
django-authorizenet Documentation

Release 2.0

Andrii Kurinnyi

Nov 13, 2017

Contents

1 Usage	3
1.1 Installation	3
1.2 Quickstart	3
2 CIM Usage	5
2.1 Using built-in models	5
3 Indices and tables	7

Contents:

1.1 Installation

Install from PyPI:

```
$ pip install django-authorizenet
```

1.2 Quickstart

Add `authorizenet` to `INSTALLED_APPS` in your settings file:

```
INSTALLED_APPS = (  
    ...  
    'authorizenet',  
)
```

The following settings are required:

```
AUTHNET_DEBUG = True  
  
AUTHNET_LOGIN_ID = "yOuRl0g1nID"  
  
AUTHNET_TRANSACTION_KEY = "Tr4n5aCti0nK3y"
```


The easiest way to use the CIM support is to use the `CustomerProfile` and `CustomerPaymentProfile` models provided by the `authorizenet` app. These models map the ORM CRUD operations to Authorize.NET calls, making it easy to keep your local and remote data in sync.

Customer profiles contain a one-to-one field `customer` which links to the Django user model by default. This foreign key target may be customized in the `CUSTOMER_MODEL` setting in your settings module.

2.1 Using built-in models

2.1.1 CustomerPaymentProfile Model

When the `save()` method is called on a `CustomerPaymentProfile` instance, the payment profile is created or update on Authorize.NET and saved to the database. A `CustomerProfile` will also be created if the specified `customer` doesn't have one yet.

When the `delete()` method is called on a `CustomerPaymentProfile` instance, the payment profile is deleted on Authorize.NET and deleted from the database.

2.1.2 Payment Profile Form

The `CustomerPaymentForm` available in `authorizenet.forms` allows a `CustomerPaymentProfile` to be easily created or updated for a given customer. This form is just a model form for the `CustomerPaymentProfile` model.

2.1.3 Generic Views

The `PaymentProfileCreateView` and `PaymentProfileUpdateView` allow `CustomerPaymentProfile` instances can be created and updated with ease. The `customer` argument sent to `CustomerPaymentForm` defaults to the currently authenticated user.

CHAPTER 3

Indices and tables

- `genindex`
- `modindex`
- `search`