to process the payment (environment, consumerKey, consumerSecret, appID, appUserName, redirectURI). * This data is obtained when the application is registered with WAC. * * @param urlType string Type of Environment URL * @param consumerKey string Developer ID/Consumer Key given by WAC * @param consumerSecret string Developer Secret * @param appID Application string Identifier for a given Application * @param appUserName string Username of the developer registered with WAC * @param redirectURI string Callback URI for the application */ \$consumer->initService(\$environment, \$consumerKey, \$consumerSecret, \$appID, \$appUserName,\$redirectURI);</pre>

Action	Example in SDK Sample App												
Determine whether to offer the customer WAC billing as an option by checking to see whether WAC billing is available to the customer as a payment option. <i>(optional)</i> Note: This API call is not currently included in the WacCinemas sample app and is called as explained here. This API is optional.	<p>If using this optional API, call it before WAC initialization. If the variable returns TRUE, WAC billing is supported; if false, WAC billing is not supported.</p> <pre> /** * @param string Type of Environment URL * @param string Developer ID/Consumer Key given by WAC * @param string Developer Secret * @param string Application string Identifier for a given Application * @param string Username of the developer registered with WAC * @return boolean True if wac billing is available */ \$status = \$consumer->checkBillingAvailability(\$environment, \$consumerKey, \$consumerSecret, \$appID, \$appUserName); </pre> <p>Note: Because the customer will not have been identified this early in the transaction, note these possible outcomes and their causes:</p> <table border="1"> <thead> <tr> <th>Outcome</th><th>Cause</th></tr> </thead> <tbody> <tr> <td>True</td><td> <ul style="list-style-type: none"> App is LIVE App published to this operator Individual user can use WAC </td></tr> <tr> <td>True</td><td> <ul style="list-style-type: none"> Mobile device is on WIFI App published in the country of the access point IP address but not the operator of the user </td></tr> <tr> <td>False</td><td> <ul style="list-style-type: none"> Mobile device is on operator network App not LIVE or not published to this operator </td></tr> <tr> <td>False</td><td> <ul style="list-style-type: none"> Mobile device is on WIFI App not LIVE or not published in the country of the access point IP address </td></tr> <tr> <td>*True</td><td> <ul style="list-style-type: none"> WAC billing is technically available however the particular user is not accepted for this transaction (user is blacklisted, lacks adequate credit, etc.) <p>Note: Operators do not yet provide customer-level access for this method, which results in a TRUE result. This will be addressed in a future release.</p> </td></tr> </tbody> </table>	Outcome	Cause	True	<ul style="list-style-type: none"> App is LIVE App published to this operator Individual user can use WAC 	True	<ul style="list-style-type: none"> Mobile device is on WIFI App published in the country of the access point IP address but not the operator of the user 	False	<ul style="list-style-type: none"> Mobile device is on operator network App not LIVE or not published to this operator 	False	<ul style="list-style-type: none"> Mobile device is on WIFI App not LIVE or not published in the country of the access point IP address 	*True	<ul style="list-style-type: none"> WAC billing is technically available however the particular user is not accepted for this transaction (user is blacklisted, lacks adequate credit, etc.) <p>Note: Operators do not yet provide customer-level access for this method, which results in a TRUE result. This will be addressed in a future release.</p>
Outcome	Cause												
True	<ul style="list-style-type: none"> App is LIVE App published to this operator Individual user can use WAC 												
True	<ul style="list-style-type: none"> Mobile device is on WIFI App published in the country of the access point IP address but not the operator of the user 												
False	<ul style="list-style-type: none"> Mobile device is on operator network App not LIVE or not published to this operator 												
False	<ul style="list-style-type: none"> Mobile device is on WIFI App not LIVE or not published in the country of the access point IP address 												
*True	<ul style="list-style-type: none"> WAC billing is technically available however the particular user is not accepted for this transaction (user is blacklisted, lacks adequate credit, etc.) <p>Note: Operators do not yet provide customer-level access for this method, which results in a TRUE result. This will be addressed in a future release.</p>												

Action	Example in SDK Sample App
Initiate WacPaymentService to get a handle to access WAC PHP payment library API's.	<pre>// Create a WacPaymentService instance \$consumer = new WacPaymentService();</pre>
Initialize WacPaymentService in your application.	<pre>/** * Initializes the application credentials to process the * payment (environment, consumerKey, consumerSecret, appID, * appUserName, redirectURI). * This data is obtained when the application is registered * with WAC. * * @param urlType string Type of Environment URL * @param consumerKey string Developer ID/Consumer Key given * by WAC * @param consumerSecret string Developer Secret * @param appID Application string Identifier for a given * Application * @param appUserName string Username of the developer * registered with WAC * @param redirectURI string Callback URI for the application */ \$consumer->initService(\$environment, \$consumerKey, \$consumerSecret, \$appID, \$appUserName,\$redirectURI);</pre>
Fetch the list of products for application	<pre>// Fetch the list of products and their details for an Application in the form of an array. \$drop_down_options = \$consumer->listProductItems();</pre>

Payment Initialization

Action	Example
Payment Initialization	<p><i>To start the authorization process for payment please do the following:</i></p> <pre>// Initialize the charge Payment process \$consumer->initChargePayment(\$appProductKey);</pre>
Initialize Check transaction	<p><i>To start the authorization process for check transaction please do the following:</i></p> <pre>// Initialize the Check Transaction process \$consumer->initCheckTransactions(\$ref_code);</pre>

Action	Example
Initialize List transactions	<p><i>To start the authorization process for List transactions please do the following:</i></p> <pre>// Initialize the List Transaction process \$consumer->initListTransactions();</pre>

Billing Redirect / Callback

The authorization flow will trigger the redirect URI that you received when you registered the application with WAC. You can find your redirect on your WAC portal [Create Payment Key screen](#). In the sample application, see this callback function illustrated for payment/check/list transactions in the WAC SDK sample application's **wac_billing_callback.php** file.

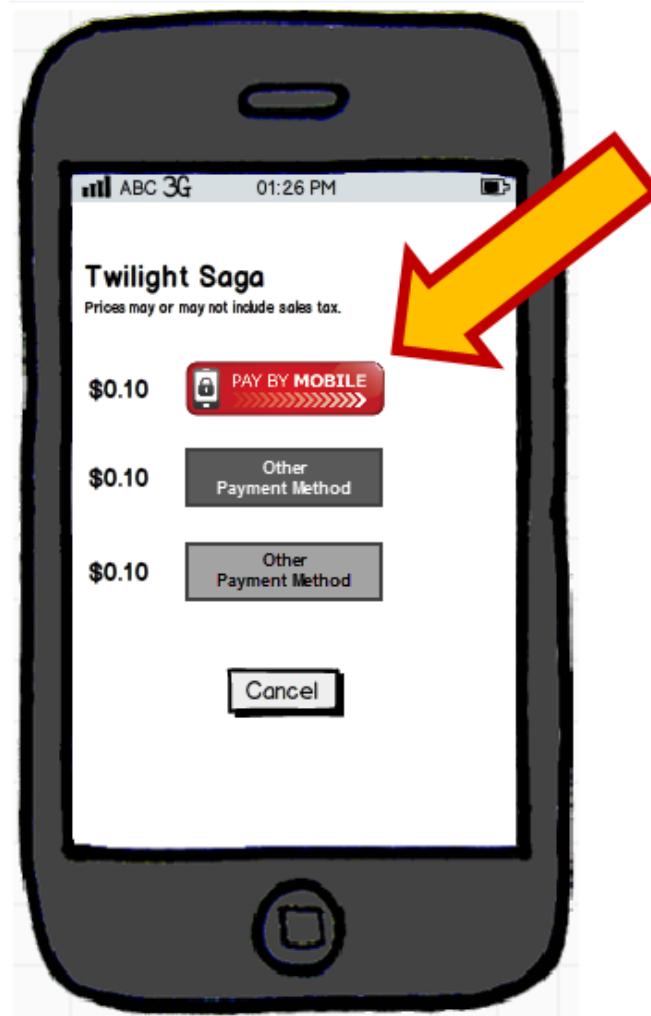
Action	Example
Make Payment	<p>Reserve and capture: After authorization, reserve the payment using <code>reservePayment()</code>. This returns the complete details required to make a payment which can be used later to make the actual payment.</p> <p>Example:</p> <pre>// Reserve the payment \$reserved = \$consumer->reservePayment(\$appProductKey); To capture the previously reserved fund, developer can invoke <code>capturePayment</code> after fulfillment of purchase (service or content delivery) is successful. // Capture the reserved payment \$res = \$consumer->capturePayment(\$reserved, \$userID);</pre>
Check Transaction	<pre>\$res = \$consumer->checkTransactions(\$ref_code, \$userID);</pre>
List Transaction	<pre>\$res = \$consumer->listTransactions(\$userID);</pre>

Utilities

Action	Example																								
Set user locale	The below line Sets the Locale for a user. This will be used to show the error/info/warn messages in the user preferred language. \$consumer->setUserLocale("en-US");																								
Debugging	All the errors are logged to apache log with tag as "WAC". To enable debug logs for Http debugging please add the below line \$consumer->debugLog('true');																								
Get product details for a particular product ID	// Get detail of a particular ItemID \$consumer->getProductDetails(\$appProductKey)																								
Show payment successful page	The following line of code can be added to the application to show a payment successful page to the user. The function takes a callback URL as a parameter to which the control should be returned after user leave the payment successful page. \$consumer->showPaymentSuccessPage(\$callback="<Callback URL to return to the application>");																								
IP spoofing for testing the application for different geographies	In order to test the application in a different geography, it is necessary for the NAPI invocation to originate from an IP in that geography. The following line can be used to achieve this with any IP address below. // Sets a spoofed IP \$consumer->setSpoofedIP('<IP_ADDRESS>'); <table border="1"> <thead> <tr> <th>Region</th> <th>Operator(s)</th> <th>IP</th> </tr> </thead> <tbody> <tr> <td>US</td> <td>ATT</td> <td>12.207.19.228</td> </tr> <tr> <td>Germany</td> <td>DT, Telefonica</td> <td>80.187.110.132</td> </tr> <tr> <td>Spain</td> <td>Telefonica</td> <td>79.146.82.130</td> </tr> <tr> <td>Bulgaria</td> <td>TAG</td> <td>31.211.128.0</td> </tr> <tr> <td>Korea</td> <td>KT, SKT, LGU+</td> <td>61.47.192.22</td> </tr> <tr> <td>UK</td> <td>Vodafone</td> <td>192.165.213.18</td> </tr> <tr> <td>Philippines</td> <td>SMART</td> <td>120.28.64.69</td> </tr> </tbody> </table>	Region	Operator(s)	IP	US	ATT	12.207.19.228	Germany	DT, Telefonica	80.187.110.132	Spain	Telefonica	79.146.82.130	Bulgaria	TAG	31.211.128.0	Korea	KT, SKT, LGU+	61.47.192.22	UK	Vodafone	192.165.213.18	Philippines	SMART	120.28.64.69
Region	Operator(s)	IP																							
US	ATT	12.207.19.228																							
Germany	DT, Telefonica	80.187.110.132																							
Spain	Telefonica	79.146.82.130																							
Bulgaria	TAG	31.211.128.0																							
Korea	KT, SKT, LGU+	61.47.192.22																							
UK	Vodafone	192.165.213.18																							
Philippines	SMART	120.28.64.69																							

B) Add Your “Pay By” Icons

For the screen in your app where your customers choose how they want to pay, WAC provides a variety of icons you can use for the WAC payment option.



Note: The WAC billing option button is only needed if there are multiple billing choices available to the customer. When WAC billing is the only option, your app can simply take the customer directly to the purchase confirmation screen.

There are two ways to use WAC's payment buttons in your app:

- **Embed a link to the graphic in your app:** If you embed a link to the button graphic (links provided in the table below) and WAC's server has a button available in the IP address' local language, then this button will automatically appear localized to the purchaser. WAC recommends this method because a localized payment button helps inspire trust for customers in non-English speaking countries.
- **Embed the graphic directly in your app:** If you embed WAC's payment button directly in your app it will *not* automatically match to the customer's local market. You can download graphics from the links in the table below. You can also download buttons all languages available at <https://www.wacapps.net/pay-by-mobile-icon>.

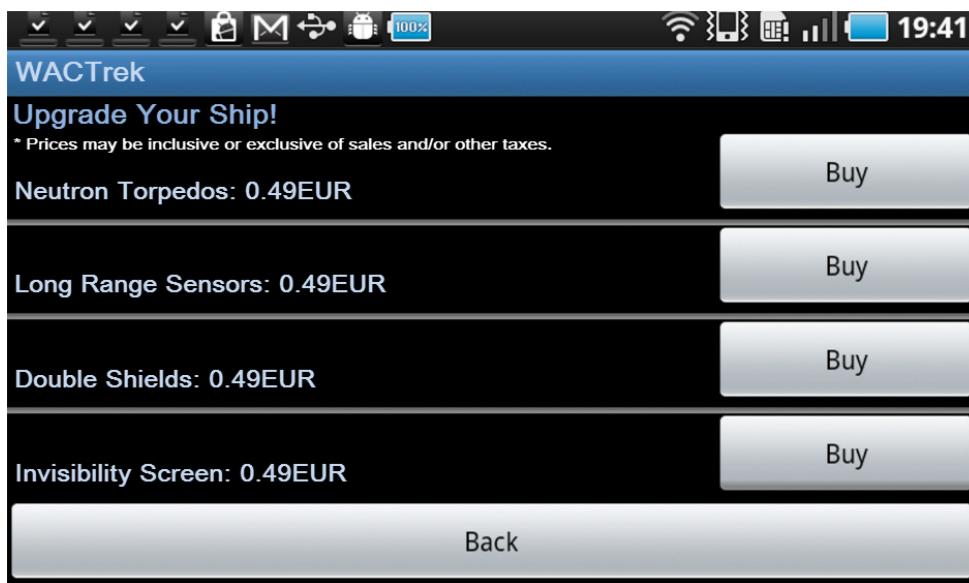
Image	URL
	http://icon.wacapps.net/icon/37w-x-23h-px.png
	http://icon.wacapps.net/icon/50w-x-34h-px.png
	http://icon.wacapps.net/icon/60w-x-38h-px.png
	http://icon.wacapps.net/icon/120w-x-30h-px.png
	http://icon.wacapps.net/icon/150w-x-40h-px.png
	http://icon.wacapps.net/icon/150w-x-60h-px.png
	http://icon.wacapps.net/icon/180w-x-113h-px.png
	http://icon.wacapps.net/icon/reversed-37w-x-23h-px.png
	http://icon.wacapps.net/icon/reversed-50w-x-34h-px.png
	http://icon.wacapps.net/icon/reversed-60w-x-38h-px.png
	http://icon.wacapps.net/icon/reversed-120w-x-30h-px.png
	http://icon.wacapps.net/icon/reversed-150w-x-40h-px.png
	http://icon.wacapps.net/icon/reversed-150w-x-60h-px.png
	http://icon.wacapps.net/icon/reversed-180w-x-113h-px.png

C) Add a Tax Notice for Your Customers

The mobile operator for each market determines whether item prices include applicable value added or sales tax or other such taxes and how they are added to the customers total at checkout. In addition, local consumer laws may require disclosure of price variations.

You are responsible for the wording and making sure that it is clear to consumers whether there is a price change and how taxes are added. Any such wording should appear clearly and concisely on the purchase screen.

Example for initial purchase screen: *Please note that the price of item, and the applicable VAT or sales tax, may vary depending on your mobile operator and/or the country in which you made your purchase.*



Step 6: Test Your App

Now comes the fun part – buy some of your app’s products and make sure it all works.

A) Verify In-App Payments Are Working

B) Test Prices in Different Regions

A) Verify In-App Payments Are Working

After you create your API keys, our app status updates to **TEST**, indicating you are ready to practice some test purchases and see how they go. At this stage all purchases are only simulations, so test all you want and no money will be transacted.

The screenshot shows the WAC API Tools interface. At the top, there are tabs for Dashboard, WAC APIs (which is highlighted with a red oval and a yellow box labeled '1'), and WAC APPS. Below the tabs, a breadcrumb navigation shows Home > WAC APIs > Payment API > Tools > Manage Payment APIs. The main area has a cartoon illustration of tools like a wrench and a screwdriver. On the left, a sidebar menu under 'WAC APIs' has 'Manage Payment APIs' circled with a blue oval and a yellow box labeled '2'. The main content area shows a table with columns: Application, Platform, Version, and Status. One row in the table is highlighted with a yellow box labeled '3' and has a blue oval around the 'Status' column, which contains the word 'TEST'.

Status	Description
TEST	You have created your WAC API keys (Step 3: Create Your API Keys). During TEST status, in-app purchases occur only in WAC’s testing “sandbox” – no money is transacted.
PROCESSING	You have requested that WAC certify your app (Step 7: Certify Your App). While WAC processes your certification, no updates are permitted to your app or your WAC account.
ACCEPTED	WAC has concluded certification and accepted your app to continue on to publishing (Step 8: Set Your API Keys Live).
LIVE	You have set your app’s API keys live in at least one market, after which all in-app purchases from your app’s WAC API keys in the market will be <i>real</i> monetary transactions.

Notes

- While your app has TEST status, your in-app purchases occur in WAC's testing "sandbox" and no money is transacted.
- If you are testing an app outside the United States, you must set your system to initiate its API calls from a US IP address (also called "spoofing"). You can do this by [adding a spoofing line](#) to the file where you initialize the Wac payment service (in the sample SDK app, this is in the index.php file).
- While WAC Operations and Support works hard to keep the Developer Sandbox available 24 hours a day, 7 days a week, occasional system downtime may occur. To check the real-time status of the sandbox at any time, click [here](#).
- Before continuing, ensure that the device you run your app on – mobile phone or computer browser – is set to your correct local time. WAC's payment gateway will have your app report a 'failed-signature validation' error if it receives calls from a system whose time varies more than 3 minutes from your local time.
- Transactions and their downloads must complete within 5 minutes or reserved customer funds will be released – test to be certain your app's transactions can complete and initiate the capture call before this time limit.

Platform	Connection Type	Description															
Phone	<ul style="list-style-type: none"> • SIM from market operator, and • WIFI off 	<ol style="list-style-type: none"> 1. The app should recognize you as a customer of the operator and let you complete the transaction successfully. Purchases appear on screen under Items you own. (Note: This test is only available for operators updated to WAC's latest API build.) 2. Customers using the live app will receive SMS confirmation of the purchase (however during testing SMS messages are not sent). 															
Phone	<ul style="list-style-type: none"> • SIM from another operator, or • No SIM, or • WIFI only 	<ol style="list-style-type: none"> 1. App reports it cannot detect your phone number with the market operator. 2. App prompts for your phone number. 3. WAC's test environment lets you simulate different scenarios by entering these phone numbers: <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Scenario</th> <th>What to Enter</th> <th>Message</th> </tr> </thead> <tbody> <tr> <td><i>Successful payment</i></td> <td>+19724891737</td> <td>Payment successful.</td> </tr> <tr> <td><i>Spend limit error</i></td> <td>+19724891648</td> <td>You have reached your operator spend limit. Please contact your operator for help.</td> </tr> <tr> <td><i>Payment failed</i></td> <td>+19724892325</td> <td>Payment failed, please try again later. You will not be charged for this transaction.</td> </tr> <tr> <td><i>Operator not supported</i></td> <td>Any operator number not live on WAC</td> <td>Operator not supported.</td> </tr> </tbody> </table> 4. App prompts you for PIN you received by SMS, which customers will receive for live apps. In testing, no PIN is sent. Instead, enter PIN 4681. 5. Results match scenario. Successful payments are followed by purchased items appearing on screen under Items you own. 6. Customers using the live app will receive SMS confirmation of the purchase, however these are not sent during your testing. 	Scenario	What to Enter	Message	<i>Successful payment</i>	+19724891737	Payment successful.	<i>Spend limit error</i>	+19724891648	You have reached your operator spend limit. Please contact your operator for help.	<i>Payment failed</i>	+19724892325	Payment failed, please try again later. You will not be charged for this transaction.	<i>Operator not supported</i>	Any operator number not live on WAC	Operator not supported.
Scenario	What to Enter	Message															
<i>Successful payment</i>	+19724891737	Payment successful.															
<i>Spend limit error</i>	+19724891648	You have reached your operator spend limit. Please contact your operator for help.															
<i>Payment failed</i>	+19724892325	Payment failed, please try again later. You will not be charged for this transaction.															
<i>Operator not supported</i>	Any operator number not live on WAC	Operator not supported.															

Step 7: Certify Your App

When your in-app payments are working to your satisfaction, you are ready to have WAC certify your API keys so your app can initiate real purchases.

To do this, first complete your banking details so WAC knows how to pay you for your in-app sales. Then complete the compliance questionnaire for the operator markets you will sell your app in.

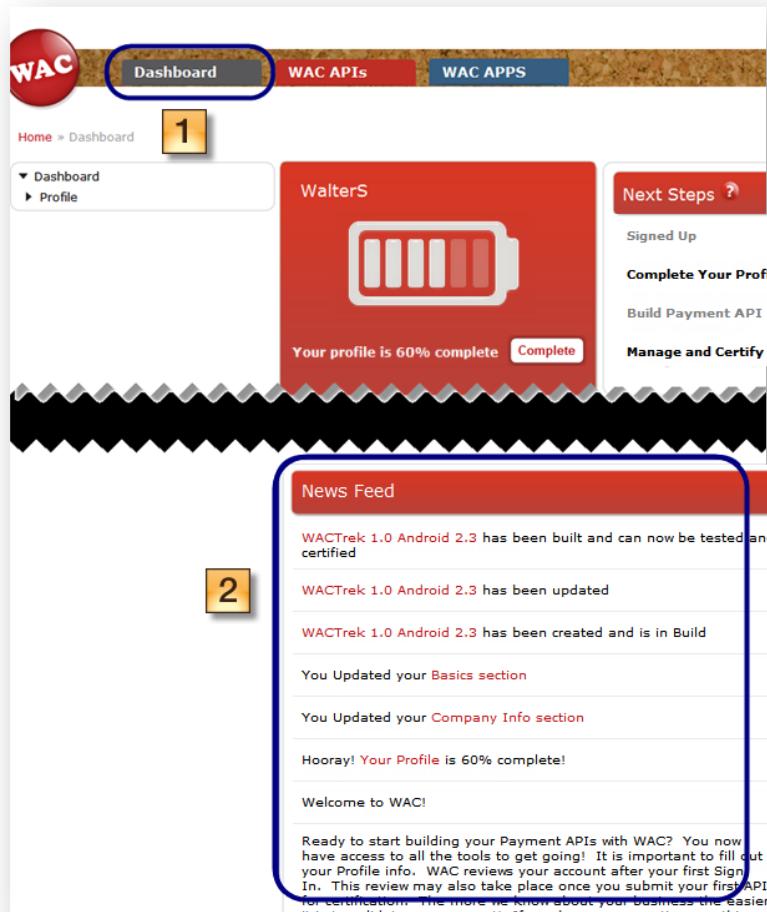
A) Tell WAC How to Pay You

B) Have WAC Certify Your App

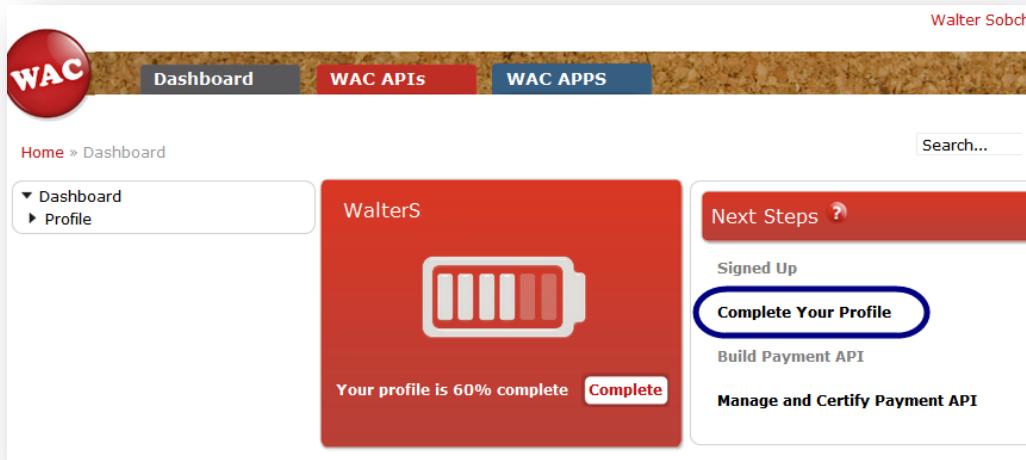
A) Tell WAC How to Pay You

- 1) Click **Dashboard**.

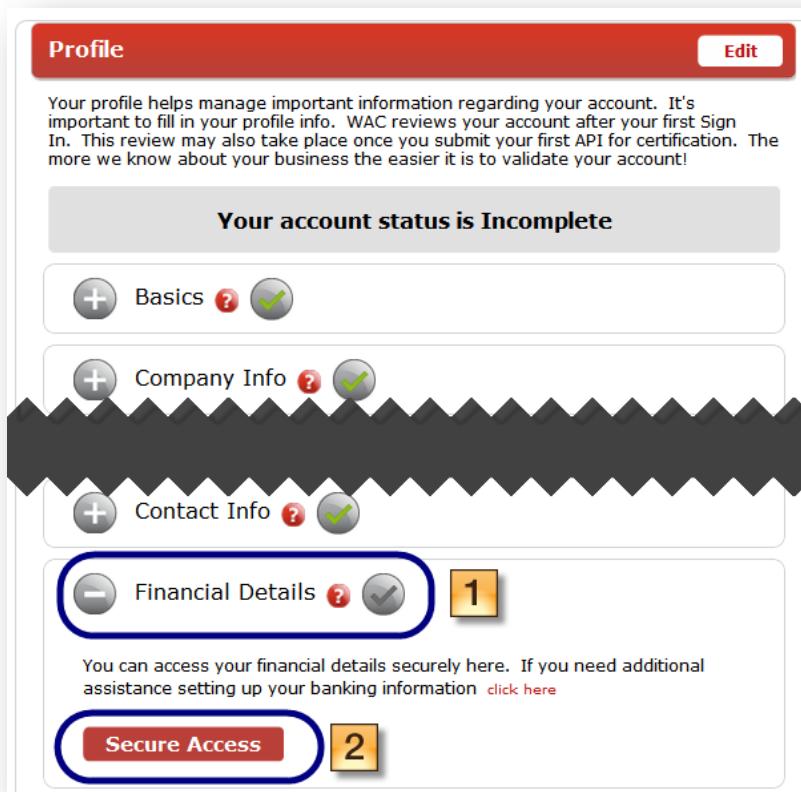
Tip: Notice that you can always see your current status and previous actions in the Newsfeed section of this screen.



- 2) Click **Complete Your Profile**.



- 3) In the Financial Details section, click **Secure Access**.



4) Complete the **Developer Information banking screen**.

Important Notes:

- Before you start entering your details, be certain you have a bank statement or your bank details handy.
- Ensure that your banking details are correct -- WAC cannot make payments to you unless your information is accurate.
- Bank details must be provided in English.
- After you have submitted your details you will receive confirmation that your account has been successfully set up. If there is a problem, you will be notified that your account has not been set up and why.
- Be certain to review [WAC's Settlement FAQ](#) for important details about getting credited for your app sales.
- If you have any questions about this form or your financial account with WAC, please contact settlement@wacapps.net.

Fields marked with a red asterisk (*) must be completed.

Developer Information

Header Information

Company: * Sobchak Security ?

Address Details

Room Number: ?
 Floor: ?
 Building Code: ?
 House Number: * 2556 ?
 Street: * Third Street ?
 Street 2: ?
 Street 3: ?
 Street 4: ?
 Street 5: ?
 District: ?
 City: * Long Beach ?
 Country: * USA ?
 Postal Code/Zip Code: * 90804 ?

PO Box: ?
 PO Box Post Code: ?

Contact Details

Telephone: 925-201-8501 ?
 Mobile Phone: ?
 E-Mail Address: * walter.s.sobchak@mailinator.com ?

Tax Details

Tax Identification Number: 111-11-1111 ?
 Tax/VAT Country: USA ?

VAT Registration Number: ?

Bank Details

Add New Bank Add New Bank

Bank Country: * USA ?
 Bank Key: * 111000025 ?
 Bank Name: Bank of America ?
 Bank Account: * 0142923887 ?
 Account Holder: * Walter S. Sobchak ?
 Control Key: ?
 Reference: ?
 IBAN: ?
 Bank Account Currency: * USD ?

Payment Details

Payment Methods: * US ACH (USD) ?
 Payment Terms: Z041 ?

Field	Description
Name	This field is important for tax and legal purposes: If you are a registered company, enter your company legal name. Otherwise, enter your full name.
Address & Contact Details	If you are a company, enter the company information. Otherwise enter your personal information.
Tax Identification Number	For developers selling within the U.S., this is a nine-digit number obtained from the U.S. Social Security Administration (SSA) or the Internal Revenue Service (IRS). Leave this field blank if you do not have a Tax Identification Number.
TAX/VAT Country	If you are VAT/TAX registered in a country other than your country of residence and you wish to use this VAT/TAX registration for the sale of your app(s), enter that country's VAT/TAX registration. If you are not registered for VAT, enter your own country.
VAT Registration Number	Enter your VAT ID, i.e. VAT registration, number. This number identifies you to your tax authority for VAT purposes.
Bank Country	Enter the country of your bank

Field	Description																																													
Bank Key	Your bank key -- also known as your sort code or routing code in the US -- is the code that identifies your bank branch. You will find it on your bank statements and printed as the middle group of computer-type figures at the bottom of your checks. It can be up to 15 figures depending on country. US dollar payments to US bank accounts always require a valid bank key/sort code/routing code.																																													
	<table border="1"> <thead> <tr> <th>Country</th> <th>Sort Code</th> <th>Bank Account</th> <th>IBAN</th> <th>SWIFT</th> </tr> </thead> <tbody> <tr><td>Germany</td><td>29050101</td><td>12226411</td><td>DE75290501010012226411</td><td>SBREDE2XXX</td></tr> <tr><td>Spain</td><td>21000178</td><td>0200452310</td><td>ES5221000178750200452310</td><td>CAIXESBBXXX</td></tr> <tr><td>Great Britain</td><td>160079</td><td>10084423</td><td>GB59RBOS16007910084423</td><td>RBOSGB21021</td></tr> <tr><td>Ireland</td><td>900199</td><td>85856560</td><td>IE37BOFI90019985856560</td><td>BOFIIE2D</td></tr> <tr><td>Italy</td><td>0102501603</td><td>100000018690</td><td>IT53A0102501603100000018690</td><td>IBSPITTM</td></tr> <tr><td>Netherlands</td><td>SOGE</td><td>0270202641</td><td>NL89S0GE0270202641</td><td>S0GENL2A</td></tr> <tr><td>Portugal</td><td>00070038</td><td>00041560008</td><td>PT50000700380004156000831</td><td>BESCPTPL</td></tr> <tr><td>Greece</td><td>0710033</td><td>0000033031352140</td><td>GR8307100330000033031352140</td><td>MIDLGRAA</td></tr> </tbody> </table>	Country	Sort Code	Bank Account	IBAN	SWIFT	Germany	29050101	12226411	DE75290501010012226411	SBREDE2XXX	Spain	21000178	0200452310	ES5221000178750200452310	CAIXESBBXXX	Great Britain	160079	10084423	GB59RBOS16007910084423	RBOSGB21021	Ireland	900199	85856560	IE37BOFI90019985856560	BOFIIE2D	Italy	0102501603	100000018690	IT53A0102501603100000018690	IBSPITTM	Netherlands	SOGE	0270202641	NL89S0GE0270202641	S0GENL2A	Portugal	00070038	00041560008	PT50000700380004156000831	BESCPTPL	Greece	0710033	0000033031352140	GR8307100330000033031352140	MIDLGRAA
Country	Sort Code	Bank Account	IBAN	SWIFT																																										
Germany	29050101	12226411	DE75290501010012226411	SBREDE2XXX																																										
Spain	21000178	0200452310	ES5221000178750200452310	CAIXESBBXXX																																										
Great Britain	160079	10084423	GB59RBOS16007910084423	RBOSGB21021																																										
Ireland	900199	85856560	IE37BOFI90019985856560	BOFIIE2D																																										
Italy	0102501603	100000018690	IT53A0102501603100000018690	IBSPITTM																																										
Netherlands	SOGE	0270202641	NL89S0GE0270202641	S0GENL2A																																										
Portugal	00070038	00041560008	PT50000700380004156000831	BESCPTPL																																										
Greece	0710033	0000033031352140	GR8307100330000033031352140	MIDLGRAA																																										
	<p>Find and add your bank key as follows:</p> <ol style="list-style-type: none"> 1) Click the finder icon.  <ol style="list-style-type: none"> 2) Search for your bank and click to add its bank key. 																																													
Bank Name	Be certain to enter the full bank name.																																													
Bank Branch	Enter the bank branch name or bank branch number for Japanese bank accounts. If none enter the bank city.																																													
Bank Street	Enter the number and street where your bank branch is located.																																													
Bank City	Enter the city where your bank branch is located.																																													
Bank Account	Enter your bank account number.																																													
Account Holder	Enter account holder name exactly as it appears on your statement.																																													

Field	Description
Control Key	<p>The control key is also known as “account type” and indicates the type of account: checking or savings:</p> <ul style="list-style-type: none"> • 01: Checking • 02: Savings <p>Important: This key is mandatory for all U.S. dollar payments and mandatory if your bank account is held in Spain.</p>
Reference	Enter the reference that you would like to appear on your bank statement. For example, ‘Network API Sales’
IBAN	<p>Enter your International Bank Account Number if your bank account is held in the European Union. The IBAN consists of a two-letter country code, followed by two check digits and up to thirty alphanumeric characters known as the Basic Bank Account Number (BBAN). For example AD12 0001 2030 2003 5910 0100. To find out what your IBAN is, look on your paper statement - it is usually near your name and address along with the SWIFT code.</p> <p>Note: This field is mandatory in the European Union and optional everywhere else.</p>
Bank Account Currency	Indicate your bank’s type of currency.
SWIFT Code	<p>The unique identification code of a particular bank; this code is sometimes found on account statements. This bank identifier code is eight or 11 characters long and composed of the following:</p> <ol style="list-style-type: none"> 1. Four characters - bank code (only letters) 2. Two characters -Country code (only letters) 3. Two characters - location code (letters and digits). 4. Three characters - branch code, optional. Letters and digits.
Payment Methods and Payment Terms	Indicate the currency and method of payment transfer you would like for your app sale proceeds.

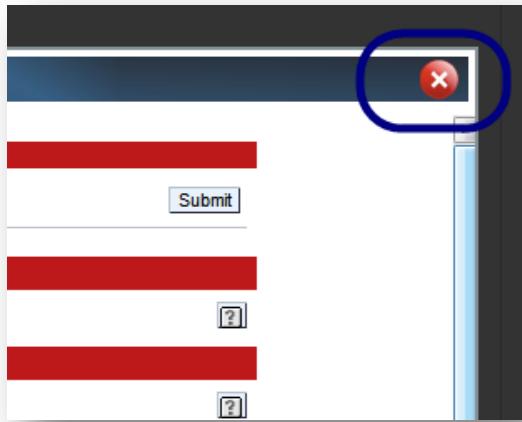
- 5) Click **Submit**.

The screenshot shows a 'Developer Information' form. At the top right is a 'Submit' button, which is circled in blue. Below it is a 'Header Information' section with a 'Company:' field containing 'Sobchak Security.' and a help icon. There is also a question mark icon in the top right corner of the form.

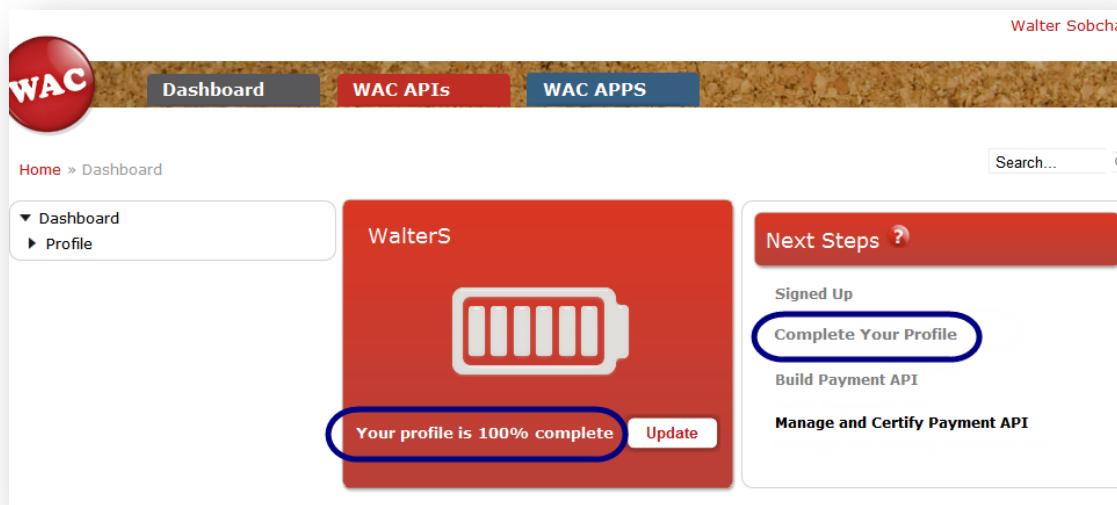
- 6) Be certain you see the submission is complete.
If there are errors, you will need to make corrections and resubmit or your changes will be lost.

The screenshot shows a 'Bank Details' form with a message 'Processing Successful' and another message 'Changes have been made', both highlighted with a blue circle. Below it is a 'Developer Information' section with a 'Submit' button. The entire window has a dark header bar.

- 7) Close the Bank Details popup.



- 8) The **Complete Your Profile** link should no longer be active – with you banking details added your WAC profile is now complete.



Within 24 hours WAC's compliance team will have verified your account and banking information and email you with the results.

 A screenshot of an email message. The recipient is 'walter.s.sobchak'. The subject line is 'WAC NOTIFICATION: Developer Account Status Update'. The message body contains a text view and a forward button. The text view part of the email reads:

Dear Walter, We are pleased to inform you that your application for a WAC Network API developer account has been approved. NEXT STEPS You can log into your WAC user account at any time at: www.wacapps.net SUPPORT Please direct any queries to: support@wacapps.net Or check out our developer forums at www.wacapps.net/forums Please add this email address to your contact list to avoid further communications being diverted to your JUNK folder. This email has been auto-generated à DO NOT REPLY DIRECTLY TO THIS EMAIL. Thank-you. WAC Support Team support@wacapps.net www.wacapps.net

B) Have WAC Certify Your App

You are ready to have WAC certify your app, after which your WAC APIs will be activated and your in-app purchases will be live transactions. When you submit your app for WAC certification, WAC verifies that:

- Your app is working correctly with your WAC API keys.
- Your app is set up correctly with the operator markets you selected.
- Your selected operator markets accept your app's content ratings.

Note: The time required for app certification is different for each operator. Please plan for the following certification times.

Operator	Country	Time to Certification
AT&T	United States	Within 48 hours
O2	Germany	Within 5 days
Telekom	Germany	Between 7 to 16 days
SMART	Philippines	Within 7 days
Mobitel	Bulgaria	Within 7 days
KT	Korea	Within 7 days
SKT	Korea	Within 7 days
LGU+	Korea	Within 7 days

Submit your app for WAC certification as follows:

- 1) Click **Manage and Certify Payment API**.

The screenshot shows the WAC Dashboard interface. At the top, there is a navigation bar with tabs: 'Dashboard' (selected), 'WAC APIs', and 'WAC APPS'. A user profile 'Walter Sobcha' is shown in the top right. Below the navigation, a sidebar on the left has links for 'Dashboard' and 'Profile'. The main content area features a large red box with the name 'WalterS' and a battery icon. It displays the message 'Your profile is 100% complete' with an 'Update' button. To the right, a 'Next Steps' section lists: 'Signed Up', 'Complete Your Profile', 'Build Payment API', and 'Manage and Certify Payment API'. The 'Manage and Certify Payment API' button is circled in blue.

- 2) Click your app.

The screenshot shows the 'WAC APIs' section of the WAC interface. The 'Dashboard' tab is selected. Below it, a 'Tools' page is displayed with a large red 'Tools' heading. On the left, a sidebar under 'WAC APIs' has a 'Payment API' section with links for 'Getting Started', 'Documentation', 'Resources', 'Tools', 'Create Payment API Key', and 'Manage Payment APIs'. The 'Tools' link is circled in blue. On the right, a table titled 'Application' shows a single entry 'WACTrek' with a 'Delete' link. The 'View' dropdown is set to 'All' and the page number is '1'.

- 3) Click **Get Certified**.

App Details

Application Name	Platform	Version	Status
WACTrek	Android 2.3	1.0	TEST
Description	Space shooter game.		

- 4) Review and agree to the terms and conditions. (You will receive an email with confirmation of this agreement.)

WAC Payment APIs Terms & Conditions

Section 2

Operator shall not collect and remit Sales Taxes described in the Sales Model, Taxation, payment and accounting section of this Agreement for sales of Virtual Gods in the following countries. You shall be solely responsible for the collection and remittance of Sales Tax as may be required in your own jurisdiction or in that of the country where the Subscriber purchases the Virtual Good:

1 (scroll down)

Bulgaria

2 Agree

- 5) Complete the certification screen as indicated in the table.

Get Your API Certified									
App Details									
Application Name	Platform								
WACTrek	Android 2.3								
Description	Space shooter game.								
Content Rating 									
Please tell us about the content of your App and rate it by selecting from the drop downs below.									
* Political Commentary 	Please Select 								
* Reference to Religion 	Please Select 								
* Violence 	Please Select 								
* Offensive Language 	Please Select 								
* Gambling 	Please Select 								
* Sexual Content 	Please Select 								
* Alcohol, Tobacco or Drug 	Please Select 								
Category Info									
* Select a Category 	Please Select 								
* Is your App intended for child audience? 	Please Select 								
App File or URL's									
We need a copy of your App to allow us to Certify it, please either upload a copy of your Appfile or enter the web URL's for web-based application.									
Native App <input checked="" type="radio"/>	Web based App <input type="radio"/>								
* Native App file: <input type="button" value="Choose File"/> No file chosen									
Recommended testing device: <input type="text"/>									
Product Item Descriptions									
To help our compliance team certify your payment API please provide a brief description of each Product Item within this App.									
<table border="1"> <thead> <tr> <th>Product Item</th> <th>Product Descriptions</th> </tr> </thead> <tbody> <tr> <td>* WACTrek - Level 4</td> <td>With level 4 your ship gets Neutron Torpedoes -- enemies beware!</td> </tr> <tr> <td>* WACTrek - Level 6</td> <td>Level 6 brings you Double Shields -- enemies beware!</td> </tr> <tr> <td>* WACTrek - Level 5</td> <td>Level 5 brings you Long Range Sensors -- enemies beware!</td> </tr> </tbody> </table>		Product Item	Product Descriptions	* WACTrek - Level 4	With level 4 your ship gets Neutron Torpedoes -- enemies beware!	* WACTrek - Level 6	Level 6 brings you Double Shields -- enemies beware!	* WACTrek - Level 5	Level 5 brings you Long Range Sensors -- enemies beware!
Product Item	Product Descriptions								
* WACTrek - Level 4	With level 4 your ship gets Neutron Torpedoes -- enemies beware!								
* WACTrek - Level 6	Level 6 brings you Double Shields -- enemies beware!								
* WACTrek - Level 5	Level 5 brings you Long Range Sensors -- enemies beware!								
Notes									
If there are any notes or comments that will assist our Compliance team please enter them below. This will help speed up certification.									
Notes for our Compliance team <input type="text"/>									
<input type="button" value="Cancel"/> <input type="button" value="Submit for Certification"/>									

Section	Description
Content Rating	Operators require this information to ensure apps comply with their market's content guidelines, such as the assignment of age ratings to an app's content.
Category Info	Indicate your app's category (such as game, book, etc.) and whether it is intended primarily for children.
Add File or URL's	Use the Choose File button and upload your app. If there is a particular device this app should be tested on, enter this.
Product Item Descriptions	Verify your descriptions.
Notes	Enter any special considerations that will help WAC's compliance team assess your app.

- 6) Click **Submit for Certification**.

Get Your API Certified

App Details

Application Name	Platform	Version
WACTrek	Android 2.3	1.0
Description	Space shooter game.	

Cancel **Submit for Certification**

- 7) The certification process begins.

Hooray! Certifying...

Your app has now been submitted for certification. The certification process typically takes 48 hours. You will be notified by email and within your Dashboard News Feed once compliance has finished reviewing the information submitted.

Once your Payment API has been approved you can then Set Live and your Payment API Key will available in production

- While your Payment API is being certified its status will appear as "Processing". You cannot make any changes to the Payment API while it's being certified
- Your app will be certified on a per market basis. It is therefore possible that it is approved for some markets and rejected for others.

[Learn More about the WAC API Certification Process](#)

You will receive an email from WAC compliance (compliance@wacapps.net) when your app's certification begins.

Note: The time required for app certification is different for each operator, note the certification duration for the operator market(s) you selected:

Operator	Country	Time to Certification
AT&T	United States	Within 48 hours
O2	Germany	Within 5 days
Telekom	Germany	Between 7 to 16 days
SMART	Philippines	Within 7 days
Mobitel	Bulgaria	Within 7 days
KT	Korea	Within 7 days
SKT	Korea	Within 7 days
LGU+	Korea	Within 7 days

While certification is underway, your app's status will be **PROCESSING** and no WAC application or account updates are permitted.

The screenshot shows the WAC API Tools interface. In the sidebar, under 'WAC APIs' > 'Payment API' > 'Tools', the 'Manage Payment APIs' link is circled in red. The main content area displays a table with columns: Application, Platform, Version, and Status. The first row shows 'WACTrek' with 'Android 2.3' as the platform, '1.0' as the version, and 'PROCESSING' as the status, which is also circled in blue. The table has a 'View' dropdown set to 'All'.

Application	Platform	Version	Status
WACTrek	Android 2.3	1.0	PROCESSING

When your app is certified, this status will change to **ACCEPTED**. WAC will also email you with the certification results.

The screenshot shows the WAC API Tools interface. On the left, there's a sidebar with a tree view under 'WAC APIs' and 'Payment API'. The 'Manage Payment APIs' option is highlighted with a red circle. On the right, there's a table with columns: Application, Platform, Version, Status, and Last. One row in the table is highlighted with a blue circle around the 'Status' column, which contains the word 'ACCEPTED'. The table also includes a 'View' dropdown set to 'All' and a 'Page' indicator showing '1 of 1'.

Application	Platform	Version	Status	Last
WACTrek	Android 2.3	1.0	ACCEPTED	26 Jan

Tip: App Status Definitions

Status	Description
TEST	You have created your WAC API keys (Step 3: Create Your API Keys). During TEST status, in-app purchases occur only in WAC's testing "sandbox" – no money is transacted.
PROCESSING	You have requested that WAC certify your app (Step 7: Certify Your App). While WAC processes your certification, no updates are permitted to your app or your WAC account.
ACCEPTED	WAC has concluded certification and accepted your app to continue on to publishing (Step 8: Set Your API Keys Live).
LIVE	You have set your app's API keys live in at least one market, after which all in-app purchases from your app's WAC API keys in the market will be <i>real</i> monetary transactions.

WAC will email you the certification results.

The screenshot shows an email message with the following details:

To: walter.s.sobchak
From: noreply@wacapps.net
Subject: WAC NOTIFICATION: WACTrek - Android
Status Update
Charset: UTF-8

Delete **Experimental View** **What's this?**

Text View **Forward**

Dear Walter,

We are pleased to inform you that the payment API for WACTrek - Android has been certified by WAC and is ready to be set live by you in the following WAC member operators/countries:

Telekom DE : ACCEPTED
AT&T : ACCEPTED

NEXT STEPS

Finally, when your app is certified / accepted, contact WAC at support@wacapps.net to do the following:

- Perform a formal run-through of your app.
- Have WAC help you set up your production credentials.
- Have WAC ensure your chosen operator markets have your production credentials.

Step 8: Set Your API Keys Live

You are ready to set your WAC API keys to status **LIVE**.

Application	Platform	Version	Status	Last Update
WACTrek	Android 2.3	1.0	LIVE	26 Jan 2012

This means that all transactions made with your app against your WAC API keys will be real monetary transactions.

- 1) Return to your **App Details screen**.

Application	Platform	Version	Status	Last Update
WACTrek	Android 2.3	1.0	LIVE	26 Jan 2012

- 2) In the **Markets section**, set your API keys live (a) in particular markets or (b) in all your selected markets at once.

The screenshot shows the 'App Details' section of a mobile application management interface. At the top, there are buttons for 'Back', 'Clone', and 'Delete'. Below this, the 'App Details' section includes fields for 'Application Name' (WACTrek), 'Platform' (Android), 'Version' (1.0), and 'Status' (ACCEPTED). A 'Description' field contains 'Space shoot'em up game.' Below these details is a large, decorative graphic of a jagged mountain range. Underneath the graphic, there is a 'Delete' button. The 'Markets' section follows, featuring a 'Set Live in All Markets' button (circled in red) and a 'Manage Markets' button. This section lists two countries: 'USA' (operator AT&T, status ACCEPTED) and 'Germany' (operator Telekom DE, status ACCEPTED), each with a 'Set Live in this Market' button (also circled in red). At the bottom, there is a 'Content and Category' section.

- 3) Click **Back**.

This screenshot shows the same 'App Details' page as the previous one, but it has been simplified. The 'Markets' section is no longer visible. The 'Back' button is highlighted with a blue circle. The remaining details are identical to the first screenshot: Application Name (WACTrek), Platform (Android), Version (1.0), and Description (Space shoot'em up game).

- 4) You see that your app's API status is now **LIVE**.

Application	Platform	Version	Status
WACTrek	Android	1.0	LIVE!

Notes:

- This status is LIVE if your app API keys are live in any one market – a status of LIVE does not indicate that your API keys are live in all markets.
- To remove LIVE status for your API keys in a market, simply return to the App Details screen and revoke the live status for one or all markets.

Markets			Revoke in All Markets	Manage Markets
Country	Operator	Status		
USA	AT&T	LIVE	Revoke	
Germany	Telekom DE	ACCEPTED		Set Live in this Market

- You can add also additional markets at any time, click **Manage Markets**.

Markets			Revoke in All Markets	Manage Markets
Country	Operator	Status		
USA	AT&T	LIVE	Revoke	
Germany	Telekom DE	ACCEPTED		Set Live in this Market

Step 9: Push App to Markets

That's it – your WAC APIs have LIVE status, indicating that you are ready to publish your app to its respective markets. Consider using WAC's publishing service, which makes it easy to publish to many major operators.

- **Having WAC Push Your App to Mobile Operators**
- **WAC's Support Policy**

Having WAC Push Your App to Mobile Operators

You can use WAC's easy publishing service to publish your app to subscribers of the world's leading mobile operators.

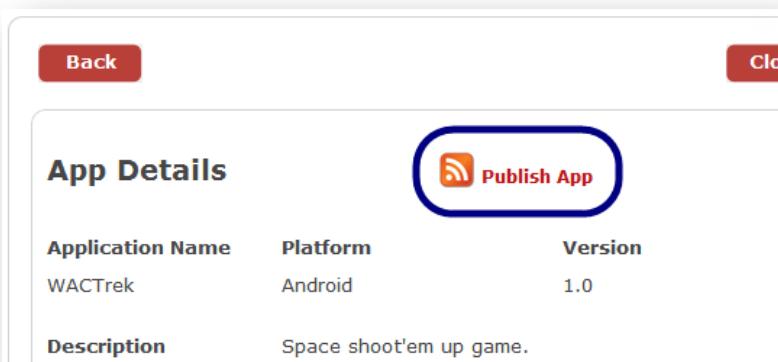
- See the list of operator markets: <https://www.wacapps.net/live-operators1>
- Find out more about WAC's easy publishing service: <https://www.wacapps.net/wac-apps>

If you decide to use this service for your app, start by activating your WAC API account for WAC's publishing service:

1. Go to your app's **App Details screen**.



2. Click the **Publish App icon**.



3. Your WAC account is now activated for WAC's publishing service and you are logged in there. From here follow the [WAC Apps Developer Guide](#) for illustrated instructions on how to continue.

WAC Support Policy

WAC's best practice is to resolve app user questions and issues within 3 working days. This commitment to quality customer service is both a good idea and a necessity:

- Customers who receive prompt, high-quality support are more likely to make repeat purchases and recommend the service to others.
- More than this, WAC's payment service is required to satisfy Ofcom requirements for speed and ease of query and issue resolution. Developers play a key role in ensuring they and WAC satisfy these quality support requirements.

Quality support helps everyone – please be responsive when your customer needs help.