

EasyIntegrator API Secure Effortless Integration

EasyIntegrator...the easy API

The ChargeItPro EasyIntegrator API makes integration effortless. EasyIntegrator is provided at no cost and includes complete documentation and source code samples. Each of our integration partners is assigned an integration specialist to assist them throughout the development process. Assistance in testing the final integrated application is offered as well.

EasyIntegrator Secure Mode

The EasyIntegrator API provides a “secure” mode of integration. ChargeItPro Easy Integrator secures sensitive card holder data at all times during the approval process. Working together, EasyIntegrator API and ChargeItPro allow Point of Sale, Accounting, Wholesale and other order entry system developers to quickly and confidently deliver a reliable and secure payment processing solution to their customers.

Future Peripheral Support

EasyIntegrator makes supporting new hardware devices easier than ever. Devices are configured directly in ChargeItPro Setup. As new devices are added, they will automatically be available to the end user. There is no need to make changes within your application. Just pass the setup parameters to EasyIntegrator and the devices will be ready to go.

Benefits of EasyIntegrator API:

- ▣ Designed and built to PCI Data Security Standards
- ▣ Full Credit, Debit and Gift Card support
- ▣ Available in .Net DLL and Win32 OCX
- ▣ Complete programming documentation is provided at no charge
- ▣ Sample code is available for the more popular programming languages (.Net, VB6, Delphi, C++, C#)
- ▣ Each account is assigned an Integration Specialist for comprehensive support and to ensure a smooth integration
- ▣ Future proof peripheral support for card readers, PIN pads, contactless card readers and more.
- ▣ Customization services available to satisfy unique integration requirements



A typical transaction consists of five easy steps:

1. Instantiate EasyIntegrator:

```
EasyIntegrator easyIntegrator = new EasyIntegrator();
```

2. Set Communication and Other Parameters:

```
easyIntegrator.LoadSetup (“<Document><Params MachineNameIP=”127.0.0.1” Port=”9010”  
ServerID=”21162” ScreenNumber=1/></Document>”); //sets the communication and other parameters
```

3. Set Transaction Fields:

```
easyIntegrator.Clear(); //clear all transaction fields  
easyIntegrator.TransFields.AmountTotal = 65.23; //amount of the transaction  
easyIntegrator.TransFields.TransactionReference = “Inv1234”; //Invoice # of the transaction (optional)  
easyIntegrator.TransFields.Cashier = “John”; //cashier of the transaction (optional)
```

4. Call the Transaction Type to Process the Transaction

```
easyIntegrator.CreditSale(); // See documentation for other transaction types
```

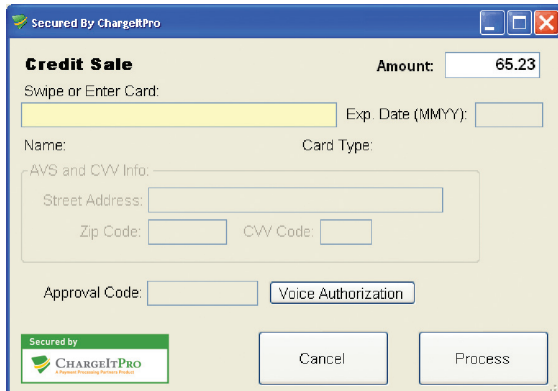


Figure 1 Screen opens to accept sensitive card holder data

5. Save and Print Results:

```
easyIntegrator.ResultsFields.AccountCardType //save and print  
easyIntegrator.ResultsFields.ApprovalNumberResult //save and print  
easyIntegrator.ResultsFields.MaskedAccount //save and print  
easyIntegrator.ResultsFields.ResultsStatus //true = approved
```