

USER MANUAL

To take advantage of our library, you must be registered with Authorize.Net and obtain a test account. Thus, to get started, you are to follow these steps:

1. Open the <http://developer.authorize.net/testaccount/> page and obtain a test account.
2. If necessary, to find out which fields are to be completed (minimum requirements) for making a transaction, please download the documentation at <http://developer.authorize.net/guides/>
3. Then move on to checking out our component in your application.
4. To test the performance of the component, you can interact with a demo Authorize.Net Payment Gateway at –

Demo - <https://secure.authorize.net/gateway/transact.dll>

Production - <https://test.authorize.net/gateway/transact.dll>

5. Then create an instance of the AimGatewayManager class

```
AimGatewayManager manager = new AimGatewayManager(login, password);
```

And fill out the following fields.

```
manager.Amount = 10;  
manager.Url = AimUrl.TestApi;  
manager.CreditCard.CardNumber = "422222222222";  
manager.CreditCard.ExpDate = new DateTime(2009, 12, 10);  
manager.TransactionMethod = AimTransactionMethod.CC;
```

```
manager.SubmitTransaction();
```

To use TransactionMethod CreditCard (CC), you are to fill out the following fields:

- Url, enter manually or take from the static AimUrl class.
- CreditCard.CardNumber
- CreditCard.ExpDate.

If the TransactionMethod is ECHECK (electronic check), fill out:

- ECheck.BankAbaCode,
- ECheck.BankAccountName,
- ECheck.BankAcctType,
- ECheck.BankCheckNumber,
- ECheck.BankName,
- ECheck.EcheckType
- ECheck.BankAccountNumber - Required only when EcheckType is ARC or BOC

To save more order details (who paid what), you can take advantage of the Itemized order option:

```
Item item1 = new Item();  
    item1.Id = "1";  
    item1.Name = "Item name";  
    item1.PricePerUnit = 1;  
    item1.Quantity = 1;  
    item1.Taxable = false;  
    item1.Description = "Item description";  
    manager.Items.Add(item1);
```

Itemized order information is not submitted to the processor and is not currently returned with the transaction response. This information is displayed on the Transaction Detail page and in QuickBooks download file reports in the Merchant Interface

Можно сохранять billing address o customer

```
manager.Customer.BillingAddress.Address = "Customer address";  
manager.Customer.BillingAddress.City = "Customer city";  
manager.Customer.BillingAddress.Company = "Customer company";  
manager.Customer.BillingAddress.Country = "Customer country";  
manager.Customer.BillingAddress.FirstName = "Customer first name";  
manager.Customer.BillingAddress.LastName = "Customer last name";
```

```
manager.Customer.BillingAddress.State = "Customer state";  
manager.Customer.BillingAddress.Zip = "Customer zip code";
```

To take advantage of the Address Verification Service security feature, fill out the Customer.BillingAddress.Address and Customer.BillingAddress.Zip fields. For more information on AVS, please check out <http://www.authorize.net/support/Merchant/default.htm>

To use custom fields

```
manager.UserDefinedFields.Add("field key", "field value");
```

Once all the necessary fields are completed, call the SubmitTransaction() method.

The IsSuccess property shows whether or not the transaction was successful.

If Authorize Net Gateway returns an error, the necessary information is stored in the AIMResponse object.

The Id of a transaction assigned by Authorize Net Gateway is stored in the TransactionId property.