ForceGento

by TechnoMile

ForceGento on Heroku
(Magento Salesforce Connector)
Installation Guide
Version 0.0.1

Table of Contents

1. ForceGento Connect

Magento is a Enterprise level ecommerce system which has several ecommerce features like Customers management, Orders management, Inventory management, Catalog management, Coupons and promotions, Newsletters and Reports etc. whereas Salesforce is CRM and has the capabilities to manage the customer interaction, call logs, marketing campaigns etc.

ForceGento Connect is an extension that's developed under the TechnoMile Development which helps to unite the Salesforce and Magento with generic feature which are needed for any business who works on CRM and Ecommerce independently.

ForceGento Connect sync the data from Magento to SalesForce and vice versa.

2. Use Cases

- Customer has Magento site & Salesforce Org (Follow steps 3 - 9)
- Customer has Magento site but has no Salesforce Org (Follow steps 3 - 9)
- Customer has no Magento site but has Salesforce Org (Follow steps 3 - 10)
- Customer has no Magento site & Salesforce Org (Follow steps 3 - 10)

3. Magento Package Code Installation

1. Unzip the downloaded file (ForceGento_Connector-0.0.1.tgz)
2. Read the installation document present in the Docs folder.
3. Once the required zip file is downloaded and unzipped, all related files would be available in your local machine.
4. Now it is required to follow instructions 4.1-4.5 mentioned below to copy the necessary files from your local machine to your Magento instance in the server box.
    1. Navigate to **app\code\community** and Place **ForceGento** folder.
    2. Navigate to **app\design\frontend\rwd** and Place **forcegento** folder.

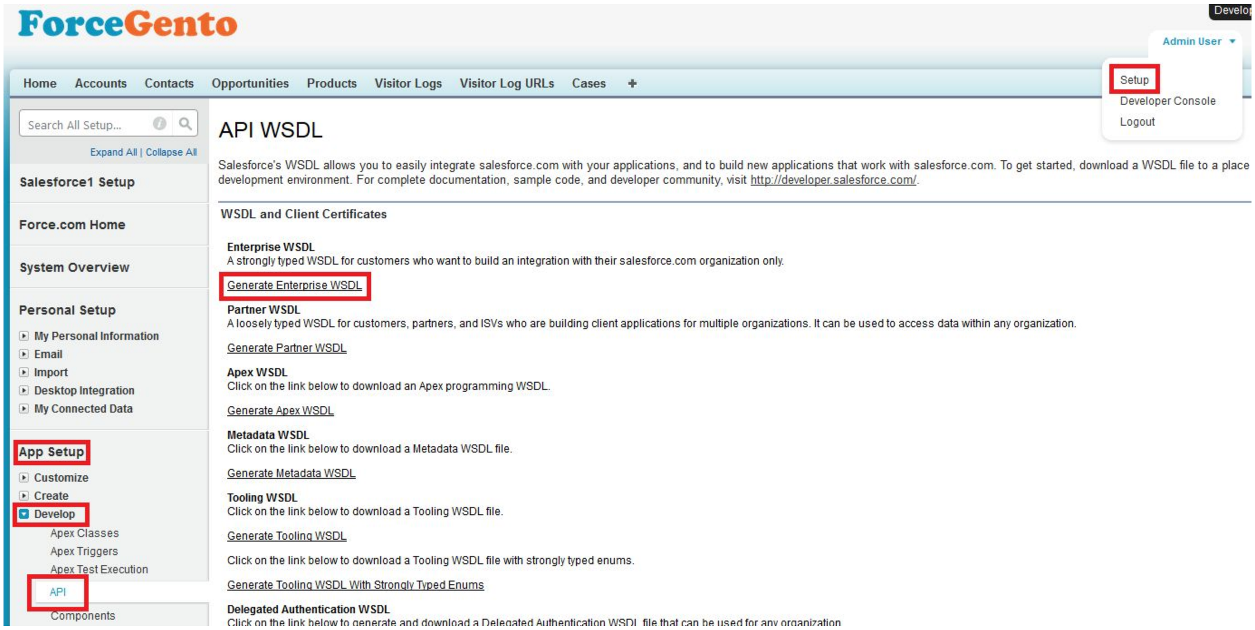
3. Navigate to **app\design\adminhtml\default\default\template** and Place **forcegento** folder.
4. Navigate to **app\etc\modules** and Place **ForceGento_Connector.xml** file.
5. Navigate to **lib** and Place **soapclient** folder.

4. Generate Token Key in Salesforce

Note: To generate enterprise WSDL file follow the below Process

Generate Enterprise WSDL file

- Go to Setup **>>** Develop **>>** API click on the link Generate Enterprise WSDL



API WSDL

Salesforce's WSDL allows you to easily integrate salesforce.com with your applications, and to build new applications that work with salesforce.com. To get started, download a WSDL file to a place development environment. For complete documentation, sample code, and developer community, visit http://developer.salesforce.com/.

WSDL and Client Certificates

Enterprise WSDL
A strongly typed WSDL for customers who want to build an integration with their salesforce.com organization only.

Generate Enterprise WSDL

Partner WSDL
A loosely typed WSDL for customers, partners, and ISVs who are building client applications for multiple organizations. It can be used to access data within any organization.

Generate Partner WSDL

Apex WSDL
Click on the link below to download an Apex programming WSDL.

Generate Apex WSDL

Metadata WSDL
Click on the link below to download a Metadata WSDL file.

Generate Metadata WSDL

Tooling WSDL
Click on the link below to download a Tooling WSDL file.

Generate Tooling WSDL

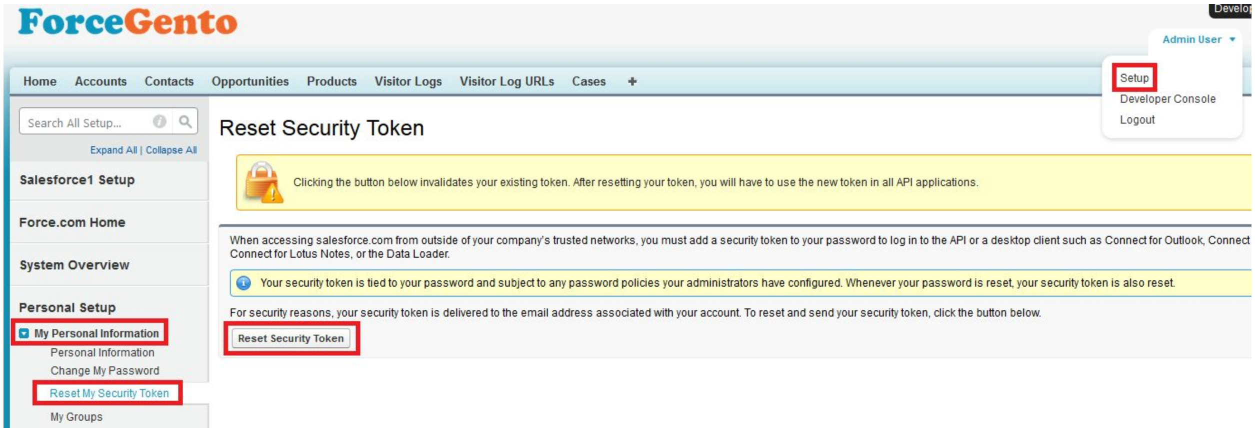
Click on the link below to download a Tooling WSDL file with strongly typed enums.

Generate Tooling WSDL With Strongly Typed Enums

Delegated Authentication WSDL
Click on the link below to generate and download a Delegated Authentication WSDL file that can be used for any organization

Fig: Generating Enterprise WSDL file

- Generate Token Key.
  Go to My Settings **>>** Personal **>>** Reset My Security Token **>>** click on the Reset Security
- Token button is shown below:



Reset Security Token

Clicking the button below invalidates your existing token. After resetting your token, you will have to use the new token in all API applications.

When accessing salesforce.com from outside of your company's trusted networks, you must add a security token to your password to log in to the API or a desktop client such as Connect for Outlook, Connect Connect for Lotus Notes, or the Data Loader.

Your security token is tied to your password and subject to any password policies your administrators have configured. Whenever your password is reset, your security token is also reset.

For security reasons, your security token is delivered to the email address associated with your account. To reset and send your security token, click the button below.

Fig: Reset Security Token

5. Install Salesforce Package

Copy link below and paste in your browser for installing Salesforce Package on your Salesforce Org.
https://login.salesforce.com/packaging/installPackage.apexp?p0=04to00000003xMk

If you don't want to our Salesforce package then create custom fields in Salesforce follow the below Process

Create a custom field Weight in Salesforce under object Products

1. Choose the field type i.e. Data Type as Text and click on Next Button.
2. Enter the mandatory field details which were highlighted with red color bar and click on Next button.
3. Again Click the Next button
4. Click on Save button

Create a custom field Backorders in Salesforce under object Products

1. Choose the field type i.e. Data Type as Picklist and click on Next Button.
2. Enter the mandatory field details which were highlighted with red color bar and click on Next button.
3. Enter values for the picklist, with each value separated by a new line. No back orders, Allow Quantity above 0, Allow Quantity below 0
4. Again Click the Next button
5. Click on Save button

Create a custom field Short Description in Salesforce under object Products

1. Choose the field type i.e. Data Type as Text and click on Next Button.
2. Enter the mandatory field details which were highlighted with red color bar and click on Next button.
3. Again Click the Next button
4. Click on Save button

Create a custom field Tax Class in Salesforce under object Products

1. Choose the field type i.e. Data Type as Picklist and click on Next Button.
2. Enter the mandatory field details which were highlighted with red color bar and click on Next button.
3. Enter values for the picklist, with each value separated by a new line. None, Taxable Goods, Shipping, Tax Exempt
4. Again Click the Next button
5. Click on Save button

Create a custom field Quantity in Salesforce under object Products

1. Choose the field type i.e. Data Type as Number and click on Next Button.
2. Enter the mandatory field details which were highlighted with red color bar and click on Next button.
3. Again Click the Next button
4. Click on Save button

Create a custom field Stock Availability in Salesforce under object Products

1. Choose the field type i.e. Data Type as Picklist and click on Next Button.
2. Enter the mandatory field details which were highlighted with red color bar and click on Next button.
3. Enter values for the picklist, with each value separated by a new line. In Stock, Out of Stock
4. Again Click the Next button
5. Click on Save button

Create a custom field Order Number in Salesforce under object Opportunity

1. If the “OrderNumber_c” custom field already exists, then increase the length to 12 by editing.
2. If not present, choose the field type i.e. Data Type as Text and click on Next Button.
3. Enter the mandatory field details which were highlighted with red color bar and click on Next button.

4. Again Click the Next button
5. Click on Save button

Create a custom field Grand Total in Salesforce under object Opportunity

1. Choose the field type i.e. Data Type as Formula (Currency) and click on Next Button.
2. Enter the mandatory field details which were highlighted with red color bar and click on Next button.
3. Again Click the Next button
4. Click on Save button

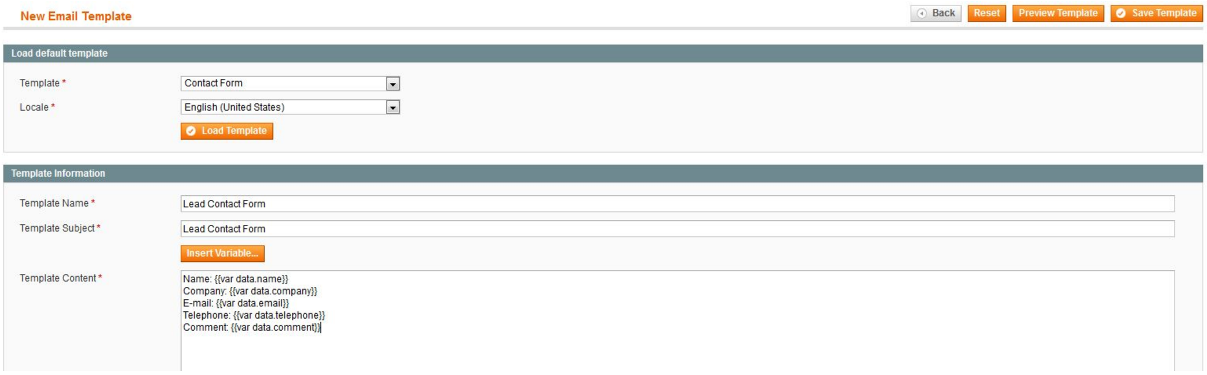
Create a custom field Shipping Amount in Salesforce under object Opportunity

1. Choose the field type i.e. Data Type as Currency(16, 2) and click on Next Button.
2. Enter the mandatory field details which were highlighted with red color bar and click on Next button.
3. Again Click the Next button
4. Click on Save button

6. Contact Us form configuration setting in Magento

Create Custom Field for Contact Us form in Magento

1. Login to Magento Admin panel.
2. Go to System >>Transactional Emails >> Add New Template
3. Select ‘Contact Form’ from Template drop down
4. Click on ‘Load Template’.



Load default template

Template * Contact Form

Locale * English (United States)

Template Information

Template Name * Lead Contact Form

Template Subject * Lead Contact Form

Template Content *

```
Name: {{var data.name}}
Company: {{var data.company}}
E-mail: {{var data.email}}
Telephone: {{var data.telephone}}
Comment: {{var data.comment}}
```

Fig: Add Contact Template

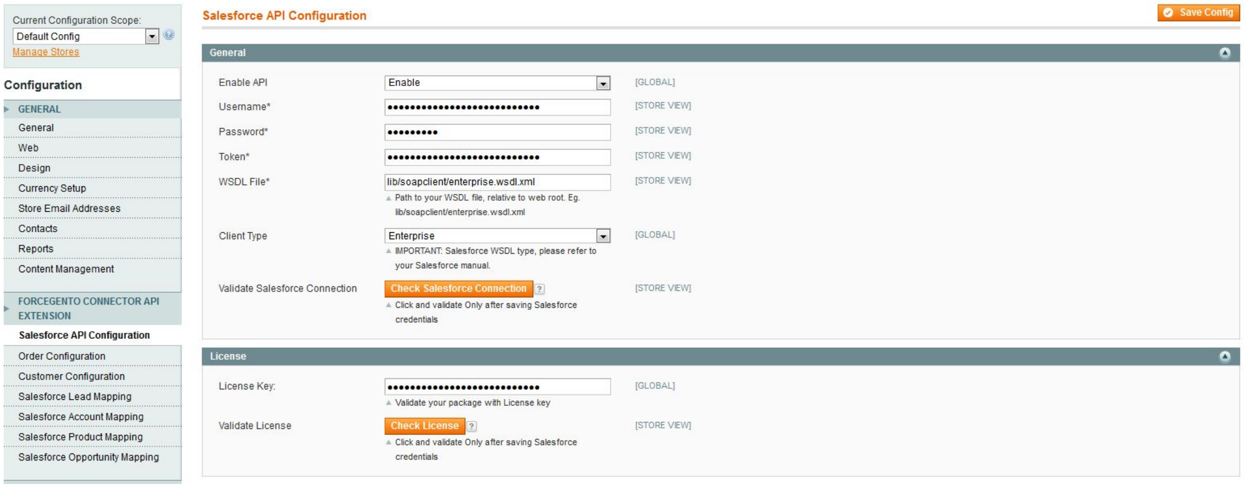
5. Add “Company: {{var data.company}}” in the template content, to add a new Company field in the form.
6. Click on ‘Save Template’.

7. Salesforce Configuration in Magento

Configure Salesforce API to allow data to sync from Magento to Salesforce

Configure Salesforce API Extension

1. Login to Magento Admin panel.
2. Go to System >> Configuration
3. In ‘ForceGento Connector API Extension’ section, click on ‘Salesforce API Configuration’.
4. Enable the API
5. Fill SalesForce username, password, token
6. Specify the path of WSDL file.
7. Select the client type (Enterprise, Partner)



Salesforce API Configuration

General

Enable API	Enable	[GLOBAL]
Username*	••••••••••••••••••••••••••	[STORE VIEW]
Password*	•••••••••	[STORE VIEW]
Token*	••••••••••••••••••••••••••	[STORE VIEW]
WSDL File*	lib/soapclient/enterprise.wsdl.xml Path to your WSDL file, relative to web root. Eg. lib/soapclient/enterprise.wsdl.xml	[STORE VIEW]
Client Type	Enterprise IMPORTANT: Salesforce WSDL type, please refer to your Salesforce manual.	[GLOBAL]
Validate Salesforce Connection	Check Salesforce Connection Click and validate Only after saving Salesforce credentials	[STORE VIEW]

License

License Key:	•••••••••••••••••••••••••• Validate your package with License key	[GLOBAL]
Validate License	Check License Click and validate Only after saving Salesforce credentials	[STORE VIEW]

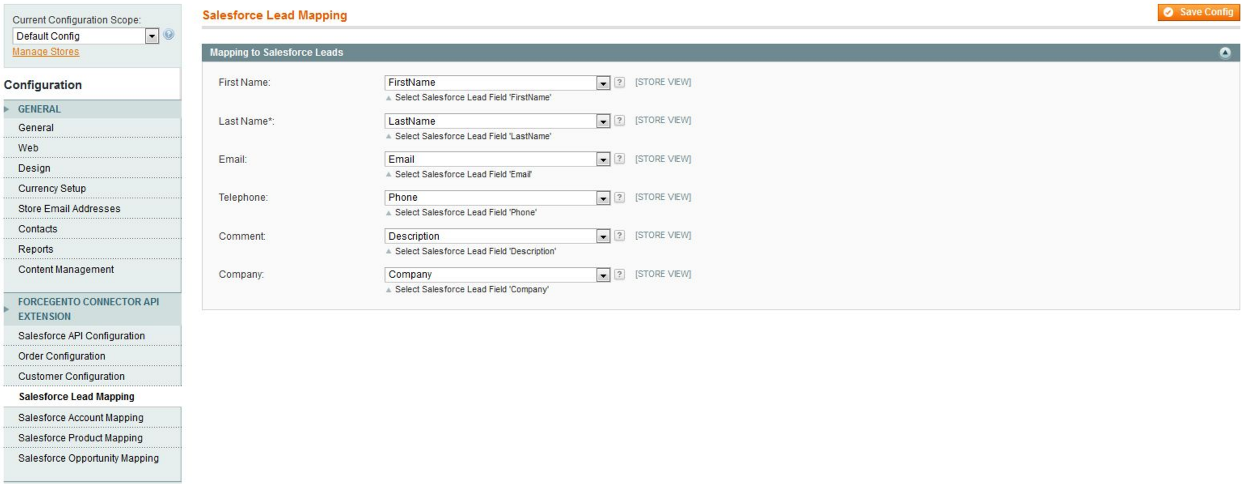
Fig: Salesforce API Configuration

8. Save the configurations.
9. This will connect the magento app to SalesForce through SOAP.
10. Add to cart product and register for the License Key from here. http://www.forcegento.com

8. Mapping Magento and Salesforce Attributes

Map Magento Contact Fields to respective Salesforce Lead Fields

1. In ‘ForceGento Connector API Extension’ section, click on ‘Salesforce Lead Mapping’.
2. Map the respective fields.



Current Configuration Scope: Default Config

Salesforce Lead Mapping

Mapping to Salesforce Leads

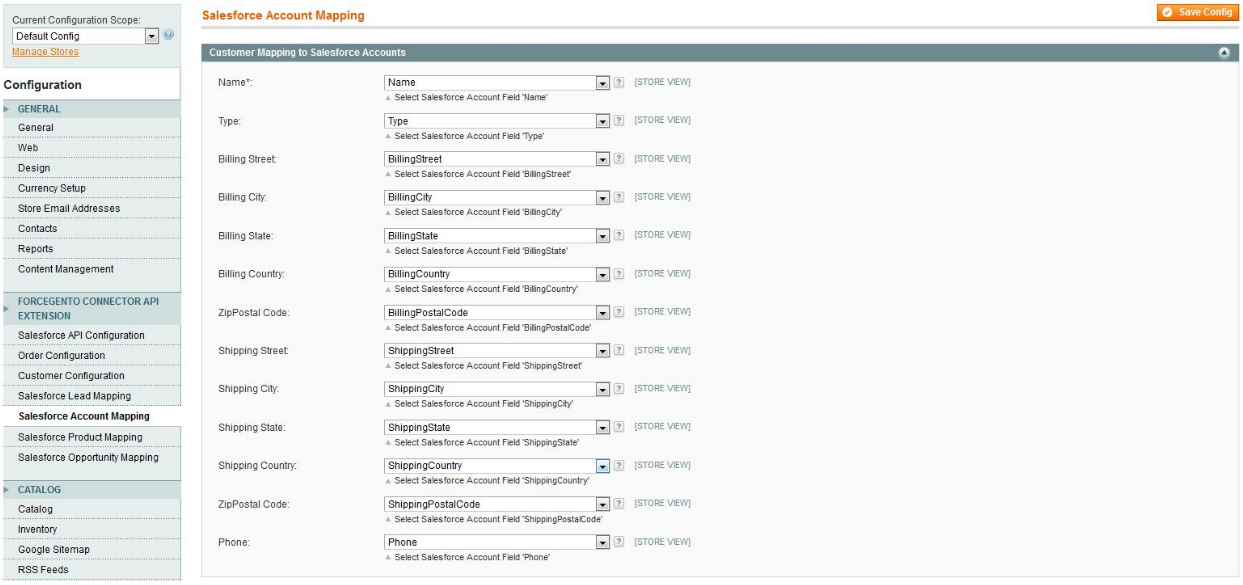
First Name:	FirstName Select Salesforce Lead Field 'FirstName'	[STORE VIEW]
Last Name*:	LastName Select Salesforce Lead Field 'LastName'	[STORE VIEW]
Email:	Email Select Salesforce Lead Field 'Email'	[STORE VIEW]
Telephone:	Phone Select Salesforce Lead Field 'Phone'	[STORE VIEW]
Comment:	Description Select Salesforce Lead Field 'Description'	[STORE VIEW]
Company:	Company Select Salesforce Lead Field 'Company'	[STORE VIEW]

Fig: Salesforce Lead Mapping

3. Click ‘Save Config’ button.

Map Magento Customer Fields to respective Salesforce Account Fields

1. In ‘ForceGento Connector API Extension’ section, click on ‘Salesforce Account Mapping’.
2. Map the respective fields.



Salesforce Account Mapping

Customer Mapping to Salesforce Accounts

Name*: Name [STORE VIEW]
Select Salesforce Account Field 'Name'

Type: Type [STORE VIEW]
Select Salesforce Account Field 'Type'

Billing Street: BillingStreet [STORE VIEW]
Select Salesforce Account Field 'BillingStreet'

Billing City: BillingCity [STORE VIEW]
Select Salesforce Account Field 'BillingCity'

Billing State: BillingState [STORE VIEW]
Select Salesforce Account Field 'BillingState'

Billing Country: BillingCountry [STORE VIEW]
Select Salesforce Account Field 'BillingCountry'

ZipPostal Code: BillingPostalCode [STORE VIEW]
Select Salesforce Account Field 'BillingPostalCode'

Shipping Street: ShippingStreet [STORE VIEW]
Select Salesforce Account Field 'ShippingStreet'

Shipping City: ShippingCity [STORE VIEW]
Select Salesforce Account Field 'ShippingCity'

Shipping State: ShippingState [STORE VIEW]
Select Salesforce Account Field 'ShippingState'

Shipping Country: ShippingCountry [STORE VIEW]
Select Salesforce Account Field 'ShippingCountry'

ZipPostal Code: ShippingPostalCode [STORE VIEW]
Select Salesforce Account Field 'ShippingPostalCode'

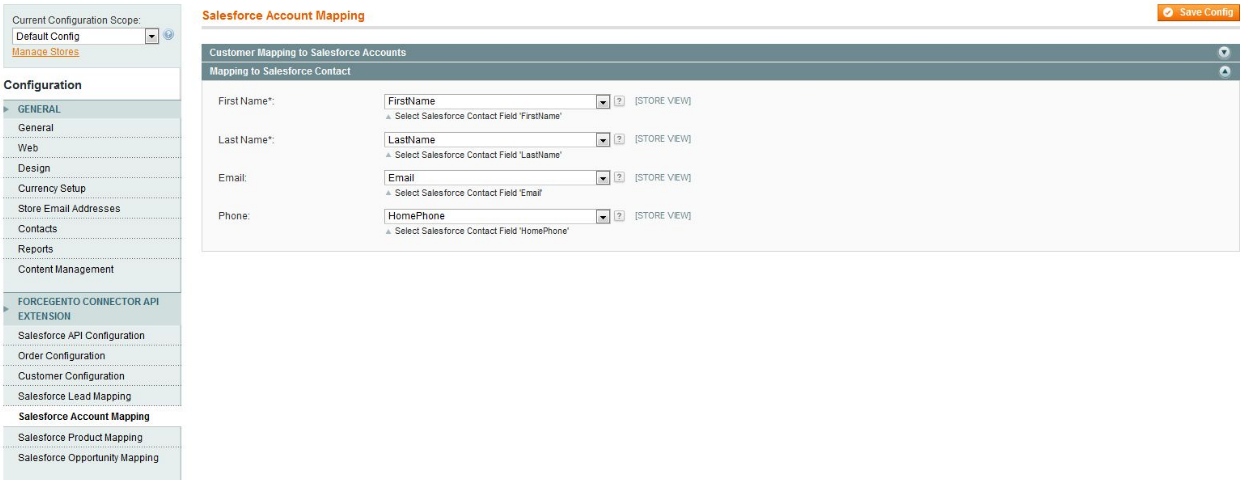
Phone: Phone [STORE VIEW]
Select Salesforce Account Field 'Phone'

Fig: Salesforce Account Mapping

3. Click 'Save Config' button.

Map Magento Customer Fields to respective Salesforce Contact Fields

1. In 'ForceGento Connector API Extension' section, click on 'Salesforce Account Mapping'.
2. Map the respective fields.



Salesforce Account Mapping

Customer Mapping to Salesforce Accounts

Mapping to Salesforce Contact

First Name*: FirstName [STORE VIEW]
Select Salesforce Contact Field 'FirstName'

Last Name*: LastName [STORE VIEW]
Select Salesforce Contact Field 'LastName'

Email: Email [STORE VIEW]
Select Salesforce Contact Field 'Email'

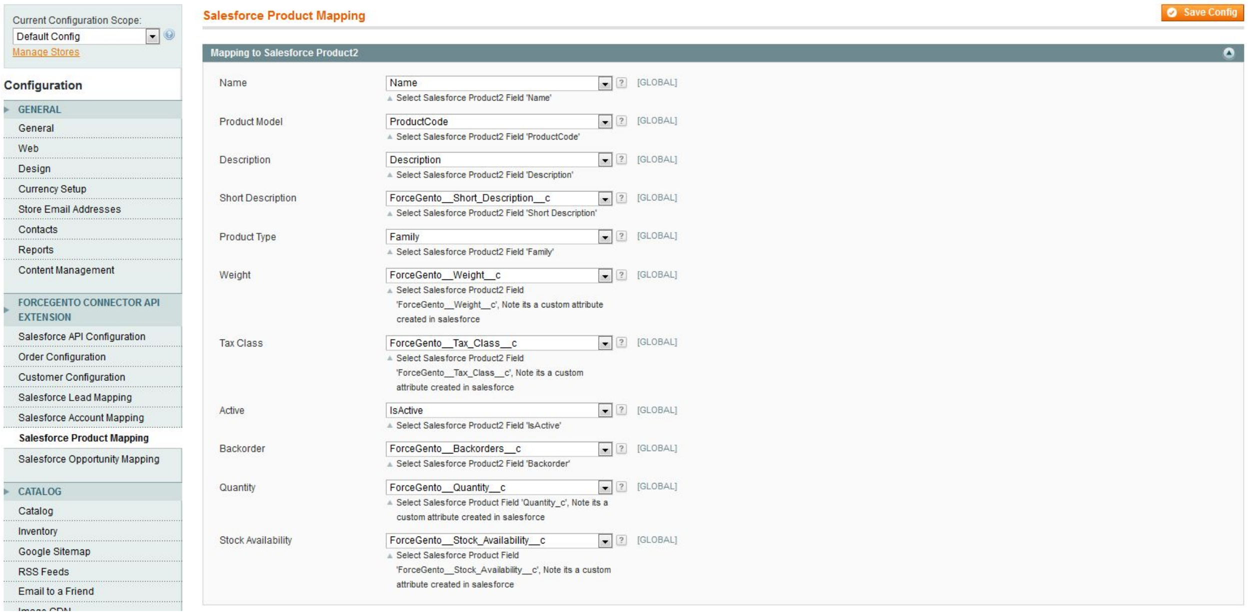
Phone: HomePhone [STORE VIEW]
Select Salesforce Contact Field 'HomePhone'

Fig: Salesforce Contact Mapping

3. Click 'Save Config' button.

Map Magento Product Fields to respective Salesforce Product Fields

1. In ‘ForceGento Connector API Extension’ section, click on ‘Salesforce Product Mapping’.
2. Map the respective fields.



Salesforce Product Mapping

Mapping to Salesforce Product2

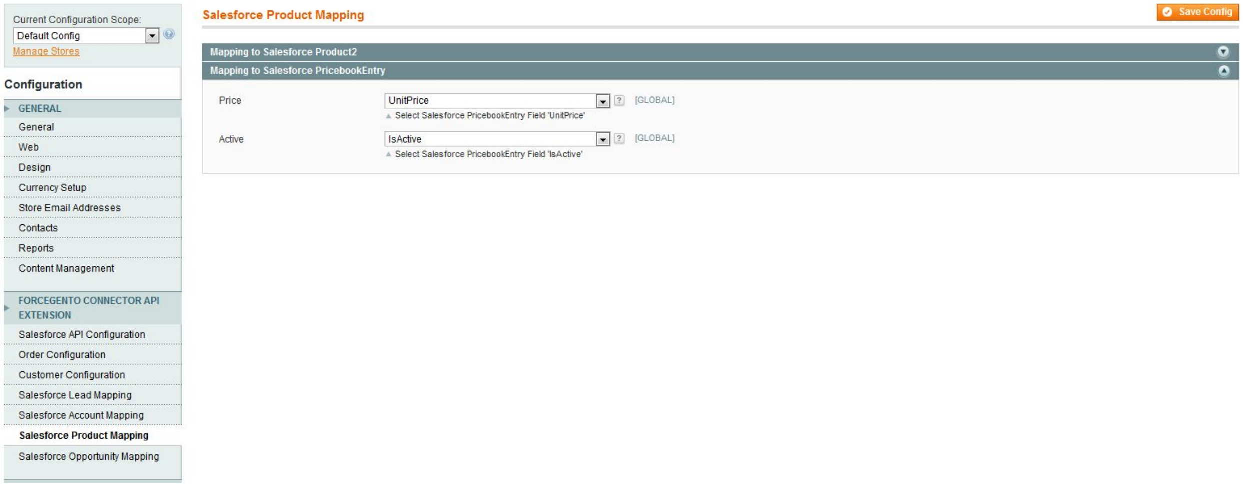
Name	Name Select Salesforce Product2 Field 'Name'	?	[GLOBAL]
Product Model	ProductCode Select Salesforce Product2 Field 'ProductCode'	?	[GLOBAL]
Description	Description Select Salesforce Product2 Field 'Description'	?	[GLOBAL]
Short Description	ForceGento__Short_Description__c Select Salesforce Product2 Field 'Short Description'	?	[GLOBAL]
Product Type	Family Select Salesforce Product2 Field 'Family'	?	[GLOBAL]
Weight	ForceGento__Weight__c Select Salesforce Product2 Field 'ForceGento__Weight__c', Note its a custom attribute created in salesforce	?	[GLOBAL]
Tax Class	ForceGento__Tax_Class__c Select Salesforce Product2 Field 'ForceGento__Tax_Class__c', Note its a custom attribute created in salesforce	?	[GLOBAL]
Active	IsActive Select Salesforce Product2 Field 'IsActive'	?	[GLOBAL]
Backorder	ForceGento__Backorders__c Select Salesforce Product2 Field 'Backorder'	?	[GLOBAL]
Quantity	ForceGento__Quantity__c Select Salesforce Product Field 'Quantity_c', Note its a custom attribute created in salesforce	?	[GLOBAL]
Stock Availability	ForceGento__Stock_Availability__c Select Salesforce Product Field 'ForceGento__Stock_Availability__c', Note its a custom attribute created in salesforce	?	[GLOBAL]

Fig: Salesforce Product Mapping

3. Click ‘Save Config’ button.

Map Magento Product Fields to respective Salesforce PriceBook Fields

1. In ‘ForceGento Connector API Extension’ section, click on ‘Salesforce Product Mapping’.
2. Map the respective fields.



Salesforce Product Mapping

Mapping to Salesforce Product2

Mapping to Salesforce PricebookEntry

Price: UnitPrice [GLOBAL]
Select Salesforce PricebookEntry Field 'UnitPrice'

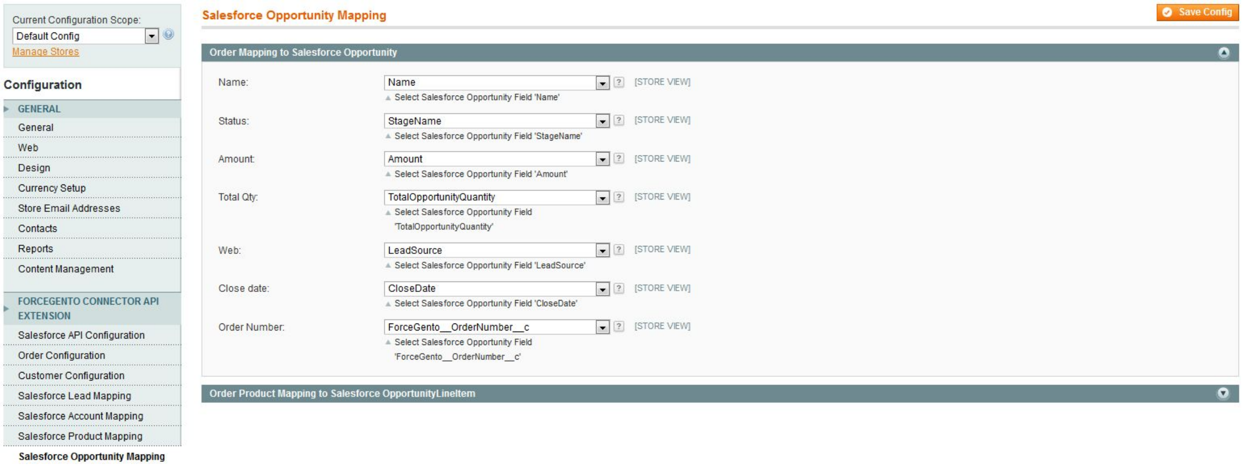
Active: IsActive [GLOBAL]
Select Salesforce PricebookEntry Field 'IsActive'

Fig: Salesforce Pricebook Entry Mapping

3. Click ‘Save Config’ button.

Map Magento Order Fields to respective Salesforce Opportunity Fields

1. In ‘ForceGento Connector API Extension’ section, click on ‘Salesforce Opportunity Mapping’.
2. Map the respective fields.



Salesforce Opportunity Mapping

Order Mapping to Salesforce Opportunity

Name: Name [STORE VIEW]
Select Salesforce Opportunity Field 'Name'

Status: StageName [STORE VIEW]
Select Salesforce Opportunity Field 'StageName'

Amount: Amount [STORE VIEW]
Select Salesforce Opportunity Field 'Amount'

Total Qty: TotalOpportunityQuantity [STORE VIEW]
Select Salesforce Opportunity Field 'TotalOpportunityQuantity'

Web: LeadSource [STORE VIEW]
Select Salesforce Opportunity Field 'LeadSource'

Close date: CloseDate [STORE VIEW]
Select Salesforce Opportunity Field 'CloseDate'

Order Number: ForceGento__OrderNumber__c [STORE VIEW]
Select Salesforce Opportunity Field 'ForceGento__OrderNumber__c'

Order Product Mapping to Salesforce OpportunityLineItem

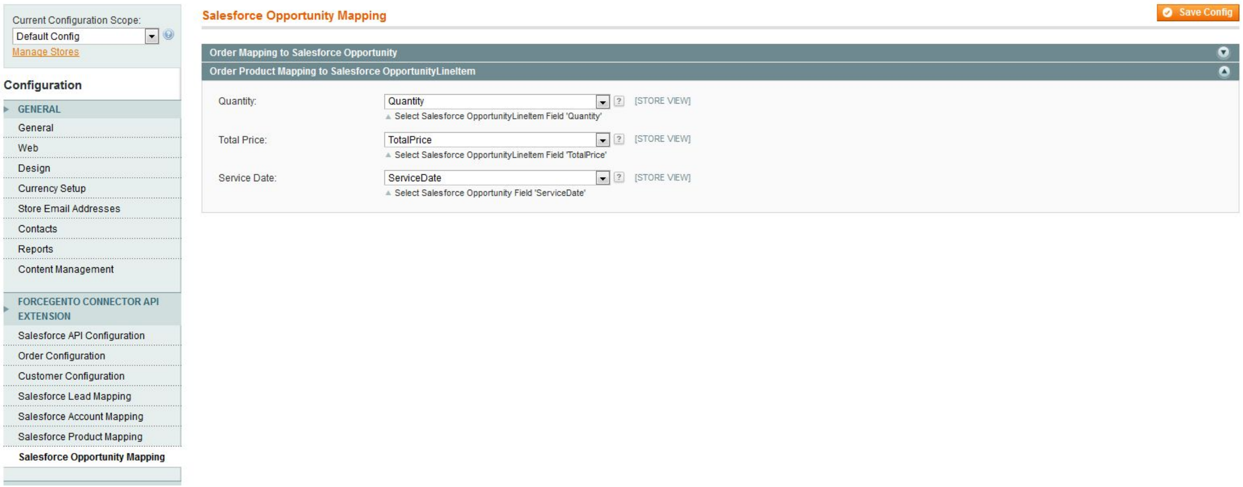
Fig: Salesforce Opportunity Mapping

3. Click ‘Save Config’ button.

Map Magento Order Fields to respective Salesforce PriceBook2 Fields

1. In ‘ForceGento Connector API Extension’ section, click on ‘Salesforce Opportunity Mapping’.

2. Map the respective fields.



Salesforce Opportunity Mapping

Order Mapping to Salesforce Opportunity

Order Product Mapping to Salesforce OpportunityLineItem

Quantity: Quantity [STORE VIEW]
Select Salesforce OpportunityLineItem Field 'Quantity'

Total Price: TotalPrice [STORE VIEW]
Select Salesforce OpportunityLineItem Field 'TotalPrice'

Service Date: ServiceDate [STORE VIEW]
Select Salesforce Opportunity Field 'ServiceDate'

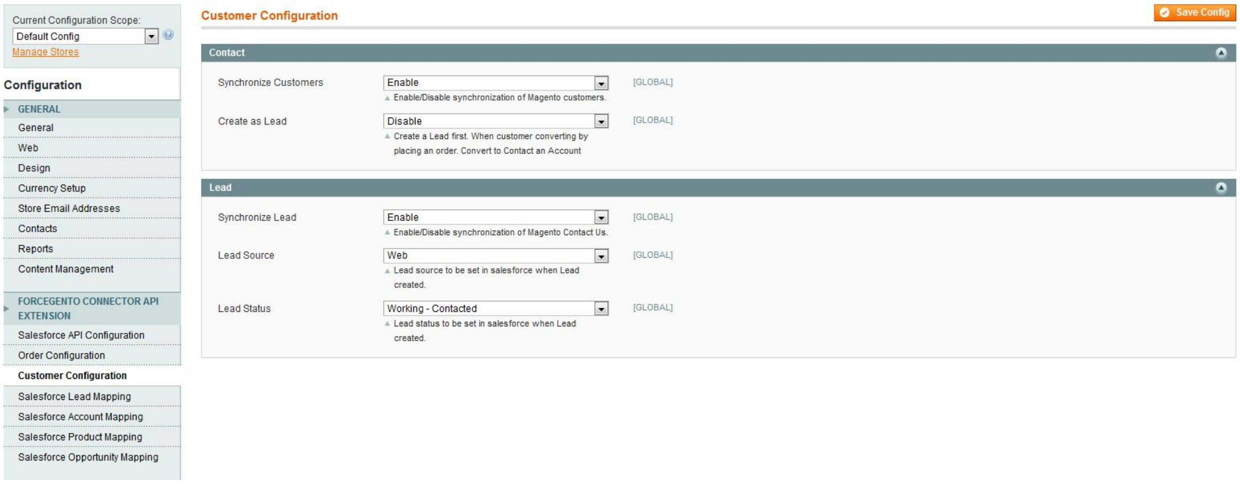
Fig: Salesforce Opportunity Line Item Mapping

3. Click ‘Save Config’ button.

9. Sync Process

Customer Configuration in Magento

1. In ‘ForceGento Connector API Extension’ section, click on Customer Configuration’.
2. Select the respective fields.



Customer Configuration

Contact

Synchronize Customers: Enable [GLOBAL]
Enable/Disable synchronization of Magento customers.

Create as Lead: Disable [GLOBAL]
Create a Lead first. When customer converting by placing an order. Convert to Contact an Account

Lead

Synchronize Lead: Enable [GLOBAL]
Enable/Disable synchronization of Magento Contact Us.

Lead Source: Web [GLOBAL]
Lead source to be set in salesforce when Lead created.

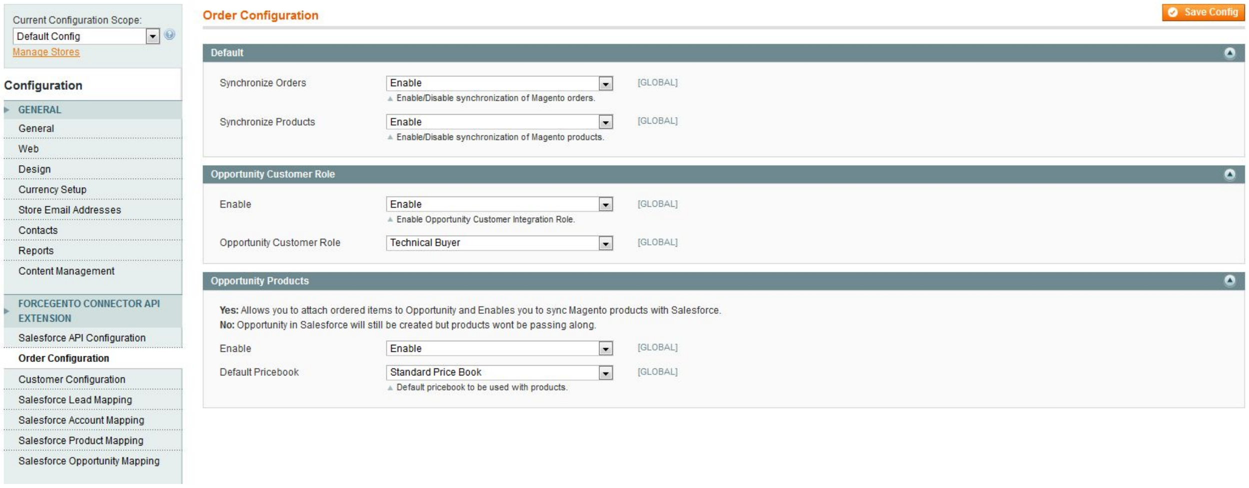
Lead Status: Working - Contacted [GLOBAL]
Lead status to be set in salesforce when Lead created.

Fig: Customer Configuration

3. Click ‘Save Config’ button.

Order Configuration in Magento

1. In ‘ForceGento Connector API Extension’ section, click on Order Configuration’.
2. Select the respective fields.



Order Configuration

Default

Synchronize Orders: Enable [GLOBAL]
Enable/Disable synchronization of Magento orders.

Synchronize Products: Enable [GLOBAL]
Enable/Disable synchronization of Magento products.

Opportunity Customer Role

Enable: Enable [GLOBAL]
Enable Opportunity Customer Integration Role.

Opportunity Customer Role: Technical Buyer [GLOBAL]

Opportunity Products

Yes: Allows you to attach ordered items to Opportunity and Enables you to sync Magento products with Salesforce.
No: Opportunity in Salesforce will still be created but products wont be passing along.

Enable: Enable [GLOBAL]

Default Pricebook: Standard Price Book [GLOBAL]
Default pricebook to be used with products.

Fig: Customer Configuration

3. Click ‘Save Config’ button.

10. Decode Source Guardian Code

- Check PHP Configuration in the server to see whether it has any Source Guardian modules or not?
- In order to check, you need to create a test file in the document root with the following script with the name **phpinfo.php**
  E.g.: http://www.domainname.com/phpinfo.php
  ```
  <? php
  phpinfo ();
  ?>
  ```
- Search for **Source Guardian** from out. If the module exists, then skip next Step else proceed with next Step
- The Source Code of ForceGento Connector extension is encrypted. In regarding to that, you need a Source Guardian Loaders to load the ForceGento Connector connect successfully.

- Open the following link to download the specific Source Guardian Loader & follow the instructions http://www.sourceguardian.com/ixeds/download.php
- In order to detect the required loader and know how to install it, please enter the phpinfo link to the phpinfo () page that is running on your server. E.g.: http://www.domainname.com/phpinfo.php
- Download the required loader ixed.xx.lin on the Server
- Move the downloaded file ixed.xx.lin to the location mentioned in the steps given by Source Guardian
- Open your php.ini file from the location
- Restart the web server

11. Heroku Configuration

- Sign up https://signup.heroku.com/
- Install the Heroku Toolbelt https://toolbelt.heroku.com/
- Login https://id.heroku.com/login
- Go to link below and click on Deploy to heroku for installing Magento Community Version 1.9.0.1 http://www.technomile.com/capabilities/application-development/heroku/magento-salesforce-integration

Heroku Setup on Local Machine

Run the following commands using git bash

1. heroku login
2. heroku keys:add
3. git config --global user.email "*YOUR_EMAIL_ADDRESS*"
4. git config --global user.name "*YOUR_NAME*"
5. git clone git@heroku.com:*NAME_OF_YOUR_HEROKU_APP*.git (ex: git clone git@heroku.com:heroku-magento-salesforce.git)
6. cd *NAME_OF_YOUR_HEROKU_APP*
7. Type this command for database details *heroku config*
8. CLEARDB_DATABASE_URL: mysql://USERNAME:PASSWORD@HOSTNAME/DATABASE_NAME?reconnect=true

Magento Database Configuration

Find a xml file at this location *app/etc/local.xml.sample* in your *HEROKU_APP* and rename this file to local.xml.Next need to write database credentials and current date in this local.xml file as below.

1. Current date format like:<date><![CDATA[*Tue, 16 Jun 2015 06:29:23 +0000*]]></date>
2. <host><![CDATA[*YOUR_DATABASE_HOST_NAME*]]></host>
3. <username><![CDATA[*YOUR_DATABASE_USERNAME*]]></username>
4. <password><![CDATA[*YOUR_DATABASE_PASSWORD*]]></password>
5. <dbname><![CDATA[*YOUR_DATABASE_DATABASE_NAME*]]></dbname>

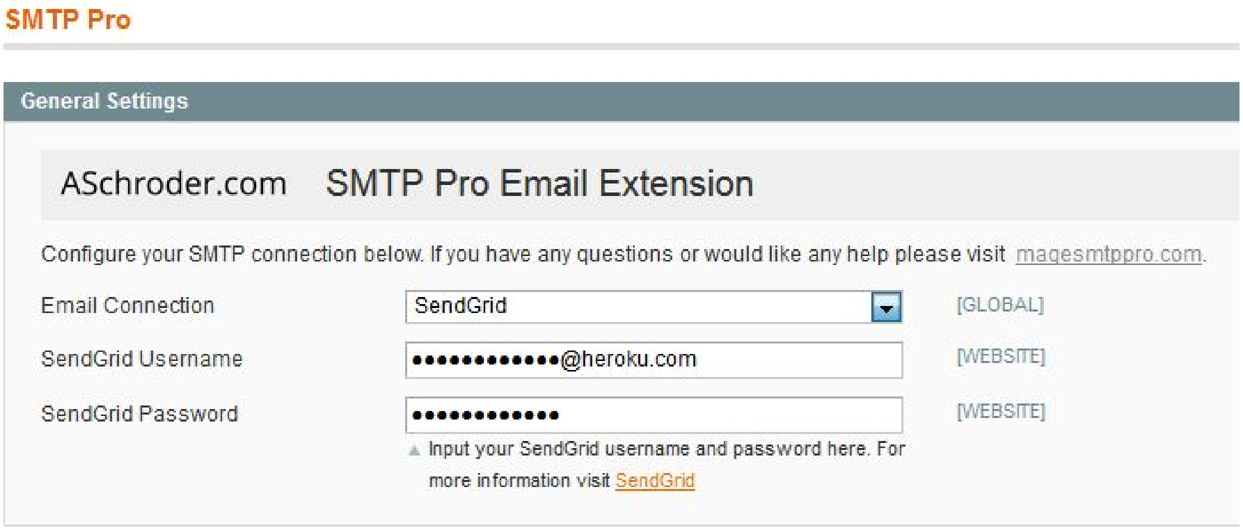
Deploy code on Server.

Run following commands using git bash.

1. git init
2. git add .
3. git commit -m *"YOUR_COMMIT_MESSAGE"*
4. git push heroku master

Update credentials for the SendGrid and in Magento.

1. Login to Magento Admin panel.
2. Goto System **>>** Configuration



SMTP Pro

General Settings

ASchroder.com SMTP Pro Email Extension

Configure your SMTP connection below. If you have any questions or would like any help please visit magesmtppro.com.

Email Connection: SendGrid [GLOBAL]

SendGrid Username: ●●●●●●●●●●●●●@heroku.com [WEBSITE]

SendGrid Password: ●●●●●●●●●●●●● [WEBSITE]

▲ Input your SendGrid username and password here. For more information visit SendGrid

Fig: SMTP Setup

3. Select Email Connection as 'SendGrid' and add SendGrid Username and Password.
4. Click on ‘Save Template’.

12. Uninstall ForceGento Connect

- Login into Magento Admin
- Go to System **>>** Configuration **>>** TechnoMile Salesforce **>>** Extension Options.
- Select No from the drop down and Click on Save Config. Or
- Go to FTP Client.
- Go to Magento Folders.
- Follow the path app\etc\modules and select file TechnoMile_Salesforce.xml & Right Click the file to EDIT.
- Press Ctrl+F and change TRUE with FALSE
- Save the TechnoMile_Salesforce.xml.