Umbraco 4.7

How to Umbraco

“Understanding complex Umbraco CMS structure
Step by Step guide”

By Jivan Thapa

Please Note: Proofreading not done yet

Table of Contents

Objectives:

In “how to umbraco” book contains basic information about Umbraco site, using this book new developer will be able to develop Web site using Umbraco from scratch. Most of case I tried to explain using examples rather than theory. “how to Umbraco” will show the way to new developer so that new developer will be able to start building complex website using Umbraco CMS. Main objectives of “how to Umbraco” is to generalizing complex structure of Umbraco CMS that new developer can easily understand relationship between different parts of Umbraco. Understanding connection between content, template and DocumentType are mostly difficult and hardest part for new developer. I am trying to explain using examples in simple word about connection between content, templates and document Type to make understandable to new developer. Installation and extending is also hardest part in Umbraco for new developer. Providing step by set guide will help to install and extend Umbraco is core concept of this book. As a whole “how to Umbraco” book is for new developer to understand basic concept of Umbraco CMS and extending Umbraco.

Inspiration:

Without reasons I did not attend to write “how to Umbraco” book. One reason is writing and publishing this book is I faced so many difficulties to learn Umbraco Site because of its complexity, finding materials to learn and lack of examples on official documentation another reason is personal interest to share knowledge. Students studying at Erhvervsakademiet København Nord Lyngby, Demark are inspiration for writing this book. Teacher **Søren Hjarlvig** taught and provided material and guide about Umbraco and **Tue Becher** taught IT security helped me to implement security topics in “how to Umbraco” book. Creuna Digital Service Agency provided chance to sharp my Umbraco knowledge is also account for inspiration. All teachers at KNORD who helped me to learn programming are inspirations.

Finally of course readers are great inspiration to writing this book.

Introduction:

Umbraco CMS is one of the CMS systems that based on asp.net which is Microsoft platform. Many developers who are familiar with one of the Microsoft platform Dot.Net languages they are interested to work with Umbraco CMS. Official web sites of Umbraco CMS claims “Umbraco CMS currently powers more than 85.000 websites worldwide”[1] which is huge numbers of websites and according to Wikipedia “1000 downloads a day and top most popular downloads via the Microsoft Web Platform Installer” [2].

I am working with Umbraco since one and half year and had chance to work with big web project based on Umbraco CMS while I was doing my internship at Creuna.dk. When I worked with Umbraco I feel bad sometimes and good most of time. I feel bad in that sense it require lots of pre settings in order to install and set up first time as compared to other CMS like Joomla and drupal. I feel good in that sense when I finished installation and setup it worked perfectly without any problem and have great functionality, flexibility and capable to support multiple programming language.

According to first paragraph and my one year of personal experience made me to conclude that Umbraco CMS is growing CMS system and will continue to grow up every year and my first choice of CMS is Umbraco. I stared to learn Umbraco from scratch before one year from school (KNORD) and still learning about Umbraco CMS. During my learning process I got very hard time to learn Umbraco CMS and still getting hard time to learn. One reason was that there was not cooked book publish[3] yet about Umbraco CMS and Second reason is documentation is not enough in Umbraco official site. One reason for having not enough documentation about Umbraco could be Umbraco CMS is open source so that Creator of Umbraco CMS wants to force developers or users to pay for official published video tutorials that’ require to pay. I was planning to provide general information and instruction material about Umbraco CMS since long time ago for beginners who want to develop his carrier as Umbraco Specialist in future.

I already started to share my knowledge about Umbraco CMS using video tutorials in YouTube social network sites. www.youtube.com/jeevan2np is my YouTube video channel. Every week number of visitors is increasing in my videos and some visitors also commented on videos that made me happy and provided energy to look more about Umbraco CMS. All videos I had uploaded to YouTube are not professional some videos was raw video, too little information, very basic, not much about programming and little much mess with information. It was mess because I did not organize message properly on videos. I created videos about “how to install Umbraco CMS in localhost” before three month ago and have 295 views (based on date May 30, 2011)that also provided me energy to look more about Umbraco CMS.

I created that video with my friends “Niraj Bana” as my final exam project for Advanced Media Technology projects.

Now I am trying to provide more academic knowledge about Umbraco CMS. Going deep inside Umbraco CMS and provide basic core concept of Umbraco CMS will help a lot to new developer who want to build carrier as Umbraco Specialist in future. This Umbraco Basic “How to Umbraco” will not provide knowledge that makes readers as Umbraco Expert rather than it will show the way by providing concept in different areas of Umbraco CMS, providing guidance to developers. So after reading a digital book, developers will be able to figure out basic concept of Umbraco CMS and develop skills and knowledge to build websites in Umbraco CMS.

Reading this book will provide general over view of Umbraco CMS, how Umbraco CMS works, what are the extension possibility of Umbraco CMS if it needs to extend. In this book I am looking into Umbraco API and extending Umbraco CMS in brief. I will cover basic concept of different parts of Umbraco like templates, macro and Content, 40% of this book is focused on programming for Umbraco CMS. By providing general programming knowledge of Umbraco CMS will help readers to be expert in Umbraco CMS in future and such general knowledge will work as pillar from where new developer can start to build big house.

References: References date: 04/07/2011

1: **http://umbraco.com/get-started/case-studies/sites-running-umbraco**

Extra link about Umbraco: **http://labs.thesedays.com/2010/07/30/why-umbraco-suits-us/**

2: **http://en.wikipedia.org/wiki/Umbraco**

3: until references date mention above.

Validity or trustworthy of References:

- Article was modified by Umbraco developers and most of are either Umbraco core developer team or freelance and Umbraco specialist have experience since 4 years. Most of have own blogs about Umbraco. Example Lee Kelleher http://blog.leekelleher.com/author/vertino/
- People are discussion about why Umbraco “extra link above”. Somehow they are represents real voice coming from developer so there is place to trust.
- http://www.cmsmatrix.org/ is one organization that provides comparison of different web site CMS. It is also possible to compare more than two CMS at a time.
- http://www.cmsmatrix.org/ has list of almost all available CMS. Information may not reliable for example Umbraco is Free CMS but it shows 30Euro + VAT per domain. We can use this site as to get ideas but should not be use as guarantee or 100% true information.

Scope:

In this “How to Umbraco” book I will look into major parts of Umbraco CMS and provide knowledge of its core concept. Here is will address these issues.

1) **Umbraco requirements**
2) **Installation**
3) **Umbraco Core Concept**
   - Developer will understand, its core concept.
4) **Document Types:**
   - Understanding of Create, use, apply Document Types and its API
5) **Templates**
   - Understand concept of templates, create, render content, extending templates using Macros and xslt.
6) **Macros**
   - Concept of macros and use it.
7) **XSLT**
   - Overview of xslt, use it, extend xslt using .net class
8) **Content**
   - Create content based on DocumentType, save, publish and delete
9) **User Control:**
10) **Database**
    - Insert data into Database
    - Fetch data from Database

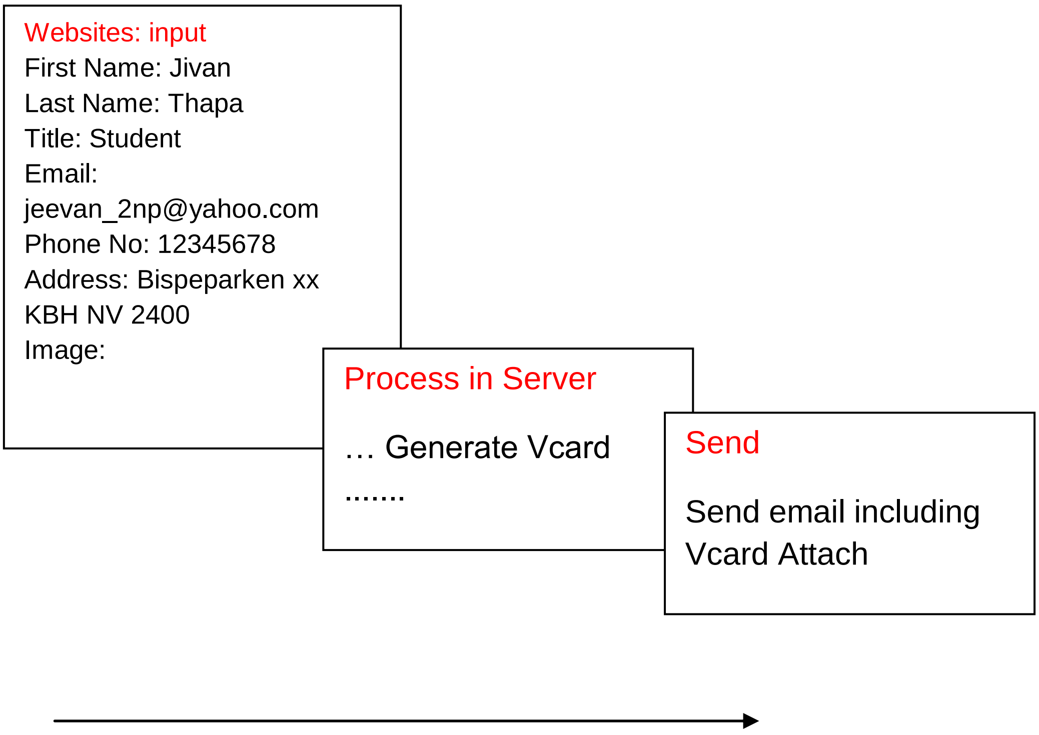
These are list of my focused area. It does not mean that it covers or represents whole Umbraco CMS. Umbraco is very huge. It’s only covers basic Umbraco. basic Umbraco can be varying from developer to developer. In my assumption by having knowledge in above, understanding, using it will be enough to make simple websites that I am going to develop along with this book as a case “online Vcard generator”.

Online Vcard Generator website as a CASE

Vcard is one part of Microformat and semantic web. In one sentence semantic web make websites or web content for human first and machine second. Using Vcard we can make it possible to read our content by human so machine will not only read content as binary format or characters it can understand like people, address, age, email etc.
I use Vcard generator as a case to demonstrate Umbraco Basic, it's a good practice for developer to make semantic web.

Scope of Vcard generator:

It will be basic but will contain core concept. By providing personal information like name, email, address, phone and image. It will generate Vcard based on personal information and will received attached Email. See visual diagram bellow Vcard can easily read by many applications for example Firefox browser extension "Operator", Microsoft Outlook and other mobile devices.



Websites: input
First Name: Jivan
Last Name: Thapa
Title: Student
Email:
jeevan_2np@yahoo.com
Phone No: 12345678
Address: Bispeparken xx
KBH NV 2400
Image:

Process in Server

… Generate Vcard
…….

Send

Send email including
Vcard Attach

What I excluded from.

There are many important things which are also part of while project, but I decided not to include here. Reason is these are not focused topic of this book.

Visual parts:

- Visual designs of websites are very important because these represent whole company, its identity, goal and target group. Without visual design it will be hard to communicate with target groups and marketing purpose.
- Web design process is not single process there are many steps for example concept, scratch, web grid, prototype design, testing etc.
- Even though these processes are important I decided not to include here because of not focused topic of this book.

Programming languages:

- In this Umbraco CMS basic book I use C# language. I am not going to explain my code in depth here so reader should be able to understand C# code and use it when it necessary. For example Class, static class, abstract class, modifier, methods, References, Web services, operators, logic, type and namespaces.
- Database is very important to store data. I am using sql server so it is necessary to know about basic sql server for example create account, setup permission, create database, backup and restore.

Client side programming languages:

- It may necessary to understand about CSS, Jquery, JavaScript and cross browser problem. I am using CSS, client side programming languages to present content nicely and to control more at client sides. These are also important but I will not cover because it’s an out of focused topic.

Umbraco hosting:

- Once web site is complete it is necessary to host. In my personal word I say “local to live process”. I called first local because we develop websites locally and when its works and bug fixed than I move it to live server where end users are browsing websites.
- To make Umbraco happy to run without any problem there are few requirements. Not all windows server hosting provider supports Umbraco CMS so it is also big topic. One major thing is if you are planning to buy hosting provides for Umbraco make sure hosting provides should have “plesk” features.
- I also skip these parts but I will include in brief in “requirements topics”

Comparison of different CMS System

- In a digital book, I did not mention about Umbraco CMS and others CMS. For example why do we need to choose Umbraco CMS? What are the advantages and disadvantages of using Umbraco CMS as compared to other CMS? Developer do not chose CMS accidently to build professional or clients web sites so almost all developers made comparison between different CMS system based on their or/and clients requirements before they go to use it. In general, how to choose right CMS for your web sites? Can be discuss as new whole topics and can be very big task if we go more dipper about choosing right CMS among 100 systems, its need to have concrete ideas in order to choose right CMS. I did not include in this report about choosing right CMS because my understanding or assumption is that you as a readers already decided to go with Umbraco CMS and ready to build websites using Umbraco CMS.

Search Engine Optimization:

- Million of websites exist in world. If nobody uses your websites than what’s a meaning of existing it? How people will find your websites among millions?
- Optimizing websites for search engine will help to find users about your websites. This can be very huge topic but I skipped here because of not focused topic in this project.

Umbraco different section:

- There are many different section or categories inside Umbraco CMS for example
- Language: this is used to make your site in more than one language.
- Cache: performances are always accountable. Umbraco Cache section will provide cache items to better performance of websites.
- Packages: this section contains list of installed packages, uninstall packages remove packages and create packages. Package is like “extension”, plug-in etc.

How I Structured Book.

Every people organized their stuff on their own style because it's a generation of customization. We customize things to fit our needs and satisfaction. When we customize things sometimes it makes difficult for other to understand. I am using some word here that can be understood in different meanings so here I explain what does it mean that if I use some common words.

Frontend: it's all about visual presentation of content to end users where end user makes interaction and involved in different transaction.

Backend: administration area where developer, Editor, Creator, Publisher and Administrator use to get access to Create, Delete, Update and Read using browser. Here browser is important means all person who have access to control panel of Websites via browser is Backend. In Umbraco backend can be found using domain name and "/Umbraco/" will redirect to backend where user must provide authentication. Example www.jeevanthapa.com/umbraco/ will redirect to Backend.

CodeInProject: This is new I created to fit in my context. Let say if I create custom class "image.cs" into visual studio project to add extra function or to extend XSLT, I say create "image.cs" in CodeInProject. Adding extra features to extend Umbraco websites using Class, Usercontrol, webservices etc are define as CodeInProject in my context and these are not possible to create using BackEnd of Umbraco sites. For example we can add .xslt file using Backend of Umbraco Sites but we can't create file like Test.cs using Backend.

Security: extending Umbraco CMS using .NET is not only to add extra feature, it will also make Umbraco CMS venerable to hackers. So I add Security topic to add extra features by explaining how to secure extension from hackers, this topic will be only included in extension of Class, usercontrols and webservices. Here I only included Cross Site Script and SQL injections.

Abbreviations:

I have used abbreviation in many places to reduce typing and spaces.

Short Form	**Real Form**
CMS	Content Management System
XSS	Cross Site Script
DocType	Document Type
Regex	Regular Expression
Temp	Template
CSS	Cascading Style Sheet
SQL	Structure Query Language
DLL / dll	Digital Link Library
URL /url	United resource locator
http /https	Hypertext Transfer Protocol / Secure
object	Umbraco object and each object have own task. For example template.
Backend / back office	Administrator control panel. Can be access using “domain name/umbraco”. It requires user id and password.
Frontend	Front end where end user sees or interact with website.

CHAPTER 01: Introduction

Introduction Umbraco CMS

Umbraco CMS is one of the CMS based on .NET languages. C# is one optional to use in Umbraco; I am also using C# here to extend Umbraco. Umbraco provides rich text editor service out of box it's a mostly common things for CMS applications. What's new thing that Umbraco can provide? Easy and fast answer is Document Type that's very important feature of Umbraco many developers are looking for that in CMS application and many developer choose Umbraco because of its out of box Document Type features. That is exactly same as Content Construction Kit if we compare between Drupal and Umbraco only difference is Drupal does not have out of box Content Construction Kit feature. Document Type allows developer to add custom fields to nodes using a browser. Example can be found in "Document Type" topic.

Extending Umbraco can be done using many languages for example we can use XSLT which use Xpath languages to move cursor back and forth, can also use Python languages, Can also use .NET user control and .NET custom control. Database can be use Mysql or Microsoft base database like mssql.

Presenting content in Umbraco CMS can be complex for first time users so I will focus more on it. Templates are use to present content to end users. CSS, JavaScript, Jquery can be use in Templates without any pain.

Running multiple websites on one Umbraco installation, we can host more than one websites using single Umbraco CMS installation means Umbraco CMS can be used to create multiple websites and no need to install each time for each websites. One installation will be enough to lunch multiple sites.

Umbraco generates XML file and those files contains all published items in websites. Using xslt or .NET custom class we can fetch content and render to templates to present content as HTML format, Umbraco is amazing software and fun to work with Umbraco. Let's get start.

How Umbraco Works?

Core concept of Umbraco and its working principle are unique id and alias. Each object inside Umbraco CMS has unique identification and defined as NodeId. In general Umbraco CMS has its rules to crate NodeId. NodeId is defined as Integer in technical term and in general term it's a numbers like 1234 and can be also negative. All NodeId start with negative value is out of box items and assign by Umbraco in installation period or it's build in value.

Node Id:

When we add object in Umbraco site for example document type, templates or content, article or even when we upload Image, files, and video etc will assign value by Umbraco as Node Id, until now node id is based on four digit numbers like xxxx and it's a unique and never repeated node id. Using node id we can easily control object, we can fetch object based on node id and perform additional work on that object. Note that NodeId will not change when we update object.

Alias:

Alias is use with names. When we create object we provide name. Name is for the editor or for human. Alias is used to identify that object by programmer or by machine (Umbraco System). Umbraco do not recommend using space and symbols in alias.

Let's take example and be clear about Alias. If we create Macro name "Vcard Render" and alias "vcardRener". Assume that "Vcard Render" macro will display input field like textbox for users to type name, email and other personal information. We need to use that macro in Templates to display textbox field because Umbraco only use templates to present content to end users. So we need to insert that macro into Templates in order to display input fields. In template we insert macro's alias; on runtime Umbraco will automatically insert content on templates to end users. This is very basic how Umbraco alias works.

Basic Requirements:

Here I listed basic requirements that need to run and extend Umbraco cms.

Server: IIS 7 or later version. You can download IIS7 or newer version in your windows 7 Machine and install to work with Umbraco locally. If you want to go live you have to buy server or windows web hosting where you can upload final Umbraco files.

Umbraco needs special file setup permission so when you buy server make sure they support Umbraco installation or ask if they have IT support to install Umbraco and made decision. Based on my experience never buy web spaces before considering several questions for example

- Does hosting provider have windows server?
- Does hosting provider have "Plesk" features; plesk is used to set up file permission?
- Is mssql or mysql database available?

  If answer is "Yes" considers it is green signal to buy it.

Database: Umbraco cms need database to store data because its better way to mange content and Umbraco also needs database to store node id, alias or other object except files. Files are located at server hard drive. Umbraco is user friendly so you can choose mssql or mysql database. It supports any one of two databases and in most of case windows hosting provider have mssql database installed. I recommend using sql express edition for your local machine. You can download it from Microsoft official sites it's a free and try Umbraco locally first before deploying to live server.

Additional Software: by having windows operating system, IIS server, database will not be enough for Umbraco so needs more software to install. .NET framework version 4 and Ajax is necessary to run Umbraco. Web hosting provider already installed this software for you but you can ask them to verify it. For your local machine you have to install .NET Framework version 4 and Ajax before start Umbraco installation.

Visual Studio: It is great software ever for me as a programmer. It's very powerful IDE (integrated Development Environment) software that provides by Microsoft. It's quite expensive software and use to program all Microsoft based applications like console application, windows application, web application etc. for Umbraco cms project you can download from Microsoft as trail version and use it for 3 months. Three month will be enough to complete this project called "online Vcard" and for your later purposes you can buy it from Microsoft.

Visual studio is not necessary if you do not use .NET to extend Umbraco sites. But for my purpose I am going to show developers how to use .NET class to extend Umbraco so visual studio requires to create .NET class. If you are building basic websites Visual Studio may not be necessary because Umbraco has many features that can achieved by XSLT, XSLT can be edited inside Umbraco using BackEnd of Umbraco sites.

We can also use other open source software like notepad++ but these are simple text editor so it is very difficult to program in simple text editor. This software does not have features like visual studio.

Which visual studio?

There are three different types of visual studio:

- Express
- Professional
- Ultimate

I recommend using visual Studio 2010 if you have, but if you have 2005 or 2008 it will be enough, no need to upgrade it just for this "Online Vcard Project". If you are planning to download Visual Studio, I recommend downloading visual studio 2011 ultimate because it has great features.

Get Visual Studio Free: you can use express edition for free provided by Microsoft, there is difference between Express and Ultimate visual studio. If you want to know difference you can easily go to official site and compare different edition of Visual Studio. If you do not want to buy visual studio Ultimate after 3 month of Trail Period you can easily download Express Edition without any cost and use it for free.

References and Download
Taken date April 12, 2011
Umbraco system requirements
http://our.umbraco.org/wiki/recommendations/recommended-reading-for-it-administrators/minimum-system-requirements
Download Ajax and .net framework 4
http://www.microsoft.com/downloads/en/details.aspx?FamilyID=9cfb2d51-5ff4-4491-b0e5-b386f32c0992&displaylang=en
Ajax latest version
http://www.asp.net/ajaxlibrary/download.ashx
sql express
http://www.microsoft.com/express/Database/
Visual studio
http://www.microsoft.com/visualstudio/en-us/try
IIS server latest version
http://www.iis.net/download
Notepad ++
http://notepad-plus-plus.org/download
Download Free Express Edition
http://www.microsoft.com/express/Downloads/

Chapter 02: Installation

Installation of Umbraco CMS

There are several steps to install Umbraco CMS, most of steps are easy but needs to take carefully each steps. Here I briefly discuss about installation process. I already made video about "installation of Umbraco CMS in localhost" and uploaded to YouTube video. I strongly recommend to watch video[1] first and come back later in this chapter to continue. Here basic steps are listed.

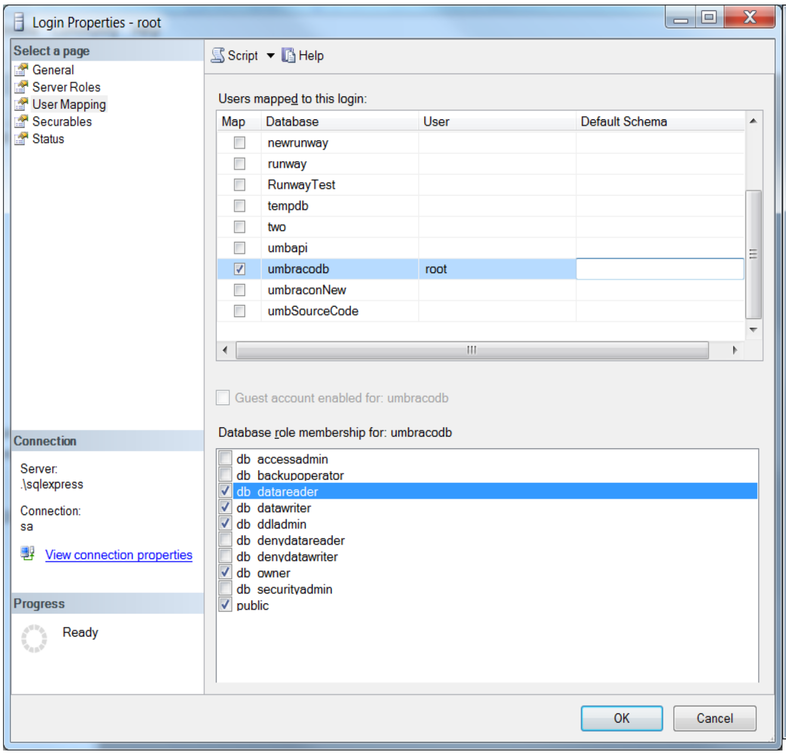
Database:

Create Empty Database, you can give any name as you like for this purpose I gave name "umbracodb" and created user account name "root". Make sure user should have database admin access otherwise Umbraco will not install properly.

Instruction:

- Open Sql Server Management studio and login as "Windows Authentication"
- Right click on Database and click on "New Database". Provide database name and click ok. *See Image: umbracodb.image*
- Expand "Security" on left side menu. Right click on Login and click on "New Login"
- Type login name and check "SQL server authentication" and provide password. Uncheck "Enforce password Policy" if you are new to Sql server and press OK. Windows will disappear. In my case I create login name "root" and password "root"
- Expand Security tree from left side menu until last child where you can see new login added *See Image: user.image* and click on that login name. it will open Login Properties
- Select "User Mapping" it will list all available database and check new database that you are going to use for Umbraco and in "Database Role Membership" checked on "db owner", "dbladmin", "datawrite" and "datareader",.
- Disconnect current working database and try to connect using new user id and password. This time do not select "windows authentication". You should be select "Sql Server Authentication" this will activate Login and Password input textbox. You must be able to login to database. If you get any error please repeat database setting process.
- If you success to log in using your new user id and password and database that assign to Umbraco is listed on right side menu than your database setup is ready for Umbraco.
- See screen shot for references

[1] http://www.youtube.com/jeevan2np#p/a/u/0/DMXw7aeSG54



Users mapped to this login:

Map	Database	User	Default Schema
□	newrunway		
□	runway		
□	RunwayTest		
□	tempdb		
□	two		
□	umbapi		
☑	umbracodb	root	
□	umbraconNew		
□	umbSourceCode		

□ Guest account enabled for: umbracodb

Database role membership for: umbracodb

- □ db_accessadmin
- □ db_backupoperator
- ☑ db_datareader
- ☑ db_datawriter
- ☑ db_ddladmin
- □ db_denydatareader
- □ db_denydatawriter
- ☑ db_owner
- □ db_securityadmin
- ☑ public

Connection

Server:
.\sqlexpress

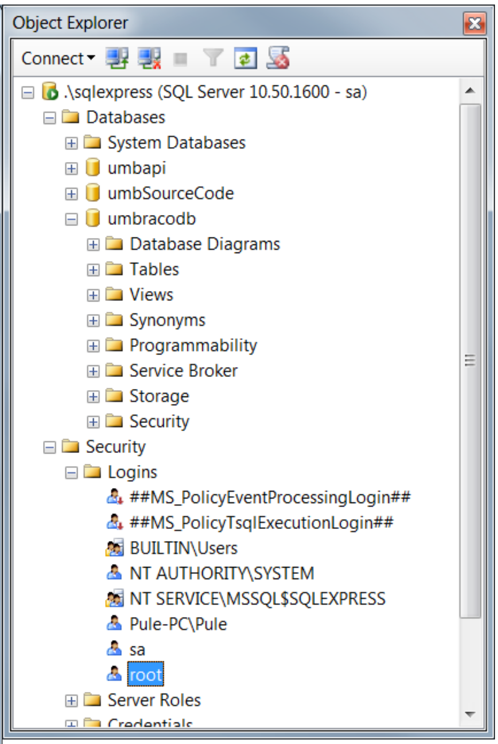
Connection:
sa

View connection properties

Progress

Ready

Umbracodb.image



User.image

Notes:

Here I did not include about MySql[2] Database. MySql server is free server and mostly used in Apache server. MySql Server is developed and maintained by Oracle Corporation. Most of PHP web application like Joomla[3] CMS, Drupal [4], Wordpress[5] CMS use MySql server.

Your case can be difference from here in that case you can installed Mysql database from official sites at www.mysql.com. When you install Mysql Server please read documentation provided on official sites and follow each step carefully otherwise MySql database may not work as expected.

Creating user, database, setting database permission for user, changing password can be difference for MySql Database as instruction provided to Mssql database. If you decided to use MySql server for Umbraco CMS it's your responsibility to find out more details about MySql set up because here I will not conver MySql database.

[2] http://www.mysql.com/
[3] www.joomla.org
[4] www.drupal.org
[5] www.wordpress.org

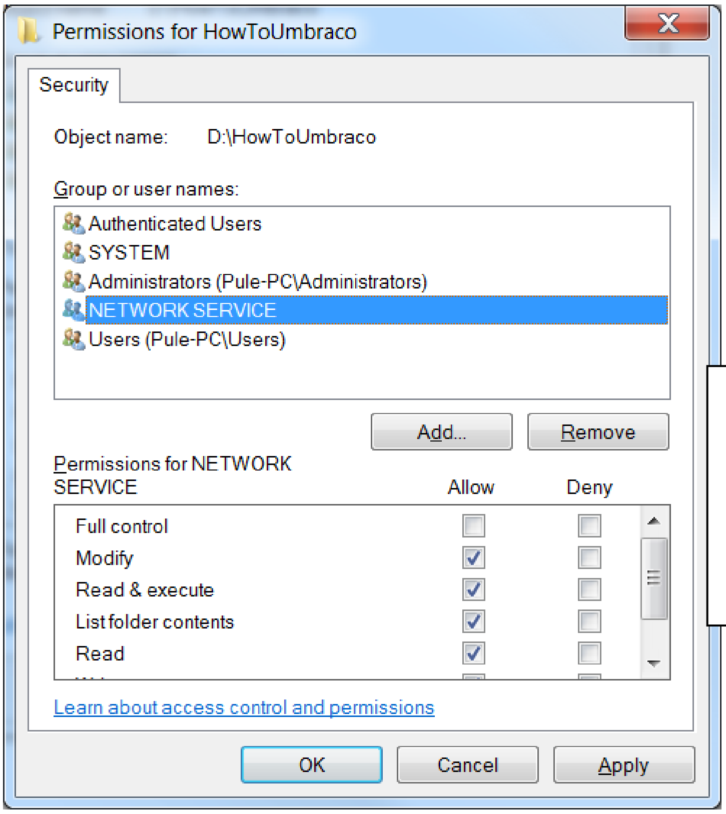
Directory:

Directory is location for physical files, all Umbraco files goes or placed inside domain name directory. I will define directory from two points of view one from server and one from local machine.

Server: If you buy web hosting including domain and space. You can see there is “httpdocs”[6] directory that is main directory for your website. All Umbraco files should be placed within main directory. Do not place Umbraco files inside sub directory because Umbraco does not support until version 4.7 and I can’t say about later version until its release. All Umbraco files directory should be placed without renaming into root directory.

Local Machine: I am installing Umbraco in local machine so that I will focus here more. First you need to Create Directory and give name “HowToUmbraco” in my case but you can give different name. Follow instructions.

- Create Directory in hard disk.
- Right click on it, click on Properties → Security → Edit → Add
- Type “Network Service” and press OK
- Select “Network service” and Checked “Modify”
- Press Apply and Press OK until it close property windows.
- Your Directory is ready to use for Umbraco websites.



Permissions for HowToUmbraco

Security

Object name: D:\HowToUmbraco

Group or user names:

- Authenticated Users
- SYSTEM
- Administrators (Pule-PC\Administrators)
- NETWORK SERVICE
- Users (Pule-PC\Users)

Permissions for NETWORK SERVICE	Allow	Deny
Full control	□	□
Modify	☑	□
Read & execute	☑	□
List folder contents	☑	□
Read	☑	□

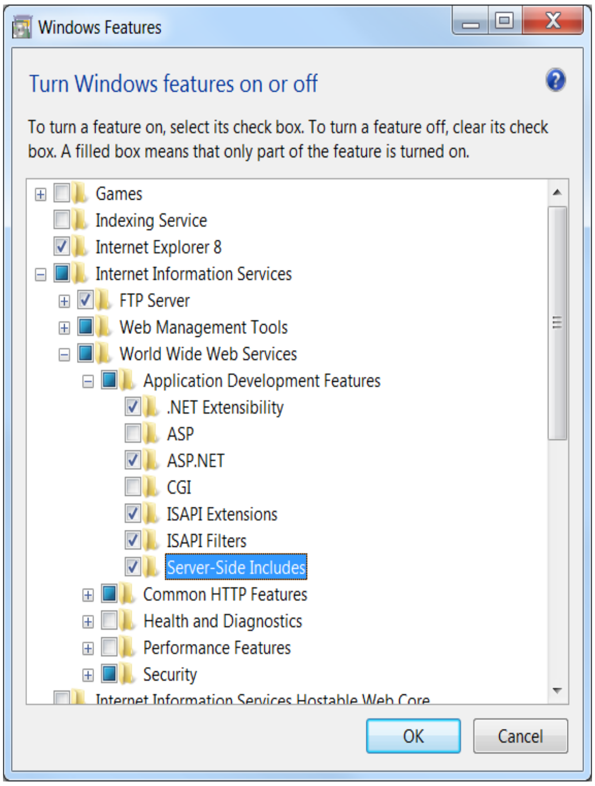
Learn about access control and permissions

Here I Created Directory “HowToUmbraco” in Drive D: and checked “Modify” file permission for “Network Service”

[6] “httpdoc” directory may not same name for all. It’s based on my experienced with www.dollar2host.com

Turn On IIS Service

Here I assume that your machine is windows 7. To turn on IIS service follow instructions.



Windows Features

Turn Windows features on or off

To turn a feature on, select its check box. To turn a feature off, clear its check box. A filled box means that only part of the feature is turned on.

- ☐ Games
- ☐ Indexing Service
- ☑ Internet Explorer 8
- ■ Internet Information Services
  - ☑ FTP Server
  - ■ Web Management Tools
  - ■ World Wide Web Services
    - ■ Application Development Features
      - ☑ .NET Extensibility
      - ☐ ASP
      - ☑ ASP.NET
      - ☐ CGI
      - ☑ ISAPI Extensions
      - ☑ ISAPI Filters
      - ☑ Server-Side Includes
    - ■ Common HTTP Features
    - ☐ Health and Diagnostics
    - ☐ Performance Features
    - ■ Security
- ☐ Internet Information Services Hostable Web Core

- Type “Turn Windows features on or off” in Search Box and press Enter

It will open new windows. Navigate Internet Information Service and expand its tree.

- Navigate

    -World Wide Web Service

    -Application Development Feature

- Checked

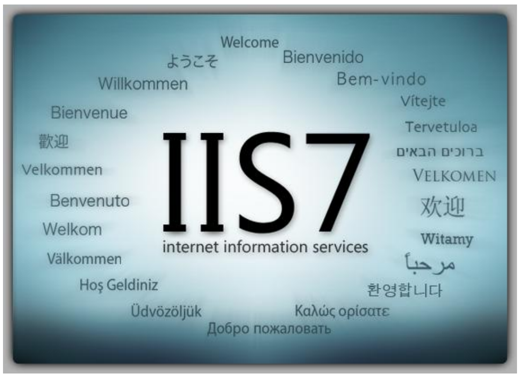
    - NET Extensibility

    - ASP.NET

    - Server-Side Includes

- Press OK

Wait until “Turn Windows Features On or Off” Close automatically.



Defaultwebsites.image

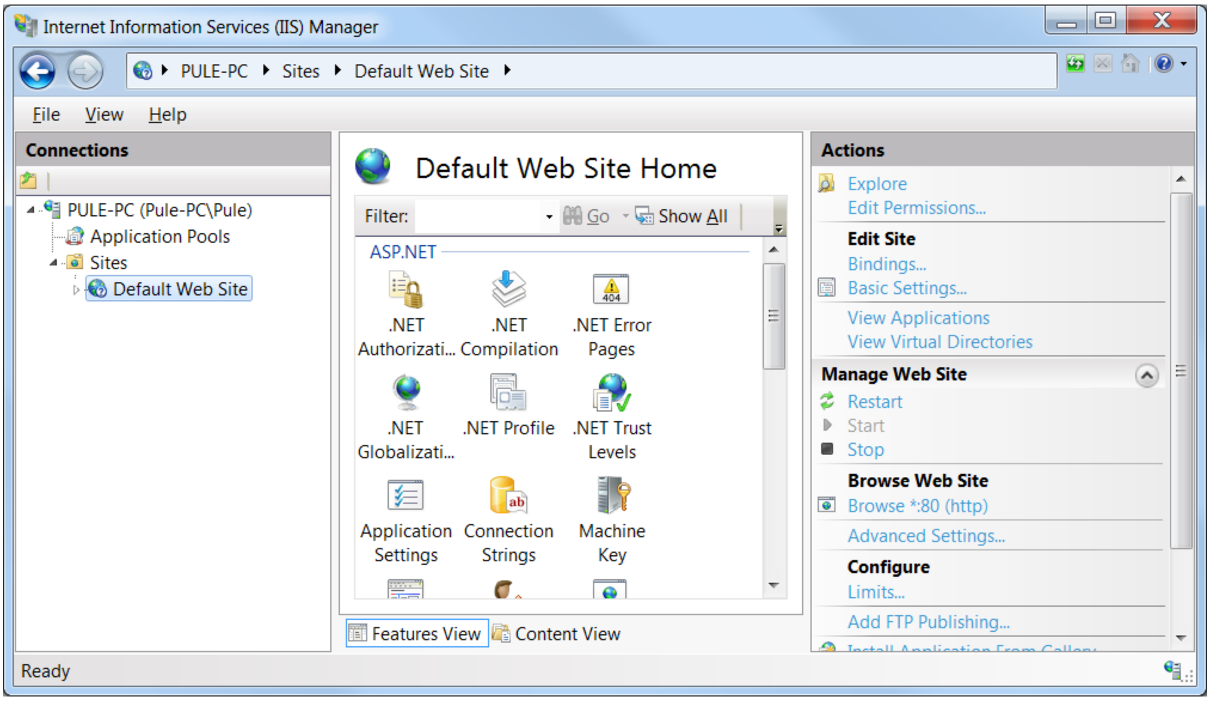
Test your IIS Server

Type “localhost” in your browser it will open default IIS websites (see left image) and also check using drive C:// “Inetpub” directory should be created by IIS Service. If there is a directory and localhost returns default websites[7] means IIS is ready to use.

[7] See image defaultwebsites.image

IIS Set Up

Until now we just turn on IIS7 in local machine. Here I will explain about Application Pool and other relevant topic to IIS and Umbraco CMS. Let's open Internet Information Service (IIS) Manager. To open IIS manager type "IIS" in search box windows search result will show up "internet Information Service" and press OK. It will open IIS manager windows. See screen bellow



General Description of IIS Manager environment

Connections:

List of all websites that created by users. By default there one website name is "Default Web Site" which is IIS created automatically when we Turn On Internet Information Service from "Turn windows features on or off" program.

Application Pools:

According to Microsoft, when you run IIS 6.0 in worker process isolation mode, you can separate different Web applications and Web sites into groups known as *application pools*. An *application pool* is a group of one or more URLs that are served by a worker process or set of worker processes. Any Web directory or virtual directory can be assigned to an application pool. For more details you can follow link provided in reference[8].

Sites:

You can create new sites inside your local IIS manager. Here we are going to create new websites for our Vcard Project. Sites menu contains list of created sites into your local machine. You can edit, delete and create easily by right clicking on specific sites.

Create new site:

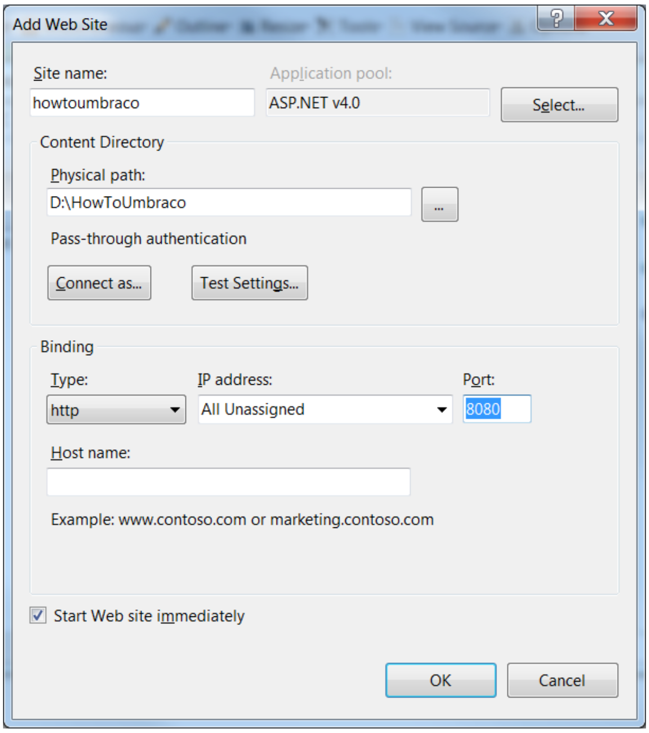
Instruction:

- Right click on Site menu on left side at communications. If you did not see "sites" expand left menu tree.
- Click on "Add Web Site"
- Provide valid information see captured screen for references.
    - Site name "howtoumbraco" – you can give your own name if you like
    - "Application Pool" is created automatically but I recommend changing it so that click on "Select" button at right side. It will list all available application pool. Select "ASP.NET v4.0" for Umbraco 4.7 and press OK.

[8] http://www.microsoft.com/technet/prodtechnol/WindowsServer2003/Library/IIS/67e39bd8-317e-4cf6-b675-6431d4425248.mspx?mfr=true

- Physical Path: this is important part here so click on browse button right side and locate directory where you created before for Umbraco installation. In my case “HowToUmbraco” directory in drive D:

- Binding: I changed port number to 8080 you can leave default but I recommend to change it. You can specify any port number if it’s not used or occupied by other application like Skype. In my case Skype already use default port number 80 so that I need to change it something else. So I change to 8080. Remember that if you change port number you have to provide port number every time when you access your websites via browser.


Add Web Site

Site name: howtoumbraco

Application pool: ASP.NET v4.0

Content Directory

Physical path: D:\HowToUmbraco

Pass-through authentication

Binding

Type: http

IP address: All Unassigned

Port: 8080

Host name:

Example: www.contoso.com or marketing.contoso.com

☑ Start Web site immediately

Press OK:

You have successfully create sites and assign application pool “Asp.net v4.0”.

TESTING IIS:

It is necessary to test our site and setting we have done before, everything should work perfectly. To test newly created site follow these steps:

- Open Notepad
- Copy and paste that line of code without changing it. Copy code bellow

```
<% Response.write(" welcome to 'How to Umbraco Basic'") %>
```

- Click on Save. Now be very careful Naming is very important here. Give file name DefaultTest and end with .aspx without space so your file name must be “DefaultTest.aspx” without quote. Now save that file inside Umbraco installation directory in my case I saved DefaultTest.aspx file inside “HowToUmbraco” Directory.
- Open Browser and type localhost:8080/DefaultTest.aspx here 8080 is port number that I changed before, if you did not change port number “localhost/DefaultTest.aspx” will be enough.
- If you can see welcome to 'How to Umbraco Basic' message in browser. That means your sites is working if you did not see that message or get some error please repeat Previous Chapter.

Modifying your Sites:

If you like to modify your sites properties or other settings your can click on your sites and on middle and right side you can see more options available to edit. For example:

- Edit bindings, Permission and basic settings
- Advanced settings
- View application, add virtual directory
- Edit SMTP or Email setting, database connection strings, SSL, Handler Mappings etc.
- For more details you can visit Microsoft official sites for IIS.

References:

Taken date April 13, 2011

IIS

http://www.iis.net

http://www.microsoft.com/downloads/en/details.aspx?FamilyID=32c54c37-7530-4fc0-bd20-177a3e5330b7

Notepad is out of box software provided by Microsoft.

Downloading Umbraco files:

It's time to download Umbraco latest file from official site. Official site contains list of Umbraco release, most resent Umbraco release version is highly focused in download site. I recommend downloading latest version because every new release contains bug fix from previous version, added extra new features and improved better than previous release. Here you can argue that new version of Umbraco also contains new bugs and it may take longer time to fix because it's a new release. You are not wrong but my point is we all are here to learn new thing and challenge new problem, help others and get help from others so do not compromise to face new challenge and learn more, technology never becomes perfect, it's a process of learning from history, face new challenge in present and improve it better for future.

In my personal opinion I download one or two previous release then latest release version only for client websites because earlier version of release has known bugs and can be solved very fast because someone or other developer might be already faced or fixed such bugs. But in my local host or for my testing purpose I always download latest release because I always want to face new bugs that nobody discovered it and have a fun with that bugs. In this Vcard project I will download latest version of Umbraco release 4.7.

What to Download?

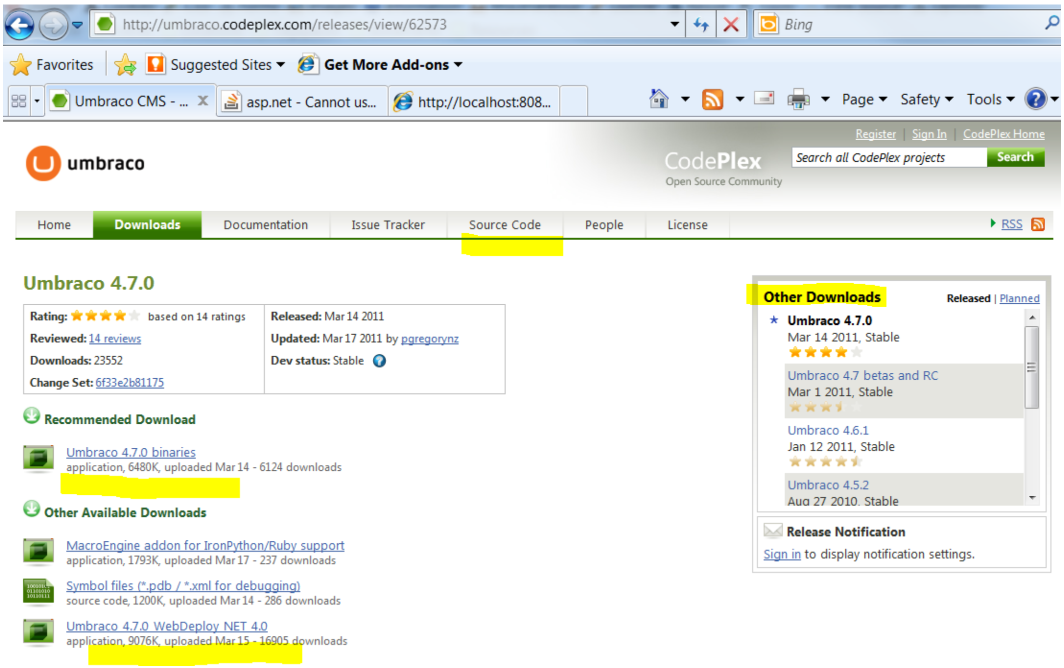
Now you get overview of downloading Umbraco. It's a good questions what to download? If you visit **http://umbraco.com/download** sites to download, you can easily install using "Web Platform Installer". Web platform Installer is Microsoft free software use to install program. Here I will not use "Web Platform Installer" because I want to show you more options. Visit Umbraco release or let's follow this link **http://umbraco.codeplex.com/releases/**. This websites contains all Umbraco release version. On left menu side you can see current release version. Inside rectangle box you can see details of latest release version, for example Rating, Release date, update, Reviews, Number of Download etc. here look for Dev Status: if it's say stable it means it's a believe to be ready to use. On right side menu bar you can see pervious release of Umbraco. I said not to use Web Platform Installer to install Umbraco because it will install new release. If you would like to install other version then you need to browse codeplex for all releases.

Let's back to main questions what to download? I would say first describe yourself and decide what to download. Here are three main categories so you need to choose one categories yourself based on your programming skills and .NET language skills.

Categories Basic – if you have basic knowledge of C#, object oriented program, MsSql database, if you know how to use usercontrol, webservices and classes to asp.net websites than I said its very basic to Umbraco so you can download Recommended Download by Umbraco sites. It's a binaries file.

Categories Tester - if you have advanced level of C#, Object Oriented program knowledge and if you do not want to use mssql database but want to test Umbraco than you are as a tester. Please download "Umbraco x.x.x WebDeploy NET n.n" here x.x.x is Umbraco Release version and n.n is .NET frame work version. Until this book release its "Umbraco 4.7.0 WebDeploy NET 4.0" so download that files and test Umbraco without using MSSQL database.

Categories Advanced – If you can analyze other's c# code and have knowledge of Object Oriented Program, database and other programming language then you can download Source code from here **http://umbraco.codeplex.com/SourceControl/list/changesets** and have a fun with source code.



Umbraco 4.7.0

Rating: ★★★★☆ based on 14 ratings
Reviewed: 14 reviews
Downloads: 23552
Change Set: 6f33e2b81175

Released: Mar 14 2011
Updated: Mar 17 2011 by pgregorynz
Dev status: Stable

Recommended Download

Umbraco 4.7.0 binaries
application, 6480K, uploaded Mar 14 - 6124 downloads

Other Available Downloads

MacroEngine addon for IronPython/Ruby support
application, 1793K, uploaded Mar 17 - 237 downloads

Symbol files (*.pdb / *.xml for debugging)
source code, 1200K, uploaded Mar 14 - 286 downloads

Umbraco 4.7.0 WebDeploy NET 4.0
application, 9076K, uploaded Mar 15 - 16905 downloads

Other Downloads — Released | Planned

- Umbraco 4.7.0
  Mar 14 2011, Stable
- Umbraco 4.7 betas and RC
  Mar 1 2011, Stable
- Umbraco 4.6.1
  Jan 12 2011, Stable
- Umbraco 4.5.2
  Aug 27 2010, Stable

Release Notification

Sign in to display notification settings.

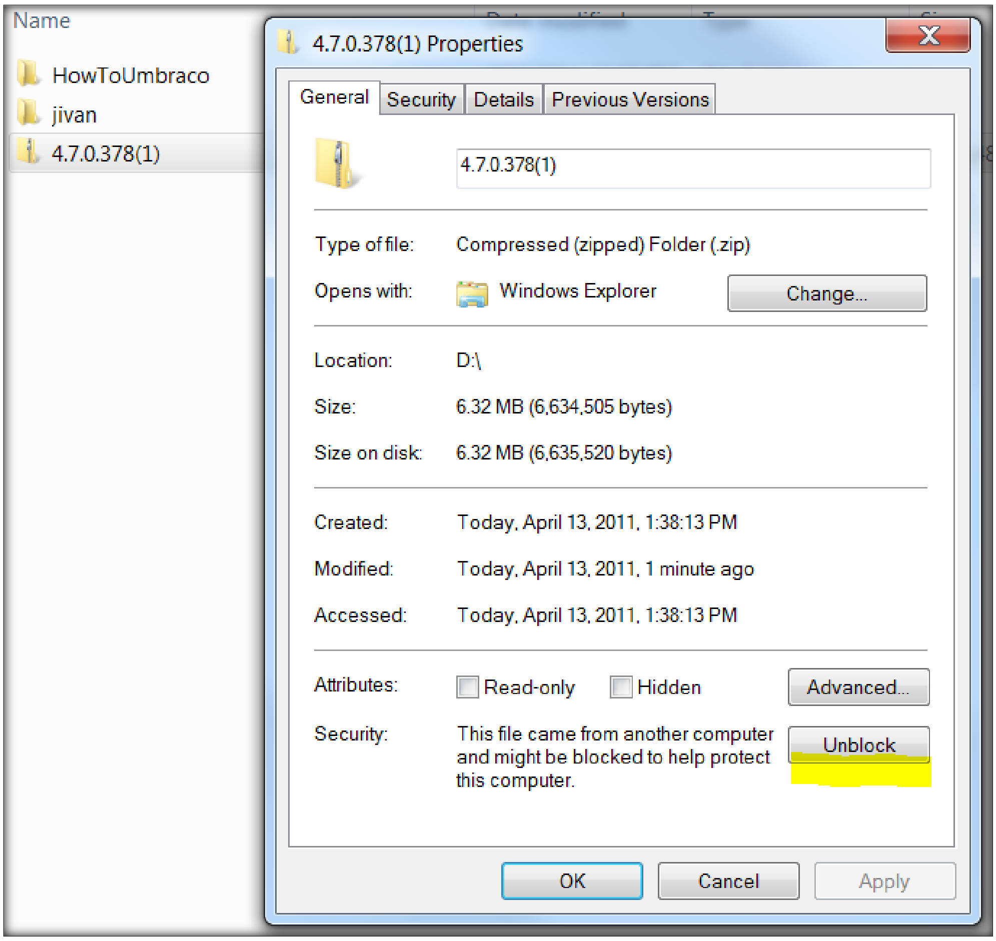
My personal opinion: I download Binary files for client websites development and I download source code files for learning purpose. I strongly recommend you to download source code because using source code files we can easily debug and see how Umbraco works. It's a good way to learn Umbraco. For Vcard project I download binary files, binary files are recommended by Umbraco official.

Unzip ZIP format

Umbraco downloaded files are compressed in ZIP format, I guess you already worked with zip format; one important thing for Umbraco zip file before extract is to unblock it. Make sure you unblock zip file before extract.

Instruction:

- Right click on Umbraco files that you just downloaded and click on Properties
- Click On "Unblock" and press OK
- Extract files by Right click on zip files and click on "Extract All"
- Your extractor can be difference from I have one. But purpose is to extract all files.



4.7.0.378(1) Properties

General | Security | Details | Previous Versions

4.7.0.378(1)

Type of file: Compressed (zipped) Folder (.zip)

Opens with: Windows Explorer

Location: D:\

Size: 6.32 MB (6,634,505 bytes)

Size on disk: 6.32 MB (6,635,520 bytes)

Created: Today, April 13, 2011, 1:38:13 PM

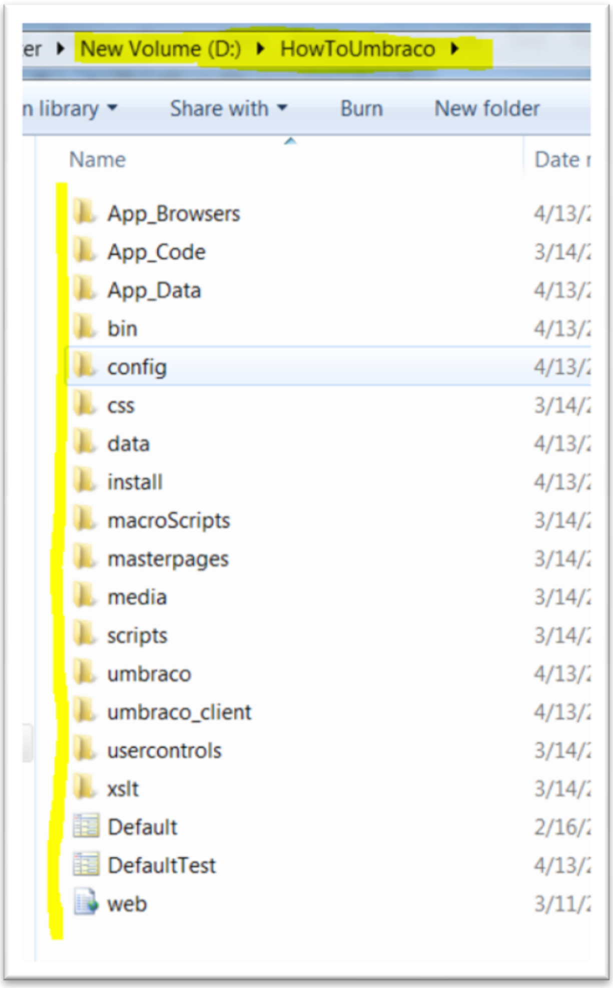
Modified: Today, April 13, 2011, 1 minute ago

Accessed: Today, April 13, 2011, 1:38:13 PM

Attributes: □ Read-only □ Hidden

Security: This file came from another computer and might be blocked to help protect this computer.

- Copy all extracted files, do not copy “build” directory[9]. Go inside “build” directory and copy all files from there.
- Paste all files inside “HowToUmbraco” directory if you follow me. Otherwise paste all Umbraco extracted files to your directory where you have decided to use. If you get confused go back to “Directory” topic and follow each step.
- If you followed me your “HowToUmbraco” directory must look like this see screen
-



Name	Date r
App_Browsers	4/13/
App_Code	3/14/
App_Data	4/13/
bin	4/13/
config	4/13/
css	3/14/
data	4/13/
install	4/13/
macroScripts	3/14/
masterpages	3/14/
media	3/14/
scripts	3/14/
umbraco	4/13/
umbraco_client	4/13/
usercontrols	3/14/
xslt	3/14/
Default	2/16/
DefaultTest	4/13/
web	3/11/

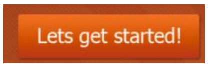
If yours file structure is same as above captured image then you are ready to install Umbraco.

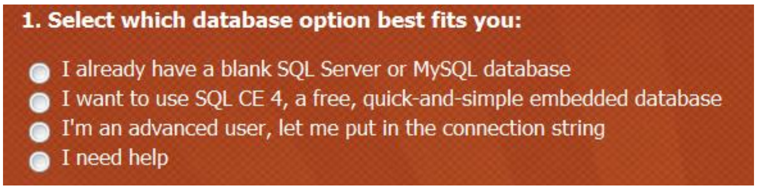
[9] You can also copy “build” directory in your case, if you want to follow me copy directory and files that have inside build directory.

Install Umbraco step by step guide

To install Umbraco Follow these process

Instruction:

- Browser you websites using browser and type localhost:8080
- Here 8080 is port number if you did not change your port number just type localhost
- You must see Umbraco welcome page click 

on "Lets get Started"
- Even its free open source CMS you need to follow rules to use Umbraco read license, term and conditions. More details can be found on official websites of Umbraco, Accept the license for Umbraco CMS. To accept license click "Accept and continue"
- You are now in step 3. This step is more fun.
- By providing not enough file permission to "Network service" or "application pool" in Umbraco installation directory will thrown out notification message "files permission are not ready" in such case you will not see this step 3, go back to file permission setting and come back later. If everything is ok you can see more option here
- 

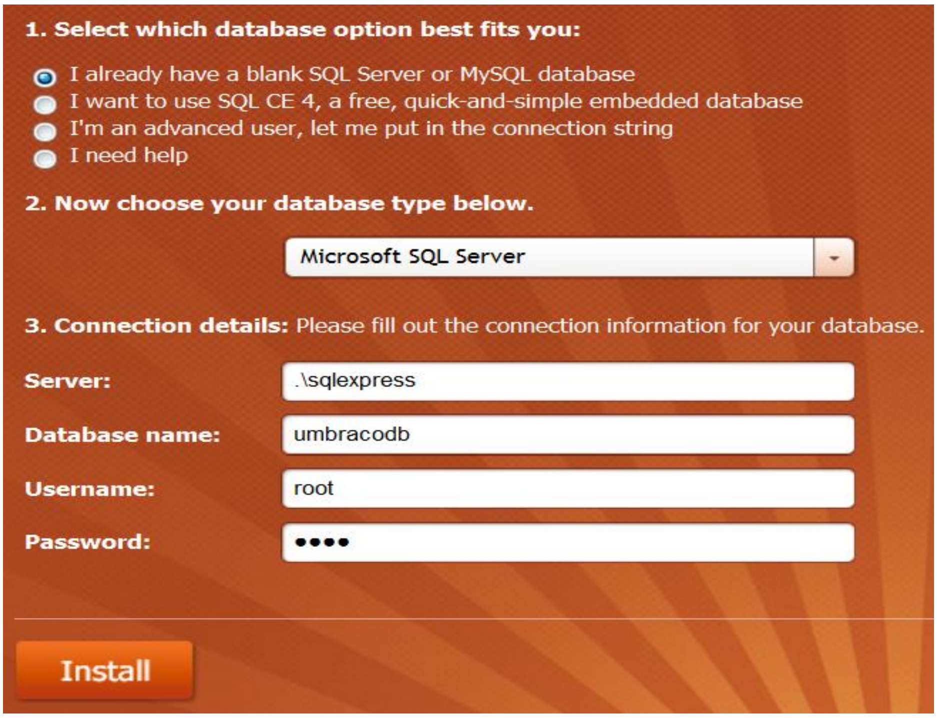
  1. Select which database option best fits you:

  ○ I already have a blank SQL Server or MySQL database
  ○ I want to use SQL CE 4, a free, quick-and-simple embedded database
  ○ I'm an advanced user, let me put in the connection string
  ○ I need help
- Four options are here and all four options have on radio button. Why it's in radio button, reason is Radio button allows to choose only one option at a time, in Umbraco it is not possible[10] to use two different databases at a same time that's why they have radio button to make sure to choose one database.
  - If you want to follow me until now click on first one "I …. Have a blank Sql server. *Also see "What to Download?" topics page 25* "
  - If you are planning to test Umbraco select second option "I want to use SQl CE4,…." It does not require database. *see "Categories Tester?" page 26*
  - If you are advanced user select third option this will allows developer to type connection string. *Also see "Categories Advanced" page 26*

[10] It is possible to use more than one database by extending umbraco using .net class. It is not possible to use multiple database during installation period.

- if you need help click on last one.
- I click on first one so Umbraco provides more option to select database type. I am using mssql, so I am selecting Microsoft SQL server.
- Your case can be deference if you plan to use MySql server than you need to find yourself how to connect to Mysql server, I am not going to include here.
- After click on first option or choice it will provide more input fields for sql connection details. See captured screen for reference your case can be deference. If you are following me it will be same as in captured screen bellow.
- Here is description
    - Server= location of sql server, in mycase its .\sqlepress, you can also use remote server database if you have for example server="www.example.com\sqlexpress"
    - Database name: Name of database that we created for Umbraco cms in database topic
    - Username: user that has access on database that we created for Umbraco
    - Password: password that assign to user to login sql server database.

-



1. Select which database option best fits you:

● I already have a blank SQL Server or MySQL database
○ I want to use SQL CE 4, a free, quick-and-simple embedded database
○ I'm an advanced user, let me put in the connection string
○ I need help

2. Now choose your database type below.

Microsoft SQL Server

3. Connection details: Please fill out the connection information for your database.

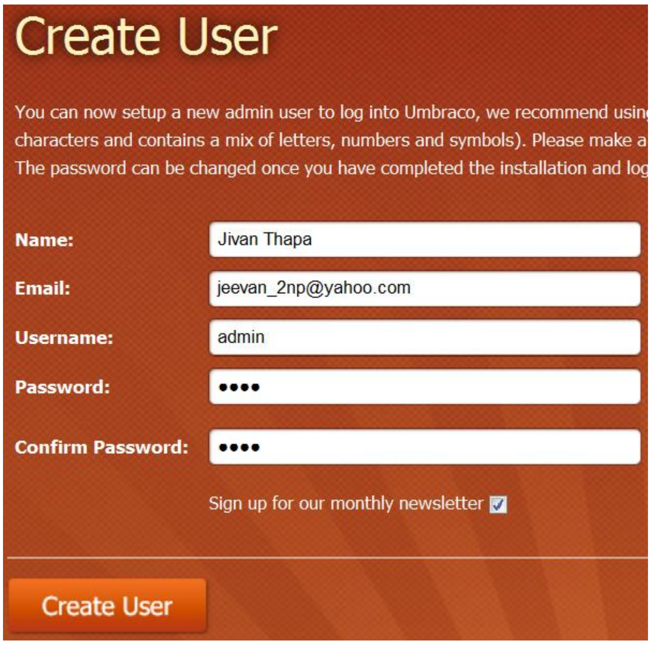
Server:	.\sqlexpress
Database name:	umbracodb
Username:	root
Password:	••••

- Press “Install”

- If server location, database name, username and password are correct then it will take only few minute to setup database for Umbraco, during this process Umbraco show up progress bar. If Umbraco show up any error or notice. Correct error based on error message provided by Umbraco and continue to install Umbraco

- Press “Continue” now you are in step 4

- Step 4 is for Create user; here you have to provide User information. That user will be supper administrator and have elevator access or root level of access to Umbraco Backend.
    - Name: name for supper administrator account
    - Email: email of supper administrator account
    - Username: user name use to identify user.
    - Password: secrete code to gain access
    - See captured screen as a reference.
    -



Create User

You can now setup a new admin user to log into Umbraco, we recommend usin
characters and contains a mix of letters, numbers and symbols). Please make a
The password can be changed once you have completed the installation and log

Name: Jivan Thapa

Email: jeevan_2np@yahoo.com

Username: admin

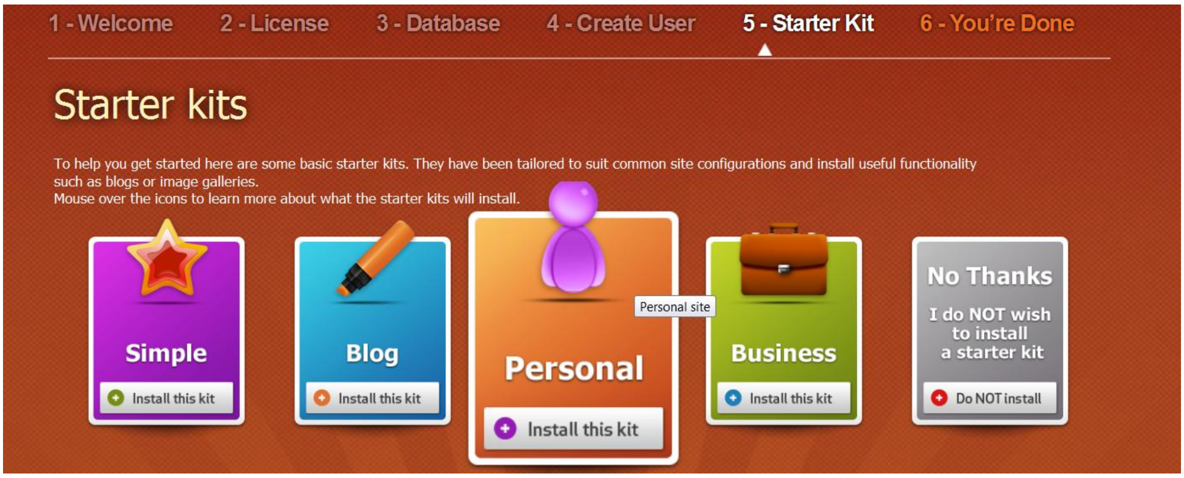
Password: ••••

Confirm Password: ••••

Sign up for our monthly newsletter ☑

- Click “Create user”

Step 5: select starter kits.



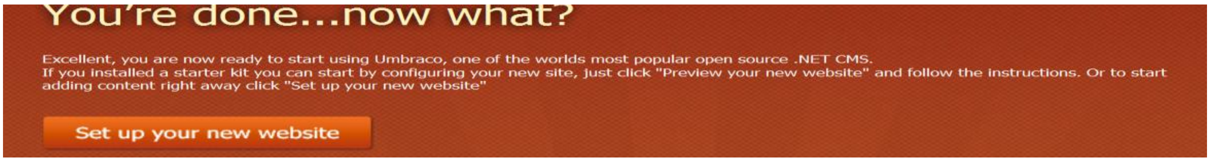
Starter kits

To help you get started here are some basic starter kits. They have been tailored to suit common site configurations and install useful functionality such as blogs or image galleries.
Mouse over the icons to learn more about what the starter kits will install.

You can choose any one for your sites. It will install selected starter kits. It contains Templates, xslt, css and other extra files based on selected starter kits. This is good way to learn for new Umbraco developer. By installing out of box starter kits will provide guidance to new user about how templates, add XSLT, Document types, Macro, CSS etc works in Umbraco. By analyzing starter kits help developer to learn quickly and get familiar with environments.

I will start from scratch that's why I am not going to install any starter kits but you can install if you want. I will click last one "no thanks". It will not install anything do not worry I will create later on. If you want to follow me click on "No thanks" and Click "Yes".

Final step: you are done now click on "Set up your new websites" it will redirect to administrator control panel that is Backend of Umbraco sites[11].



You're done...now what?

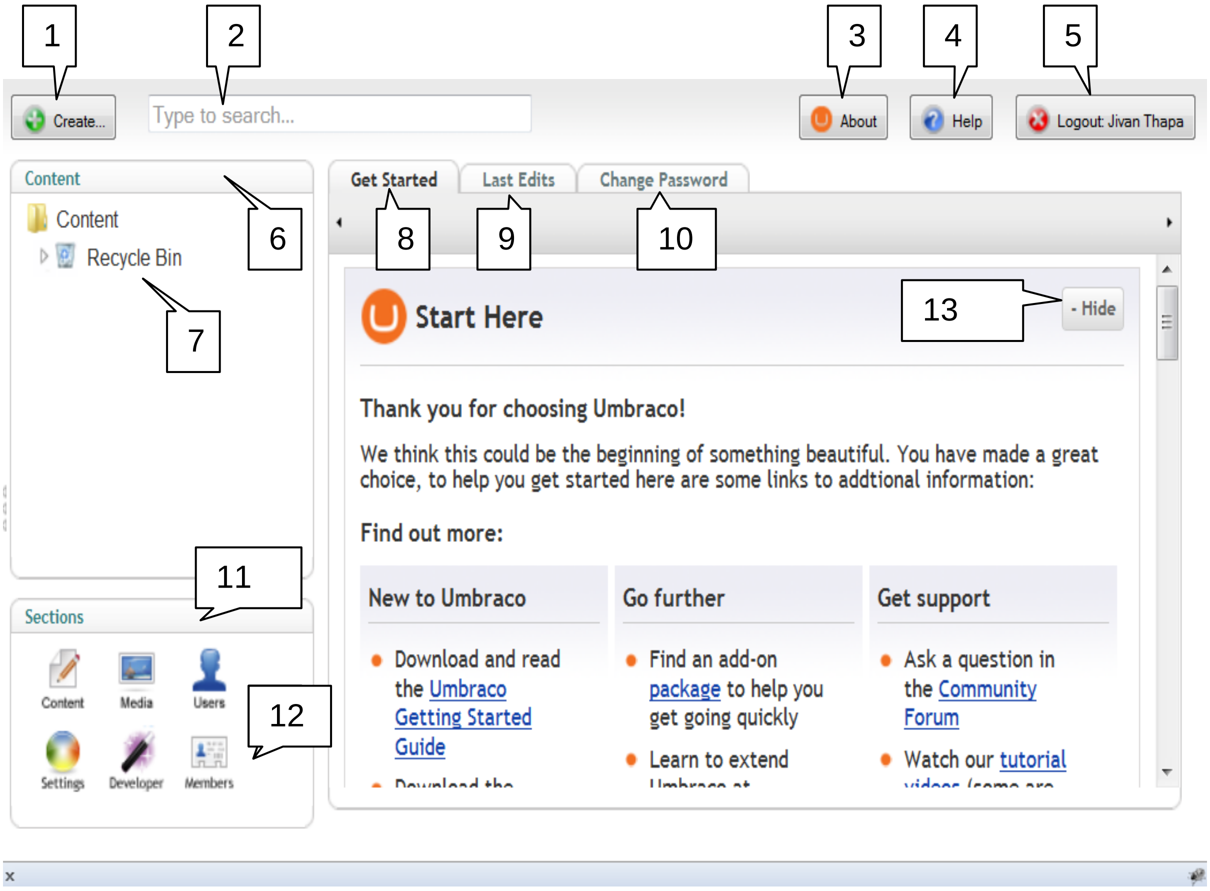
Excellent, you are now ready to start using Umbraco, one of the worlds most popular open source .NET CMS.
If you installed a starter kit you can start by configuring your new site, just click "Preview your new website" and follow the instructions. Or to start adding content right away click "Set up your new website"

Note: if you browse you sites you will see Umbraco notice "Looks like there's still work to do". It's because we did not set up our Umbraco sites. I will show step by step guide to set up.

Note: to install starter kits, it should be connected to internet. Being offline may not work to install starter kits.

[11] Also called back office.

Back End Working Environment



Start Here

Thank you for choosing Umbraco!

We think this could be the beginning of something beautiful. You have made a great choice, to help you get started here are some links to addtional information:

Find out more:

New to Umbraco	Go further	Get support
Download and read the Umbraco Getting Started Guide Download the	Find an add-on package to help you get going quickly Learn to extend Umbraco at	Ask a question in the Community Forum Watch our tutorial videos (some are

1) Create: it’s dynamic, will change based on Section item see number 12.

2) Search: type text to search content

3) About: To view Umbraco installed version, licensed and other information.

4) Help: external link, will open Wiki page about Umbraco

5) Logout: click to log out. It will contain login user name

6) Content: it’s a dynamic, will changed based on Section Items see number 12

7) Secondary items or child items of Main Items.

8) Get Started: Umbraco provides information, news and external links

9) Latest edits: list latest change you have made on you site.

10) Change Password: change admin password. Password is hashed

11) Section: Main menu items, we can add custom section. Not covered here

12) Section Items: Items inside section. Sub Menu part of Section Menu. Mostly used in developing Umbraco, creating content and other administrator purpose. Umbraco come up with six default section items Content, Media, User, Settings, Developer and Members.

Content: here editor create articles, edit, upload image. It's not technical stuff

Media: it contains Media items. Image, Video, Audio is mostly Media items. Uploaded files go inside Media section. Using .net control developer can easily use others Directory to store files on their needs and requirements.

User: contains list of Users. All registered user are listed under User. Here you can set up user id, password, Email, permission and other personal information.

- Users
- User types
- User Permission

Members: it contains

- Members
- Member Groups
- Member types

Settings: widely used in this "how to Umbraco". Setting used to set up umbraco site. It contains

- Document Type
- Templates
- Style Sheet
- Scripts
- Dictionary
- Language
- Media types

Developer: it contains all technical stuff.

- Cache browser
- Data Types
- Macro
- Packages
- Scripting Files
- Xslt Files

Chapter 03: Document Type

Document Type:

Document type is a core feature[12] of Umbraco cms; it is out of box document type feature. If you already work with Drupal CMS, you might be working with CCK (Content Construction Kit), Document Type is exactly same purpose as CCK that allows you to create fields.

Document type is a group of fields, that controls fields and fields are used to input data by content creator or editor while creating content on Umbraco sites. Content is second step after Document Type so that Document type structures the input fields, creates environment to add or create new content and facilitate content creating services to Editor.

Document Type for Developers:

Developer creates document type based on groups of data that needs to be saved to and provides rich service to manage data and customize input fields.

You may still not clear what document type is so I am going to give one example as vcard case. Vcard websites I called “Semic Provider” has many fields like

- Logo
- Title of Websites “Semic Provider” and
- Slogan “Where content is understood by human and machine”.

As a developer I want to provide a service to client so s/he can upload Logo, edit title of websites and slogan. Developer could do hard coding text into templates to display logo, title of websites and slogan but client can’t change it later if s/he decided to change it. Client wants cms because s/he wants to control content without having knowledge of technical stuff like editing templates so as a developer we should provide more control over content. Logo, Title of Website and slogan are not same for all time; they can be changed later by client. To solve that problem we need to create Document type.

Document type is not single subject it’s more complex for new Umbraco developer. I am going to look more details about document types. It contains Properties, Info, Structure and Tabs.

Properties: it’s an input field that you create in document type, has more properties

- Name: name of your input field
- Alias: alias used to retrieve value or fetch value that properties have
- Type: input type for example , Text box, upload, Button, check box etc. these are based on Data Type
- Tab- by default properties is assign to generic properties but you can assign to your custom tabs

[12] http://umbraco.com/help-and-support/video-tutorials/introduction-to-umbraco/sitebuilder-introduction/document-types official video about Document Type introduction

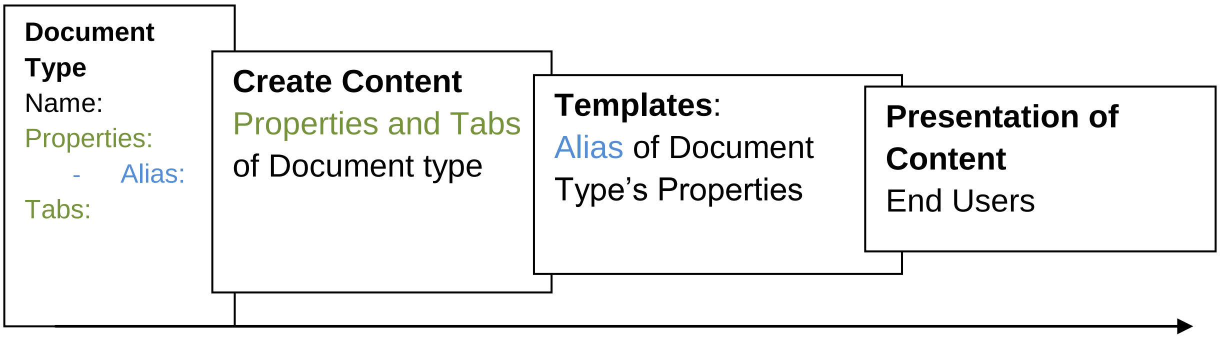
Tabs: Tabs is name of group for example name, age, address and email properties of person can be group together with tabs "personal Details" so in Tabs we create "Personal Details" and in Property tab we assign name, age, address and email properties to "Personal Details" tab. Tab and Tabs should not be confuse. Tabs, Properties, Structure and Info have save level inside Document Type tree and tab is child of Properties.

Structure: Document type can have parent or child document Type here you check what's should be parent or child document type is allowed

Info: it's not that much important as Properties, Tabs and Structure. Here you can assign template and other general setting or get more information about Document Type. More available settings inside Info are listed below. These are taken from official Umbraco sites. Please visit references.

- Name: name of Document Type
- Alias: alias of Document Type, use by Umbraco and developer when they work with Umbraco API and CodeInProject.
- Thumbnail: Thumb image of Document Type displayed on create page dialog.
- Icon: Icon of Document Type.
- Description: Description of Document Type.
- Allowed Template. It is use to present data to end uses. If you fetch data using Xslt you can make document type without templates but Content based on that document type will not be able to provide preview mode if there is no associated template. One Document Type can have more than one template.
- Default Templates: that is a default template of document type. And used by Document Type

Visual Diagram[13] for Document Type:



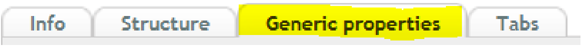
[13] These diagrams are my own concept and may be other developer argues with Diagrams. There is no official diagram for Document Type.

References taken dates April 14, 2011

Diagram show, first create Document Type, second create Content based on Document Type Properties, use Alias of Document Type's properties and display data to End users.

Practical: let's create Document Type for "Sites Setting"

Document Type Generic Properties:

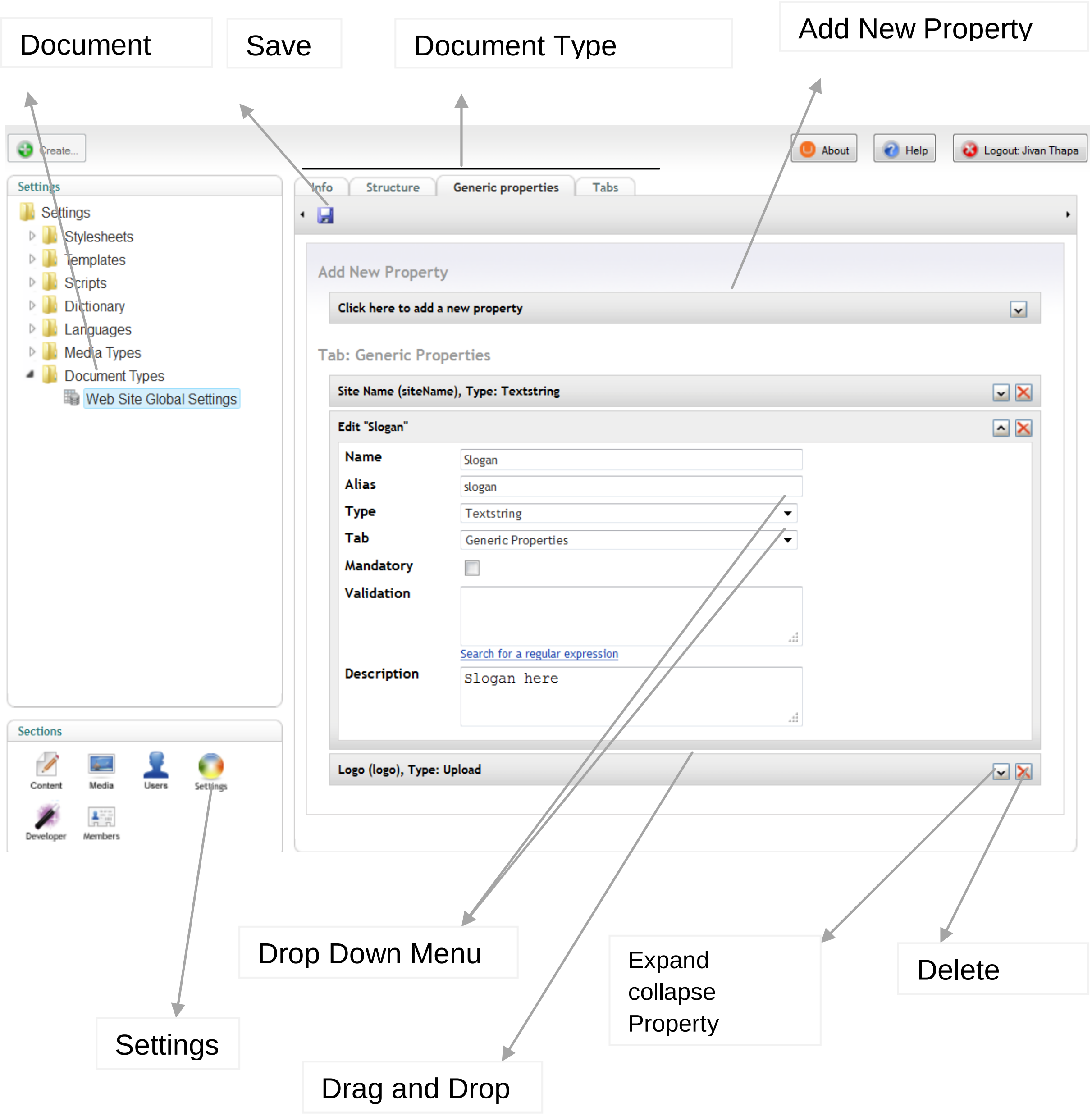


Instruction:

- open Backend of Umbraco sites (your site /Umbraco/) in my case localhost:8080/umbraco/
- Provide login credentials user id and password (my case "admin" "root" login and password respectively)
- Click Setting on Section at bottom left corner, it will list all available features on Setting on top left corner. Expand Document Type
- If you follow me Document Type must be empty so nothing will happen it's because I decided not to install any starter kits. Now I will create new document Type
- Right Click on Document Type and Click "Create". New windows popup.
- Master Document Type: select none…
- Name: name for Document Type I choose "Web Site Global Setting" but you can choose different if you want. Its name of document type.
- By Default "Create matching templates" is checked so leave default, it will create matching templates and we will use that templates as master templates.
- Press "Create" Button
- Now Umbraco created new document Type and provide more option to input.
- Click "Web site global Setting" to open document type. It will open four properties Info, Structure, Generic Properties and Tabs
- Info: if you want to follow me provide these information
    - *Name* and *Alias* is not recommended to change because it will effect whole system
    - *Icon*: Setting Agent. Select from dropdown
    - *Thumbnail*: developer.png select from dropdown
    - *Description*: "Manage your websites. Here you can set up your global sites. Example change logo, slogan etc."
    - *Allowed templates* and *Default Templates* leave default do not change it.
    - **Remember to save before jump to next property**.
    - You must see balloon success message at bottom right corner if anything goes wrong it will show up balloon message with error details.
- Structure: leave Default
- Tabs: do not create for this document type
- Generic Properties: click on Generic Properties. It's a most important in document Type. Click on "Add New Property".

- It will expand fields to input data, if you want to follow me provide these information without quotes
    - ○ *Name*: “Site Name”
    - ○ *Alias*: Leave Default, Umbraco will generate for you
    - ○ *Type*: Textstring , select from Drop down menu
    - ○ *Tab*: Leave Default
    - ○ *Mandator*: checked, means, it should not be empty
    - ○ *Validation*: for now leave empty
    - ○ *Description*: “Your Site Name”
    - ○ **Remember to save before jump to next properties**.
    - ○ You can see balloon “success message” at bottom right corner if anything goes wrong it will show up balloon message with error details, correct error based on error message and Save again.

See captured screen as references.



Add New Property

Click here to add a new property

Tab: Generic Properties

Site Name (siteName), Type: Textstring

Edit "Slogan"

Name	Slogan
Alias	slogan
Type	Textstring
Tab	Generic Properties
Mandatory	□
Validation	Search for a regular expression
Description	Slogan here

Logo (logo), Type: Upload

Drag and Drop: To sort property order, Umbraco automatically saves sorted order.

Delete: To delete Property Type, if you delete property type it will delete all content relevant to that particular Property Type

Expand collapse Icon. It will allows to expand or Collapse property Menu

Add New Property: To Add new property

DropDown Menu: to select value from Types or assign property to Tabs.

You must save manually after change; Umbraco does not automatically save for you.

Alias: alias is case sensitive and if you modify alias you may lose data if you are not sure how alias works. It's better not to change it in case you have to do so please make sure update in other parts of Umbraco system when you used that alias to fetch data. Alias is used to fetch data by Umbraco.

Type: there are many types out of box available, for example, Rich text editors, True/False, Textstring, Upload, all are called Data Type I will discuss later. Changing type will lost data in content editing page but will still remain on database unless you save content after type changed. It's better not to change so make decision before you decide what type should be used.

Some tips: if you want simple text editor select simple text editor, Textstring allows only to type text, Rich text editor will allows more rich features to content editor.

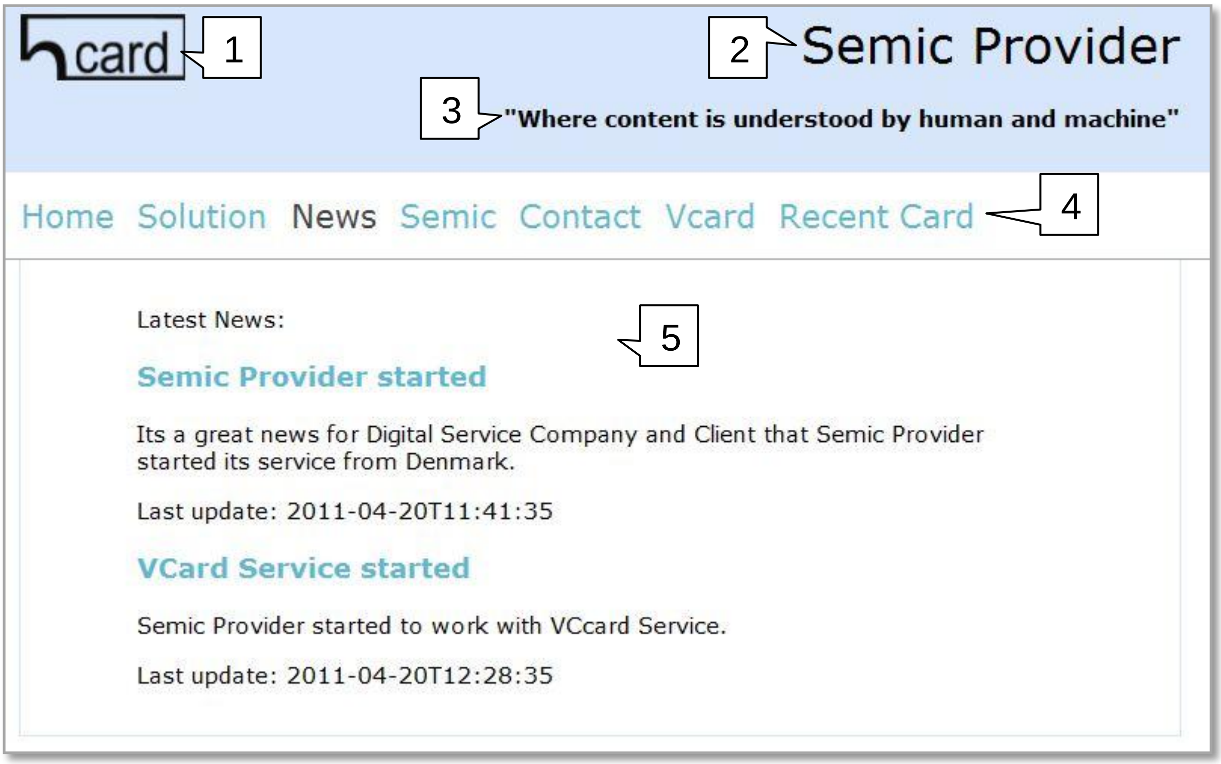
Note:

Repeat Same Process for *Name*: "Slogan", *Alias*: Leave Default, Umbraco will generate for you, *Type*: Textstring, *Mandatory*: Checked, *Description*: "Slogan here"

Repeat Same Process for *Name*: "Logo", *Alias*: Leave Default, Umbraco will generate for you, *Type*: Upload, *Mandatory*: leave default, *Description*: "Upload logo here".

> **Notes: Type: upload allows content Creator to upload Image or files while create content, Type: Textstring allows to type text in input text field.**

Let's see our final web grid how it look likes and create more Document Type for Semic website. See captured screen bellow.



Semic Provider

"Where content is understood by human and machine"

Home Solution News Semic Contact Vcard Recent Card

Latest News:

Semic Provider started

Its a great news for Digital Service Company and Client that Semic Provider started its service from Denmark.

Last update: 2011-04-20T11:41:35

VCard Service started

Semic Provider started to work with VCcard Service.

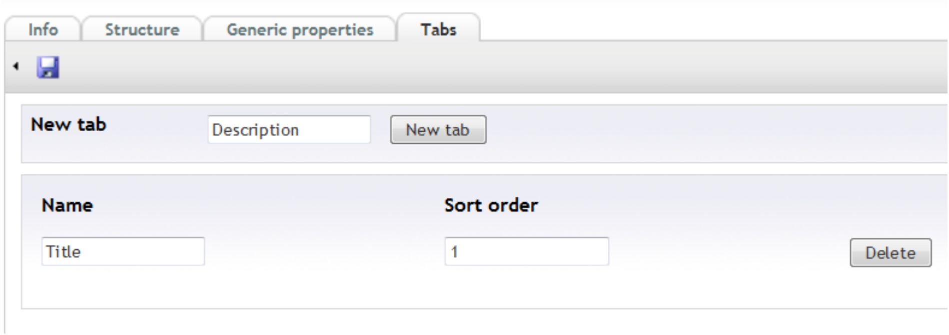
Last update: 2011-04-20T12:28:35

1) Logo
2) Title
3) Slogan
4) Navigation Bar
5) Body

Document Type “Tabs”

Here we just finished document type for global settings, but little document type is still remaining, take a look at navigation bar. There is title of Article like “News”, “Solution”, “Semic” and “Contact”. Now it’s time to create Document type for these articles. I will introduce input pattern here, Input Pattern is fields or value that needs to provide when Editor creates content (Article) on Content page.
Input Pattern is “Title” and “Body”. Title is a title of article that shows up in navigation bar, Body is text or description of Title that show up when user click on particular title in navigation bar. For example if user clicks on “News” Title it will show up Description of News, in my case it will show up recent news details. Here “news”, “Solution” and “Semic” have only two Input Pattern title and Body, it means we can create many article with different title and body on same input pattern. So let’s create one Document Type that can implement Two Input Pattern. Now let’s create new document type “General Content” and also check matching templates and Press OK.
If you want to follow me repeat same process as above to add new properties
Add following properties into “General Content”

- *Before Adding Following Properties. Create first Two Tabs "Title" and "Description".*
- *Type Tab name and Press "New tab" it will create new tab*



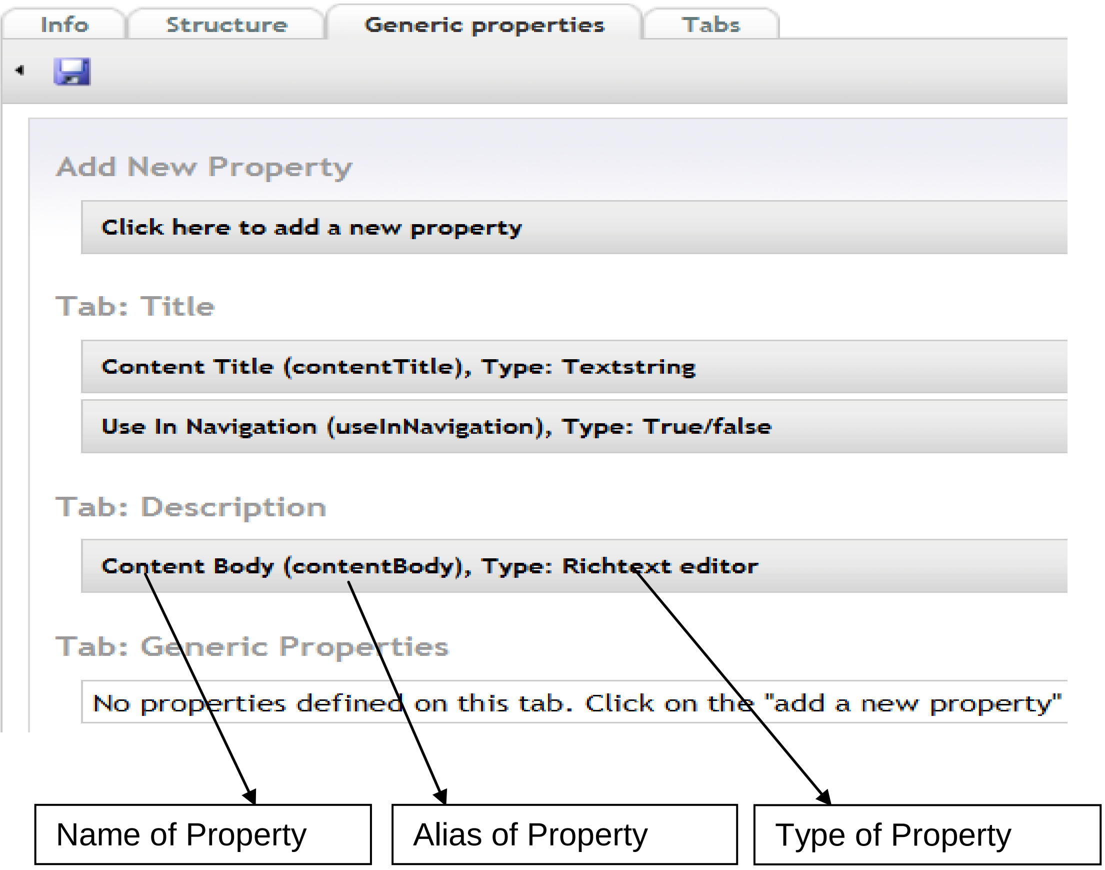
New tab: Description

Name	Sort order
Title	1

  - *Name: name of Tabs you created*
  - *Sort order: ordering number of tabs, lower sort order number will appear at top of the grater sort order number in "Generic Properties"*
  - *Delete: To delete tabs, related property will automatically assign to Generic Properties if tabs is deleted.*

- Name: "Content *Title*", Alias: Use Default, Type: *Textstring*, Tab, use Default, Mandator: *checked*, Description: *Article Title*
- Name: "Content *Body*", Alias, Use Default, Type: *Rich Text Editor*, Tab, use Default, Mandator: *checked*, Description: *Body text here for Article*
- Name: "*Use in Navigation*", Alias, Use Default, Type: *True/False*, Tab, use Default, Mandatory: *checked*, Description: *Checked to show content in Navigation bar.*
-

See captured screen as references for Document type "Tabs"



Add New Property

Click here to add a new property

Tab: Title

Content Title (contentTitle), Type: Textstring

Use In Navigation (useInNavigation), Type: True/false

Tab: Description

Content Body (contentBody), Type: Richtext editor

Tab: Generic Properties

No properties defined on this tab. Click on the "add a new property"

Name of Property

Alias of Property

Type of Property

Document Type Structure:

It's a little bit tricky, news items are added each time if there is new things happen, so we need to create two Document Types. One Document Type is for News Container and another for news article. News Container is like a container that contains news article.

Purpose: Umbraco developer can set up hierarchy structure so it will limit or prevent from creating non relevant content to content editor. For example I want only news items inside News Container, we want to mange content or data. I don't want to create any content that is not news items inside News container, to achieve that goal I can set up structure to News Container Document Type so it will only allows to create child content only relevant to news article. Let's begin to create Two Document Type "News Container" and "News Article"

Instruction: process is similar to create News Container Document Type as above mentioned.

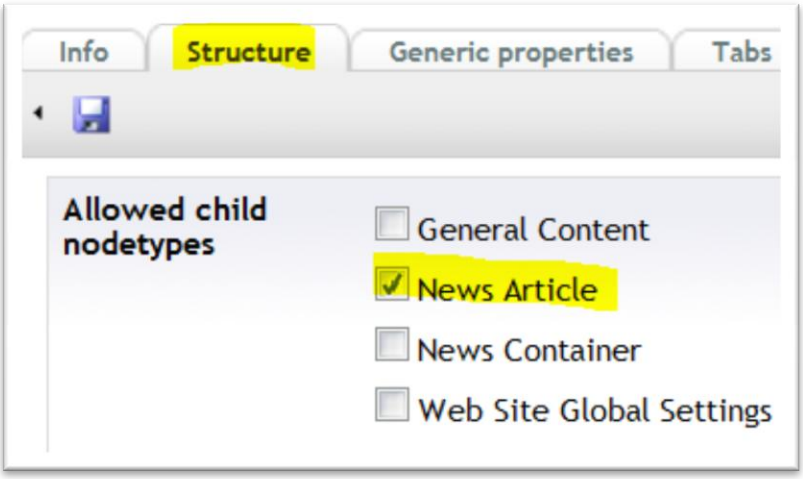
Document Type "News Container"

- Create, Document Type "News Container"
- Name: "*News Items*", Alias: Use Default, Type: *Label*, Tab, use Default, Mandatory: Use Default, Description: "*it's a news container*".
- *Remember to Save*

Document Type "News Article"

- Create Document Type "News Article"
- Name: *News Title*, Alias: Use Default, Type: *Textstring*, Tab, use Default, Mandatory: Use Default, Description: "*News Title".*
- *Remember to Save*
- Name: "*News Body"*, Alias: Use Default, Type: *simple Editor*, Tab, use Default, Mandatory: Use Default, Description: "*news details".*
- *Remember to Save*

Let's set up structure Open "Structure" of Document Type "News Container" as checked document type you want to set up as child nodetypes. In my case "News Article" see screen bellow.



Allowed child nodetypes

☐ General Content
☑ News Article
☐ News Container
☐ Web Site Global Settings

Creating Child Document Type

There are two ways of creating child Document Type.
First Ways. If you remember that we selected none "Master Document Type" when we created document type for global Setting, News Container and General Content. If you want to create child document type you can select desire master document type while you are creating Document Type first time.
Second Way: second way to create child document type is Right click on Document Type that you want to use as master Document Type and Click on "Create". This is another way to create child document type.
Purpose: creating child document type allows Developer to reuse properties that have in master document type.

Notes: Do not forget to play around all types of Child Document type, Structure, Info, Properties, Tabs, Validation and different Data Types or Type etc.

Document Type API:

We are creating document type manually by right click on Document Type using backend of Umbraco site. It's also technical stuff and you need to know Umbraco stuff to set up, if you want to be a Umbraco developer in future, you should not look on top to ocean, you have to go more deep until bottom of ocean. Umbraco CMS will convert all your manual work in programming language and execute code. Let's look what's happening inside Umbraco when you manually create document type, how it's working and get basic Umbraco API knowledge.
Umbraco has namespace "`umbraco.cms.businesslogic.web`" that contains los of Class. Here I will discuss DocumentType only.

- *Manual Work: "Create"*
  - When you right click on Document Type and press "Create". You have to provide Document Type Name.
  - At that time Umbraco execute MakeNew() methods on server that methods are responsible to create Document Type that will effect on Database. `DocumentType.MakeNew(User, docTypename).Save();`
  - User will be automatically assigned by Umbraco based on your backend login information, Note that not all uses have access to create document type. docTypename is Name of Document Type you provided first time. Save() is void methods, it will save Document Type to database. Saving to database means to make data available next time to use.

- *Manual Work: "Add New Property"*
  - Add new property will execute AddPropertyType(DataTypeDefinition, name, description) methods. Three parameters DataTypeDefinition, name and description and there is also more overloads on that methods.
  - DataTypeDefinition – Data Type you selected, Umbraco only use Id of Data Type. When you select data type, Umbraco automatically take id of data type.

- *Manual Work: "New Tab"*
  - Umbraco execute AddVirtualTab(tabName) to add new tab, tabName is name of tab you provided.
  - DocumentType
  - `dt.AddVirtualTab(tabName);`

- **Setting Tabs on Generic Properties.**
  - When you select tab to organize you preoperty Umbraco will assign tabs on that Property.
  - `DocumentType dt;`

    - dt.SetTabOnPropertyType(PropertyType, Tab Id);
    - here property Type is type of new property and Tab Id is node id of Tabs you created, all Names are for visual representation of data and id or aliad is for machine or Umbraco system to represent object.
-
- **Manual Work "Delete"**
    - If you would like to delete Document Type you will right click on document type and press "Delete".
    - Umbraco will execute Delete() methods to delete document type.
    - Here is an example
    - DocumentType.GetByAlias(docAlias).delete();
    - First call GetByAlias() method to select document type by alias, also you can select document type by Id.
    - Second Call Delete() method to delete Document,
    - This time you don't need to save. Because Delete will delete selected document type from Database as well.
- Whatever you made change on Umbraco, you have to press Save icon 💾 to save your work. That will execute Void methods Save().

More Document Type API:

Here is more API listed, for full list of API, you need to download source code or decompile DLL files that provided by Umbraco binary.

- getVirtualTabs
    - returns list of available tabs on document type
- HasTemplate()
    - Return bool value ,true if document type has template
- allowedTemplates
    - return List of Allowed templates.
- PropertyTypes
    - Return list of Property types.
- HasChildren
    - Return Bool Value, True if Document type has children document type.

These are few mostly used API. To change value using API you can use visual studio. If field has "Set" methods you can change value using API.
Example:

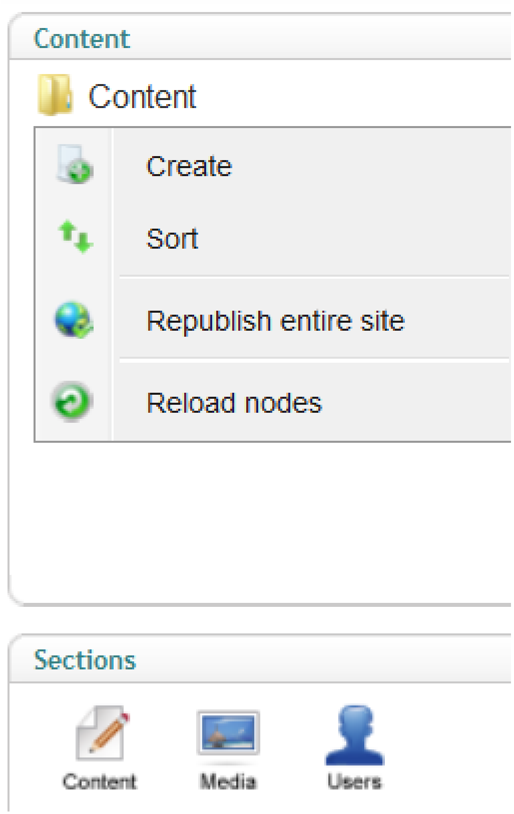
```
DocumentType.GetByAlias("NewsContainer").Alias = "newNewsContainer";
```

It will change alias of News Container to 'newNewsContainer'.

Chapter 04: Content

Content:

Content is one of the Section menu item in umbraco. Backend user will have access to content area. It contains articles or data that created by Editor based on document type. Lets Familiar with Envornment of Content if you Right Click over Content you can see different options. See captured screen bellow



Create: To create new content

Sort: to sort content items that will effect to end users. Top items will show first so on.

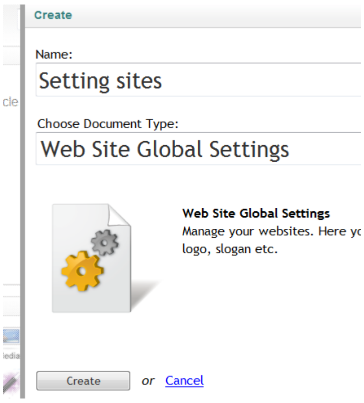
Republish Entire Sites: it will publish whole sites again, it will generate Umbraco.config file inside “App_Data” directory. Most of time if you move you Umbraco websites into different Deployment Environment you need to “Republish Entire sites” because umbracon.config is temporary files create by Umbraco system. Note that it will not publish unpublished article.

Reload nodes: To clear cache on server side, client sites and recreate new cache.

Creating Content:

It's a very simple process, if you click on Create it will show up new windows promoting you to type name of content and selecting option of Document Type in drop down list. After clicked on "Create" will create content.

Name: name of content and it's not the title of article that will show up in end users. It's a name of content and only used in Content section, it will not show up to end users.



Name:

Setting sites

Choose Document Type:

Web Site Global Settings

Web Site Global Settings

Manage your websites. Here yo
logo, slogan etc.

It's time to create Content for Vcard "SemicProvider" sites. First let's start with creating content for global settings. *I will show you only for global settings, for rest of content you can repeat similar process.*

Instruction: Right click on Content and Click on "Create" provide following information

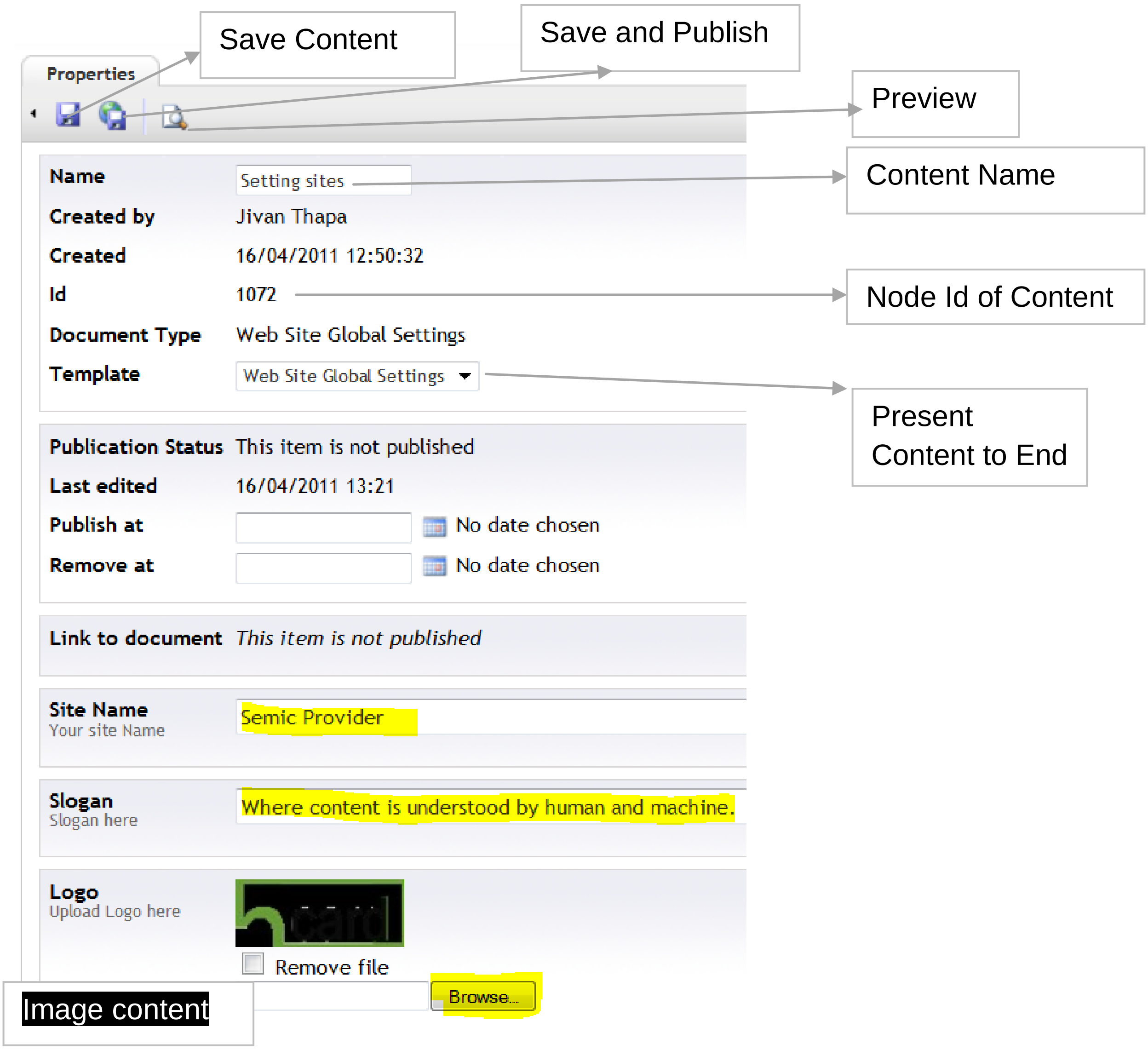
Name: "Setting sites"

Document Type: select "Web Site Global Settings" from drop down list.

Press "Create"

It will ask more information based on document type. Provide following info and press "Save" button. See highlighted color text. See captured screen "Image content"

Image: Content



Name Setting sites
Created by Jivan Thapa
Created 16/04/2011 12:50:32
Id 1072
Document Type Web Site Global Settings
Template Web Site Global Settings

Publication Status This item is not published
Last edited 16/04/2011 13:21
Publish at No date chosen
Remove at No date chosen

Link to document *This item is not published*

Site Name Your site Name: Semic Provider

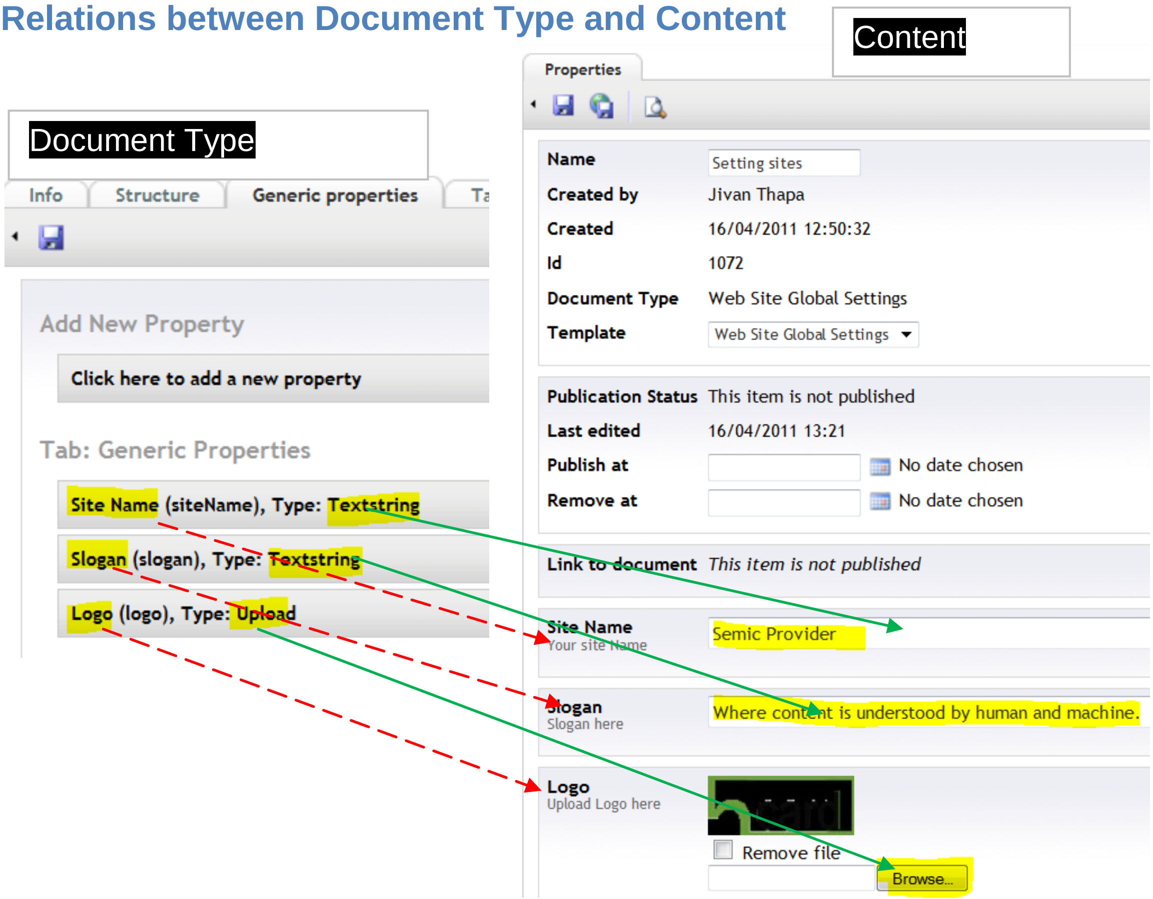
Slogan Slogan here: Where content is understood by human and machine.

Logo Upload Logo here
□ Remove file

Did you remember Site Name, Slogan and Logo were created at Document Type “Web site global settings”? We added type properties Textstring for Site name and Slogan and type:Upload for Logo. Here in Content page we can type site name and slogan because we selected type:Textstring and we can upload image in logo because we selected type:Upload in Document Type “Web Site Global Settings”. Now I hope you understand concept of Document Type and relation between Content and Document Type:

Notes: Create content for “Developer”, “Solution” and “Semic” using document type “General Content”. It’s a same process as we created just now. After that you will be more familiar with Creating Content in Umbraco sites and relation of Document Type and content. Let’s See next image relations between content and Document Type.

Relations between Document Type and Content



Document Type

Info | Structure | Generic properties | Ta

Add New Property

Click here to add a new property

Tab: Generic Properties

Site Name (siteName), Type: Textstring

Slogan (slogan), Type: Textstring

Logo (logo), Type: Upload

Content

Properties

Name	Setting sites
Created by	Jivan Thapa
Created	16/04/2011 12:50:32
Id	1072
Document Type	Web Site Global Settings
Template	Web Site Global Settings ▾
Publication Status	This item is not published
Last edited	16/04/2011 13:21
Publish at	No date chosen
Remove at	No date chosen
Link to document	*This item is not published*
Site Name Your site Name	Semic Provider
Slogan Slogan here	Where content is understood by human and machine.
Logo Upload Logo here	□ Remove file Browse...

Red Long-dash arrow: Name of Properties
Green arrow: Type of Properties

To understand relation between Document Type and Content may be difficult for first time but when you practice more it will be easy to understand. See two images above and follow red and green arrow telling relation between Document Type and Content.

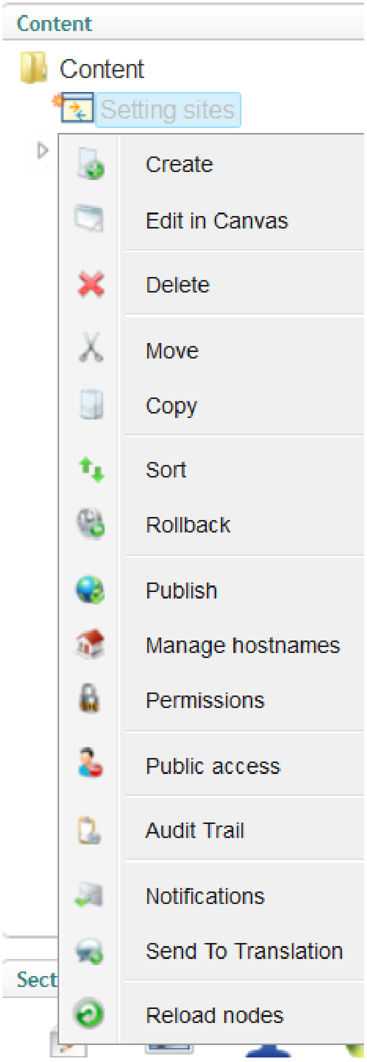
Delete Document Type*:* if you delete Document Type all associated content will be also deleted from websites, it's not possible to undo or restore Document Type in case delete.

Update Document Type Properties: you can change document type property for example you can change property type from type Simple Editor to type Rich Text Editor, data will not be lost because both are related type, both editor allows editor to edit text as simple or rich text environment. When you change type from type Textstring to type Label or type Upload, in that case all content modified after document type property has changed will lost previous data because type are not similar. So changing property type may cost data lost.

Delete Content: you can delete content. Deleted items will moved to Recycle Bin so if you accidently delete any content no need to feel panic you can easily restore from Recycle bin directory.

Explore content Properties:

There is more option for content that we created on Content section based on document type. Let's right click on "Setting site" content and explore more options. See captured screen.



Create: Create new child article example News Container.

Edit in Canvas: will allows editor to edit content in Frontend using browser.

Delete: Move content to Recycle Bin

Move: Move content from one place to another place, move will delete content from old location.

Copy: Copy content, Copy will not delete content from old location

Sort: arrange content by drag and drop to sort content, move up to display first and move down for low priority.

Rollback: allows to rollback content based on previous changed,

Manage Hostnames: will not cover in this basic book. It will used for multiple websites.

Permission: set up permission

Publich Access: Set up access level so only certain group Registered user can view or require to log in to view page

Audit Trill: list of history of content

Notifications: In big company you can set up notification in different condition to get notified when someone made change on article.

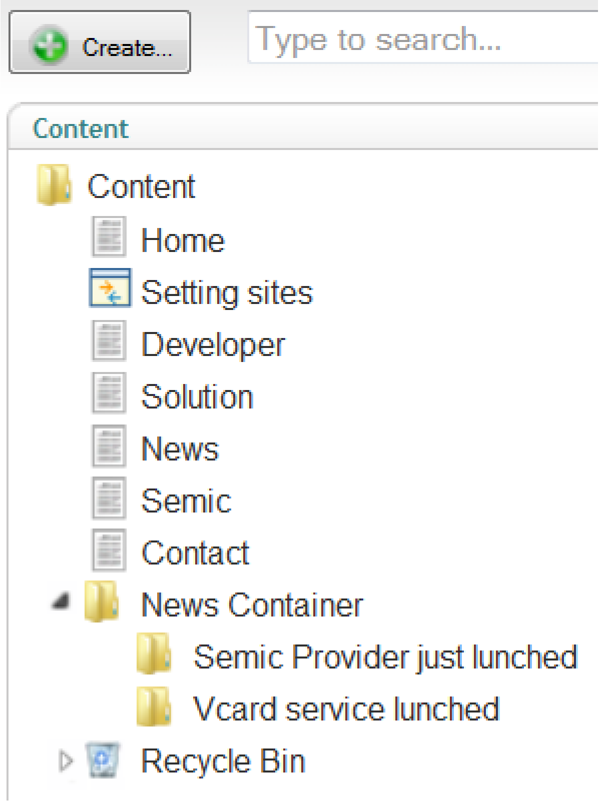
Send to Translation: used for multiple languages

Notes:

I am not going to create content for other document types because creating content is same for all document type just you need to select different document type when you create content. You can easily follow my process to create content for other document type.

Creating News Container and News Article are a bit different. First Create Article on "News Container" after that right click on just created content ("News Container" Document type content), then create news article. You can see Document Type "News Article" is only option to choose if you followed me. That is Allowed child nodetypes that we set up in Document Type "Structure" property of "News Container" Document Type.

I have created more content for Semic Provider site. You can see captured screen and compare with your version.

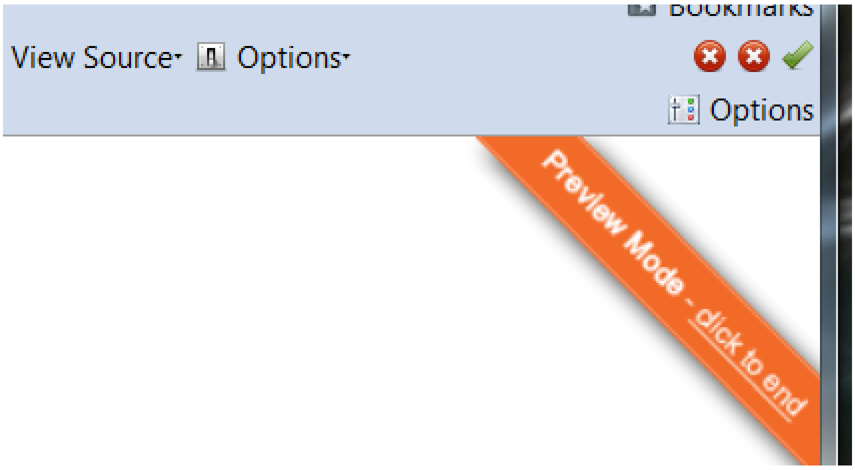


Six content;, Home, Developer, Solution, News, Semic and Contact are based on "General Content" Document Type.

News Container is based on "New Container" Doc Type and two child Content are based on News Article Doc Type

: Sort order. Here Home content is at first order so Umbraco will load Home Article first when you access website using domain name. Example www.jeevanthapa.com will load Home content first or by Default.

We are just working in backend part of Semic Provider site. If you preview article you see empty page. No content its because we have not touch any Templates yet, templates are responsible to output content or to present data to end users. I will discuss later or about templates. When you preview your content Umbraco will insert red strip with white color text "Preview Mode-click to end" on top right cornner of browser. Umbraco provides notification to you about your are viewing on Preview Mode. To exit click browser will close automatically.



For now you can't preview you data because associated templates are empty.

Content API:

In Umbraco code Content is understood by Document so that creating content in Umbraco Backend means executing Document code in Umbraco Core code inside server. Core code is programming language written in C# language. Content API is very important, if you want to create content using API, for example if you move sites from another websites to Umbraco CMS site, probably you want to get all data or content from old site as archive. If there is 1000 of articles in old site, it's not good idea to import content manually. You have to think some sort of automatic process to import content. Here is basic API for Umbraco Content.

namespace umbraco.cms.businesslogic.web contains Document Class. Creating Content means Executing Document.MakeNew() methods of Document Class. MakeNew() takes four parameters

- 1st parameter String Title: its title of Content, that title shown in Content Tree.
- 2nd Parameter DocumentType: Document Type, when you select document type Umbraco will automatically use that document type.
- 3rd parameter User: Umbraco does not allow to crate content without valid and has access user. When you logged in Umbraco automatically take your login user and use to create document.
- 4th parameter parent id: it's a hierarchy of content. If you want to create content at root level you must provide parent id -1. In Semic Provider content 'News Article' is child of 'news container' in that case all news article's content's parent id is Node id of "news Container". Its means if you provide -1 as parent id, Content will be created at root level. If you provide node id of content, it will create content as child of provided node id content.
- Example:
    - Document dt = Document.MakeNew("Developer",GeneralContent, User(0), -1);
    - dt.Publish();
        - Description: Create document name "Developer" on Document Type "GeneralContent". Use user 0. 0 is admin user id and always exist on Umbraco system. -1 will create Document at Root level.
        - Publish() method should be invoke to publish content or when you press Publish icon Umbraco execute publish() method

- Delete(): Delete method executed when you delete content.
        - Example: Document dt = new Document(id);

dt.delete(); id is node id of content. Delete() method will move content to Recycle Bin.

That is very basic of Umbraco API for Creating Content. For more details you can look inside umbraco.cms.businesslogic.web.document namespace. I strongly recommend looking source code and analyzing codes.

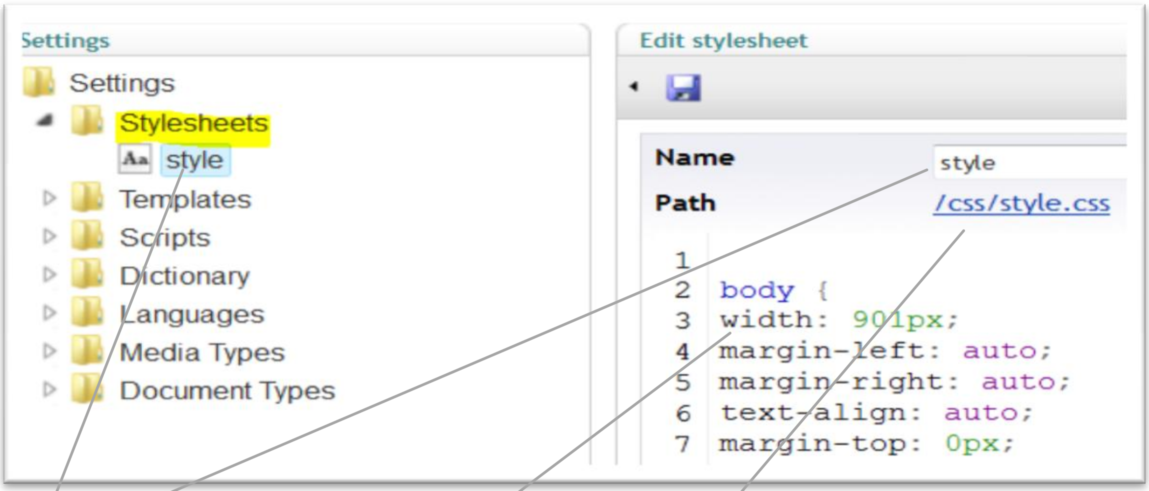
Chapter 05: Templates

Templates:

Purpose of templates is to present data to client or end users. Umbraco cms make totally separate visual presentation and content, to proof that concept you can preview any content we created before, we don't see any content on web it's just blank web page because we have not touch template means content is totally separated from visual presentation. Using templates we can present data or content to end users.

Templates are user friendly: The way you write code in template is same way as you write html code. You start with doc type deceleration, html, head, and body. No difference I will show you later. Only difference is to fetch data and insert into templates, fetching data can be done in different ways using alias, alias is created by Umbraco in Document Type creation period, use user control to fetch data, use xslt and xpath to fetch data using marco or use Razor Engine. I will focus in XSLT to fetch data from xml, use macro to render data to templates and present to end users.

CSS: the way we use css is not difference other than using in normal html page. It's very simple and you can copy your normal css text from text editor and paste to Umbraco CSS editing environment page. It will work as expected without any problem. Two ways are there to use CSS. One way is create css using Umbraco service or Create CSS file separate into normal text editor and place into "css" directory. You can also create your custom directory and use that, directory is not important; important is linking style sheet to templates. Linking style sheet is same as you use in normal html page.
<link rel="stylesheet" type="text/css" href="css/style.css"/> that is use to link document is style.css file exist inside css directory.No worries Umbraco will also show you path you can copy path to href easily. Lets look captured screen.



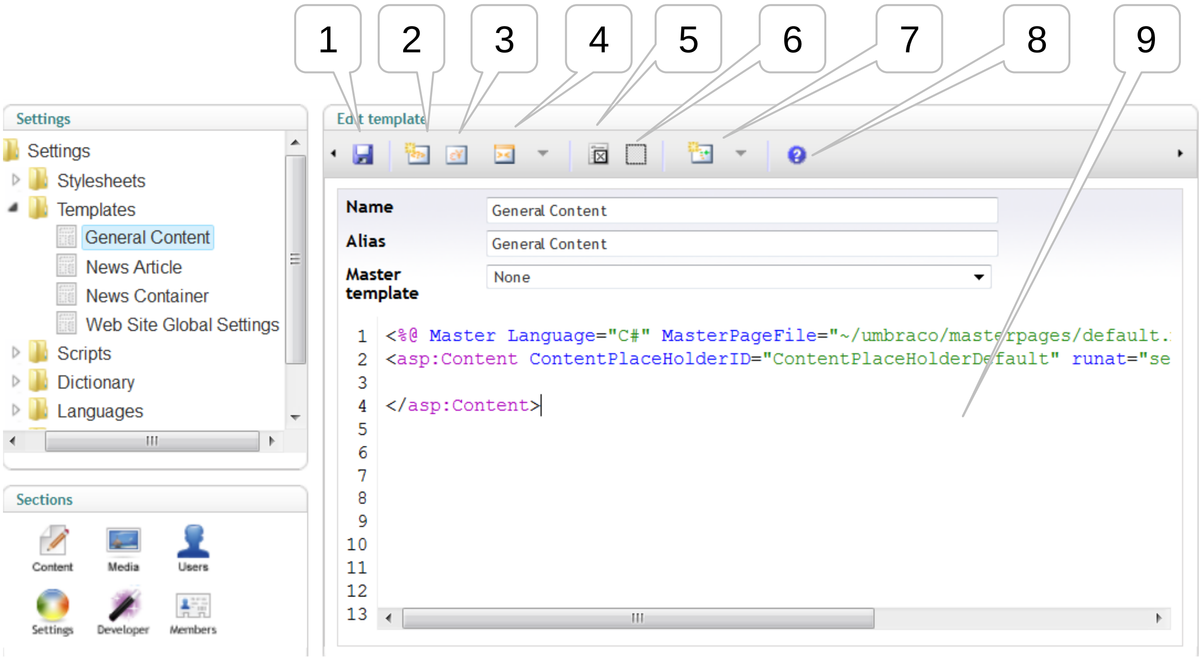
Name style
Path /css/style.css

```
body {
width: 901px;
margin-left: auto;
margin-right: auto;
text-align: auto;
margin-top: 0px;
```

Name: name of style sheet.
Start writing your CSS code here
<link rel="stylesheet" type="text/css" href="css/style.css"/>

Note: I will not cover CSS explanation in howtoumbraco basic. Style.css will be available in CD.

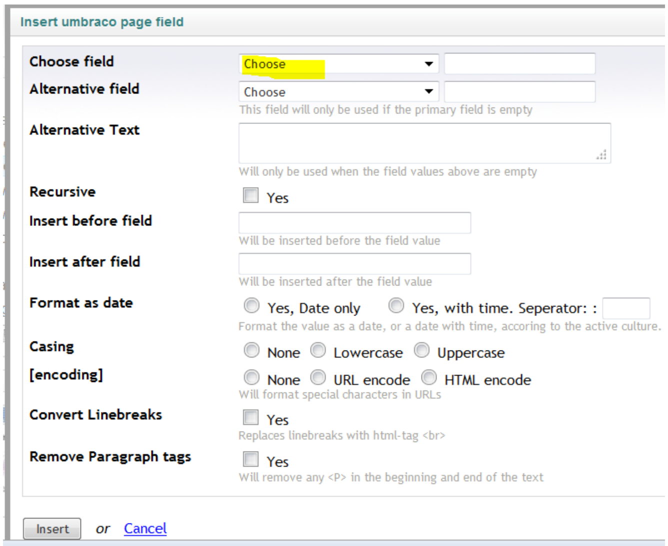
Template working Environment



Name: General Content
Alias: General Content
Master template: None

```
1  <%@ Master Language="C#" MasterPageFile="~/umbraco/masterpages/default.
2  <asp:Content ContentPlaceHolderID="ContentPlaceHolderDefault" runat="se
3
4  </asp:Content>
5
6
7
8
9
10
11
12
13
```

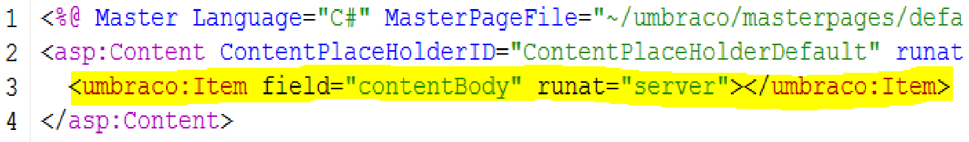
1- Save button, whatever you made change on template, you need to press Save button to save your work.
2- Insert Umbraco page field: use to insert page field. Using page field document type alias is used here to insert data into template. Umbraco will fetch and insert content on the fly. Alias that we had in document type are necessary in order to fetch content. By click on icon will list all page alias available to insert.



Choose field: Choose
Alternative field: Choose
This field will only be used if the primary field is empty
Alternative Text:
Will only be used when the field values above are empty
Recursive: □ Yes
Insert before field:
Will be inserted before the field value
Insert after field:
Will be inserted after the field value
Format as date: ○ Yes, Date only ○ Yes, with time. Seperator: :
Format the value as a date, or a date with time, accoring to the active culture.
Casing: ○ None ○ Lowercase ○ Uppercase
[encoding]: ○ None ○ URL encode ○ HTML encode
Will format special characters in URLs
Convert Linebreaks: □ Yes
Replaces linebreaks with html-tag
Remove Paragraph tags: □ Yes
Will remove any <P> in the beginning and end of the text

I am going to show one example: instruction: click on empty area at line 3 and Click on "Insert Umbraco Page field" number 2 icon. You can see new windows open with input options.

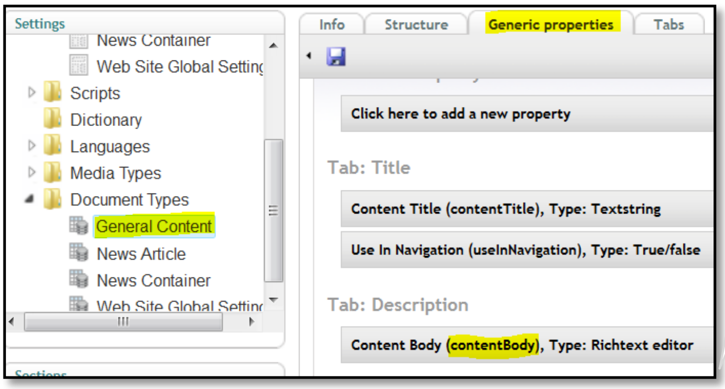
- ○ Choose field: here you can get list of alias that you created at document type. If you want to follow me select "contentBody" from dropdown list and press insert. It will insert filed. See captured screen bellow



```
1  <%@ Master Language="C#" MasterPageFile="~/umbraco/masterpages/defa
2  <asp:Content ContentPlaceHolderID="ContentPlaceHolderDefault" runat
3    <umbraco:Item field="contentBody" runat="server"></umbraco:Item>
4  </asp:Content>
```

- ○

- <umbraco:Item field="contentBody" runat="server"></umbraco:Item>
- <Umbraco:Item runat="server"> </Umbraco:item> will executed by Umbraco cms. In my case filed="contentBody", here Content that provided in Document type "General Content's property 'Content Body' " by editor on Content page will be inserted on the fly. So user will see content body data on web page. for more clear see diagram.



Tab: Title

Content Title (contentTitle), Type: Textstring

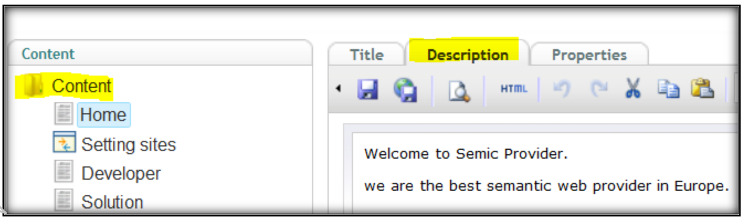
Use In Navigation (useInNavigation), Type: True/false

Tab: Description

Content Body (contentBody), Type: Richtext editor

Document Type: General Content
Tab: Description
Property Name: Content Body
Property Alias: contentBody
Type: Richtext editor

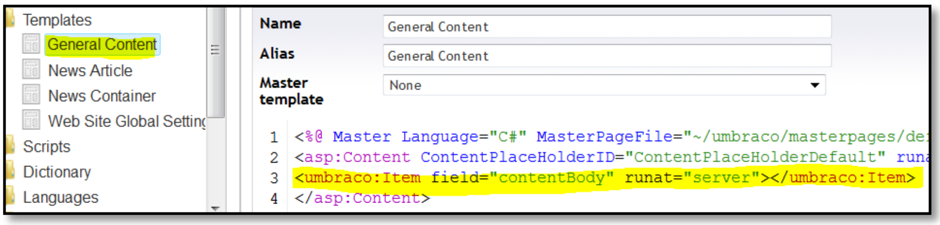
Content: Home
Description: welcome to semic provider.



Welcome to Semic Provider.

we are the best semantic web provider in Europe.

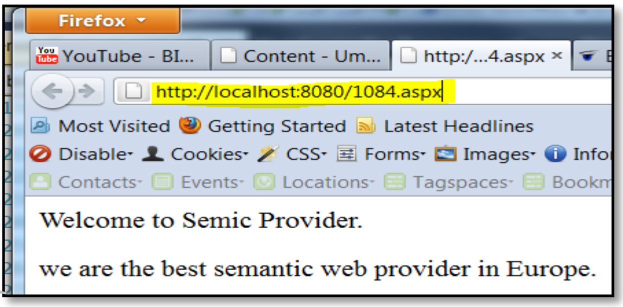
Field: contentBody



Name	General Content
Alias	General Content
Master template	None

```
<%@ Master Language="C#" MasterPageFile="~/umbraco/masterpages/de
<asp:Content ContentPlaceHolderID="ContentPlaceHolderDefault" runa
<umbraco:Item field="contentBody" runat="server"></umbraco:Item>
</asp:Content>
```

Preview of Content Home: Now Umbraco fetching content to Web sites. 1084.aspx where 1084 is node id of content Home.



Welcome to Semic Provider.

we are the best semantic web provider in Europe.

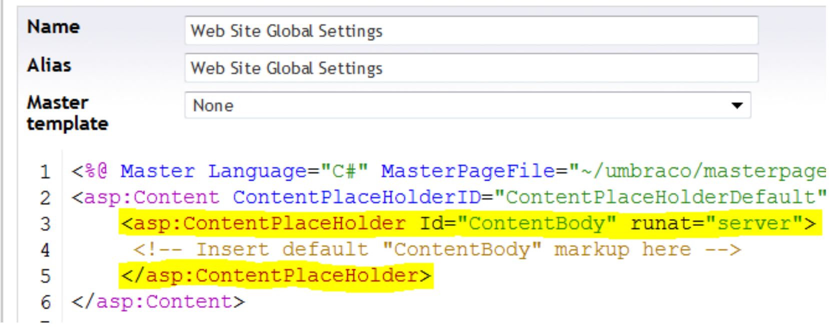
We created document type "General Content", added new property "Content Body" with alias "contentBody" and type "Richtext editor". In Content we add content on Richtext environment. In Template we use alias "contentBody" to fetch data that added on content using Richtext environment. It's a core concept of Umbraco for templates.

3- Insert Dictionary Field: Similar as Umbrcao page filed but it is used to insert Dictionary Items. Not covered in this “Howtoumbraco” basic.

4- Insert Macro: it’s a mostly used in templates and I will cover this Macro Chapter. Macro is very important chapter. Developer should not miss about Macro.

5- Insert Content Area Placeholder: We can easily create child templates in Umbraco, child has access to parents generic properties. Umbraco templates have great features “Inheritance Character” that child template can inject data to Parent template using “Insert Content Area Placeholder”, first you create Placeholder in parent template and in child template you insert content area. If you want follow me I want to use “Web Sites Global Settings” templates as Master templates or parent templates. Parent templates can have placeholder to inject data from child templates. Now let’s insert Content area Placeholder at “Web Sites Global Settings” templates.

- Instruction:
- Open “Web Site Global Setting” Template
- First Click on any place at working environment where you would like to insert content Place holder. Crusher should blink now, Click on “Insert Content area Placeholder” it will open new windows now type placeholder name “ContentBody” and Press Insert. See captured screen highlighted code are just added.
-

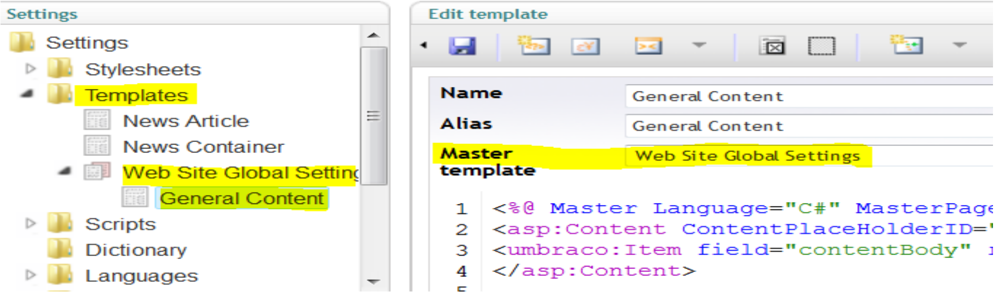


Name	Web Site Global Settings
Alias	Web Site Global Settings
Master template	None

```
<%@ Master Language="C#" MasterPageFile="~/umbraco/masterpage
<asp:Content ContentPlaceHolderID="ContentPlaceHolderDefault"
    <asp:ContentPlaceHolder Id="ContentBody" runat="server">
      <!-- Insert default "ContentBody" markup here -->
    </asp:ContentPlaceHolder>
</asp:Content>
```

Inheritance Properties: Umbraco template support inheritance properties so we can easily share global setting in sites for example. Navigation bar, heading, logo and web layout are global setting that will remain same look and feel for different content. “Web Sites Global Settings” template contains such global settings. Now I want to inject content from child templates on that place holder “ContentBody” I just created above. To inject data from child template first I have to assign child template. Click on General Content Template, in Master template change from “none” to “Web Site Global Settings” and Save “General Content” template. You can see General Content Templates now moved from root level to child template of “Web site Global Settings” Template. See captured screen

-



Name: General Content

Alias: General Content

Master template: Web Site Global Settings

```
<%@ Master Language="C#" MasterPage
<asp:Content ContentPlaceHolderID='
<umbraco:Item field="contentBody" 
</asp:Content>
```

6- Insert content Area: if parents' templates have content area placeholder, using insert content area we can insert content area placeholder at child templates. In my case general content is child of “Web Site Global Setting” template. Lets Insert Content area by clicking of “Insert Content Area” icon, select “ContentBody” placeholder and press insert. If you like to cancel, just press cancel. Umbraco content placeholder “ContentBody” inserted after Content place holder “ContentPlaceHolderDefault”. This content place holder is created default by Umbraco. Delete “ContentPlaceHolderDefault” from Template “General content” and Move Umbraco field “contentBody” inside content Placeholder “ContentBody”. Remember to save. If you follow me “General Content” Template code should be like bellow.

```
<%@ Master Language="C#"
MasterPageFile="~/masterpages/WebSiteGlobalSettings.master"
AutoEventWireup="true" %>
<asp:Content ID="Content2" ContentPlaceHolderId="ContentBody"
runat="server">
<!-- Insert "ContentBody" markup here -->
<umbraco:Item ID="Item1" field="contentBody"
runat="server"></umbraco:Item>
</asp:Content>
```

- Remember to preview your content after changed in your template.

7- Insert Inline Razor Macro: Razor engine is used if you don't want to use Xslt to fetch data from xml file. Not covered in this “HowtoUmbraco Basic”.

8- Help: it will open documentation. Remember to read it if you need help.

9- Working ‘Environment or area: Here you add all your template coding. Template <%@ Master Language= "C#" MasterPageFile= " ~/umbraco/masterpages/default.master" AutoEventWireup="true" %>

Description: all templates are inherited by default.master template, it's a Umbraco default master template.

Content Placeholder: all your html code should go inside Content place holder. If you want to follow me copy and paste code bellow in “Web Site Global Settings” Templates. Note: I am not going to explain code because here nothing new just added html tags. You can look yourself.

“Web Site Global Settings” Templates:

Note: if you are copying code from here, do not copy <%@ Master Language="C#" MasterPageFile="~/umbraco/masterpages/default.master" AutoEventWireup="true" %>

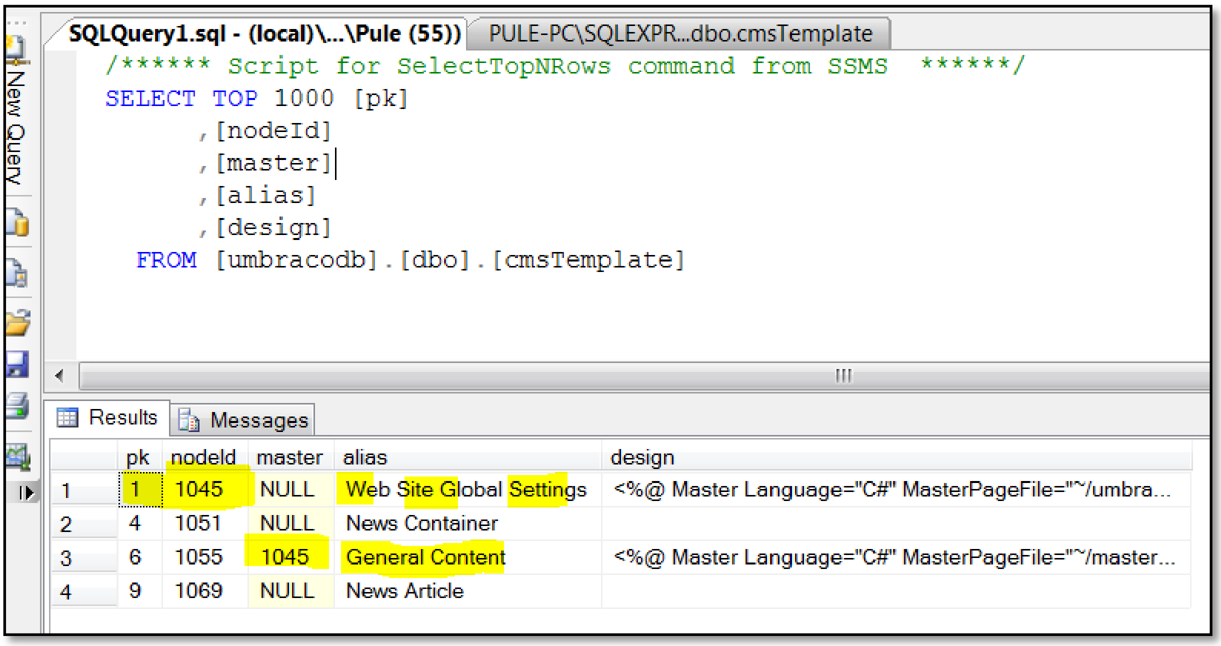
line because Umbraco will insert that line automatically in that case Umbraco CMS throw out yellow screen of death with Error message “ **Parser Error Message:** There can be only one 'master' directive.” To solve this problem make sure there is only one master page template exist on that template. If you want to follow me your “Web Site Global Settings” template code should looks like bellow.

```
<%@    Master    Language="C#"    MasterPageFile="~/umbraco/masterpages/default.master"
AutoEventWireup="true" %>
asp:Content ContentPlaceHolderID="ContentPlaceHolderDefault" runat="server">
<html>
<head>
  <link rel="stylesheet" type="text/css" href="css/style.css"/>
  </head>
<body>
 <div class="head">
  <div class="top">
      <div class="title">

      </div>
   </div>
  </div>
  <div class="navi">
    <div class="navigation">
      </div>
  </div>
  <div class="middle">
    <div class="content" >
      <asp:ContentPlaceHolder Id="ContentBody" runat="server">
        <!-- Insert default "ContentBody" markup here -->
      </asp:ContentPlaceHolder>
    </div>
   </div>
</body>
</html> </asp:Content>
```

Template API:

Template API can be found using namespace `using umbraco.cms.businesslogic.template;`

- Manual work Create Template:
    - That manual work will execute Template.MakeNew(name, u);
    - Name: name of Template
    - U is user. Umbraco will automatically assign user when you logged in backend. If you want to create Template using API you should provide valid user, user should have access to create Template otherwise Umbraco will throughout exception.
- Manual work Delete
    - It will execute Template.Delete() void methods.
    - It's not possible to restore deleted templates.
- Setting master Template:
    - Umbraco will execute Template. MasterTemplate = Template.Id;
    - Where Template.Id is node id of parent Template.
- Save:
    - Save button will execute Save() methods to save work. It will effect to Database see captured screen bellow as reference. I will not explain here but its not difficult to understand if you know basic knowledge of Sql server and Database.


New Query
SQLQuery1.sql - (local)\...\Pule (55))
PULE-PC\SQLEXPR...dbo.cmsTemplate
Results
Messages

```
/****** Script for SelectTopNRows command from SSMS  ******/
SELECT TOP 1000 [pk]
      ,[nodeId]
      ,[master]
      ,[alias]
      ,[design]
  FROM [umbracodb].[dbo].[cmsTemplate]
```

	pk	nodeId	master	alias	design
1	1	1045	NULL	Web Site Global Settings	<%@ Master Language="C#" MasterPageFile="~/umbra...
2	4	1051	NULL	News Container	
3	6	1055	1045	General Content	<%@ Master Language="C#" MasterPageFile="~/master...
4	9	1069	NULL	News Article	

These results are from Database name 'umbracodb' table 'cmsTemplate'. In your case database name can be deference as compared to mine.

Notice:
I am not using anymore "insert page field" ways to fetch data in templates because if I fetch data using document type alias it will fetch data only if template are checked in Document type allowed templates. If templates are not associated with document type, Data will not be fetching using "Insert Page field" technique so that I am going to continue to fetch data using Xslt and use macro to render content to templates.

Chapter 06: XSLT

XSLT:

Xslt provides fast data fetching processing from xml file. All published content in your site are structured in xml format by Umbraco system. Xslt is the way to transfer xml data to html format, Xslt and Xpath are used together to provide rich service. Xpath is used to move pointer back and forth to crawl content in xml file or to move pointer from one node to another node. My assumption is you already familiar with Xslt and Xpath, I will directly going to show creating and using Xslt.

Using XSLT:

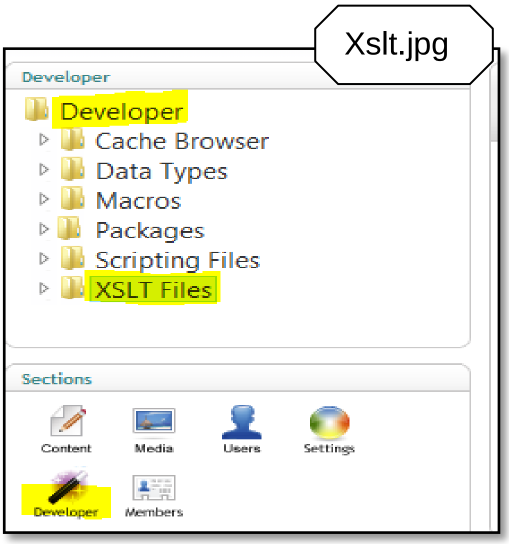
Xslt transfer xml to html, using Xslt we are fetching data from xml and passing to template in Umbraco site. We need to create associated macro first, link that macro with Xslt and Use that macro in Template. Here macro works as bridge between Xslt and Template. Umbraco is user friendly so you don't have to worry about macro. Each time you check "Create Macro" when you create new Xslt, Umbraco will automatically generate Macro for you.

Let's starts to work with Xslt

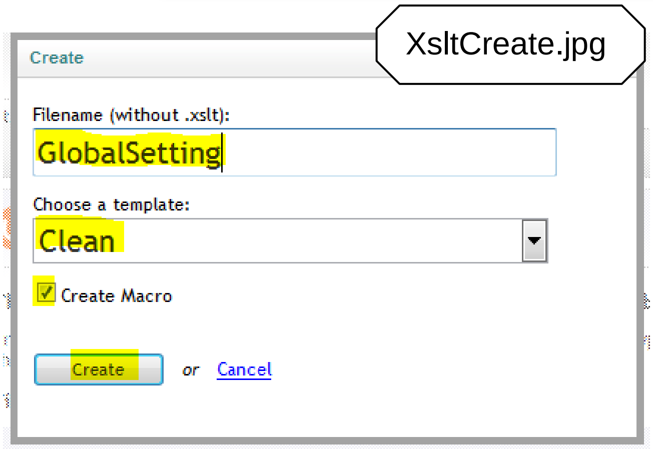
Instruction:

- Click "Developer" in Sections menu. "XSLT Files" menu is available inside Developer menu items. See captured screen: "Xslt.jpg".
- Right click on "XSLT Files" and Click "Create". Umbraco open new windows: if you want to follow me provide these information
  - Name: "GlobalSetting"
  - Choose a Template: Clean
  - Checked: Create Macro
- Press "Create" see captured screen bellow "XsltCreate.jpg"

Xslt.jpg



XsltCreate.jpg



Filename (without .xslt): GlobalSetting

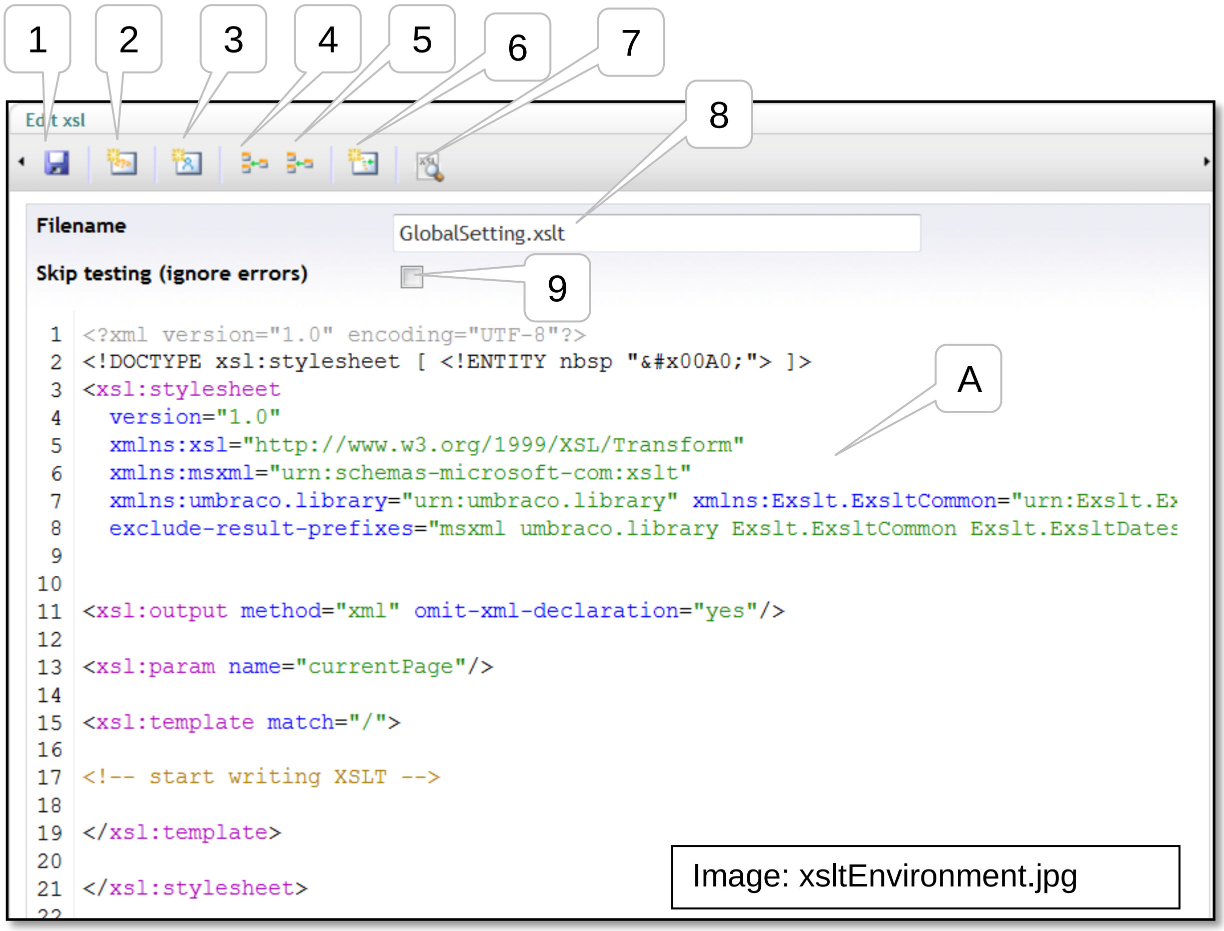
Choose a template: Clean

☑ Create Macro

"GlobalSetting.xslt" file must be created inside XSLT file menu at Developer section.

Purpose of creating Globalsetting.xslt is to fetch Site name, site title and logo.

Xslt working environments:



Filename GlobalSetting.xslt

Skip testing (ignore errors) □

```
1  <?xml version="1.0" encoding="UTF-8"?>
2  <!DOCTYPE xsl:stylesheet [ <!ENTITY nbsp "&#x00A0;"> ]>
3  <xsl:stylesheet
4    version="1.0"
5    xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
6    xmlns:msxml="urn:schemas-microsoft-com:xslt"
7    xmlns:umbraco.library="urn:umbraco.library" xmlns:Exslt.ExsltCommon="urn:Exslt.E
8    exclude-result-prefixes="msxml umbraco.library Exslt.ExsltCommon Exslt.ExsltDates
9
10
11 <xsl:output method="xml" omit-xml-declaration="yes"/>
12
13 <xsl:param name="currentPage"/>
14
15 <xsl:template match="/">
16
17 <!-- start writing XSLT -->
18
19 </xsl:template>
20
21 </xsl:stylesheet>
```

Image: xsltEnvironment.jpg

A- Coding area: here you can type Xslt code.

1- **Save** Button To save work after changed
2- **Insert xslt:value of** : use to insert xslt value, you can also select prevalue.
3- **Insert xslt:variable :** use to insert xslt variable
4- **Insert xslt:if** : use to insert condition statement if
5- **Insert xslt:for-each :** use for looping purpose
6- **Insert xslt:choose :** use to insert choose condition statement
7- **Visualize Xslt:** it's a big feature in Umbraco. Using visualizes Xslt you can see what output will be made by your Xslt Code on templates. You can also choose different Content and see what will be output.
8- **Name** of xslt file
9- **Skip testing:** if you checked on Skip Testing, Umbraco will not validate your Xslt code so you will not see any error message.

Coding Area:

- It starts with xml deceleration and version, it is using xml version 1.0 and encoding utf-8. *See at line 1 Image: xsltEnvironment.jpg*

- You define replacement text yourself at Xslt using DTD[14]. To use that features you have to have DTD schema knowledge. By default Umbraco inserted one entity.

<!DOCTYPE xsl:stylesheet [<!ENTITY nbsp "&#x00A0;">]>
here Enitity nbsp can be used anywhere in xslt. On execution time Umbraco replace nbsp with text '&#x00A0;' that we defined at DOCTYPE Entity. *See line 2 at Image xsltEnvironment.jpg*

- Stylesheet Declaration: You can extend Xslt further more using .NET Class libraries. In some cases Xslt may not enough to fetch data in such cases you can extend Xslt. You need to add your Xslt extension here in order to use it.

.NET Class Registration
Adding Assembly to Umbraco sites:
Open ~/config/xsltExtensions.config file, here you add Assembly first.
<ext assembly="assemblyName" type="assemblyName.namespace.type" alias="custom">
Assembly **= location of your files**
Type **= namespace name**
Alias **= you use that alias at Xslt to refer assembly.**

Add alias in xslt stylesheet.
Xmlns:LocalName = alias from xsltExtesion registration
LocalName is used inside Xslt templates
Alias = alias that you define when you registered assembly name at ~/config/xsltExtesion.config file
Now you are ready to use .NET Class library in Xslt.
exclude-result-prefixes="Assembly Alias"
Use exclude to prevent Xslt from outputting namespace name.
Note: ~ means your Umbraco root installation directory level.

- <xsl:output method="xml" omit-xml-declaration="yes"/>

It will omit xml deceleration output
Let's start to work with Umbraco Xslt

- <xsl:param name="currentPage"/>
- It's a very use full param in Umbraco. Umbraco will insert current page data. Current page means page that Templates, Content and Document Type using specific Xslt file in other words currentPage means current page. Value of currentPage depends on article or Content you are visiting now. Let's try to output currentPage value and see what will be output.
- Template Match

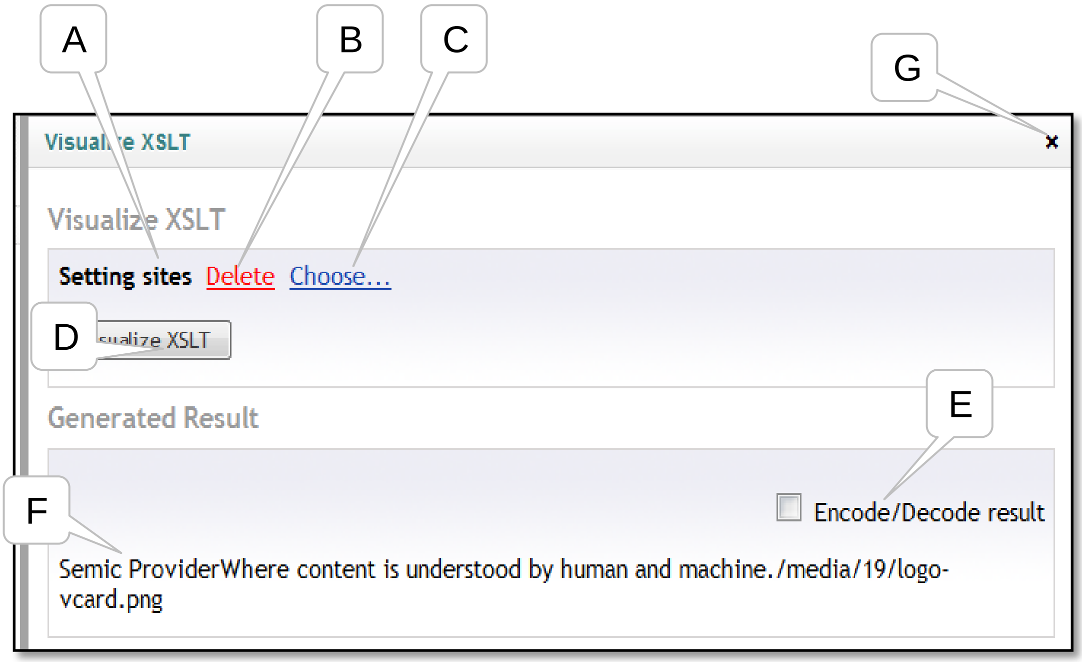
```
<xsl:template match="/">
      <xsl:copy-of select="$currentPage" />
</xsl:template>
```

  - All your logic or html output you code between opening and closing tags of <xsl:template>

[14] Document Type Definition

- Here we use xsl:copy-of select="$currentPage" to output current page data.

- To see output value use "visualize Xslt" button at top-right corner of Edit xsl working environment. Make sure you choose first Content using "Choose…" menu. Select Content after press "Visualize XSLT" button. You should see some data if content has some data. See captured screen bellow.



Semic ProviderWhere content is understood by human and machine./media/19/logo-vcard.png

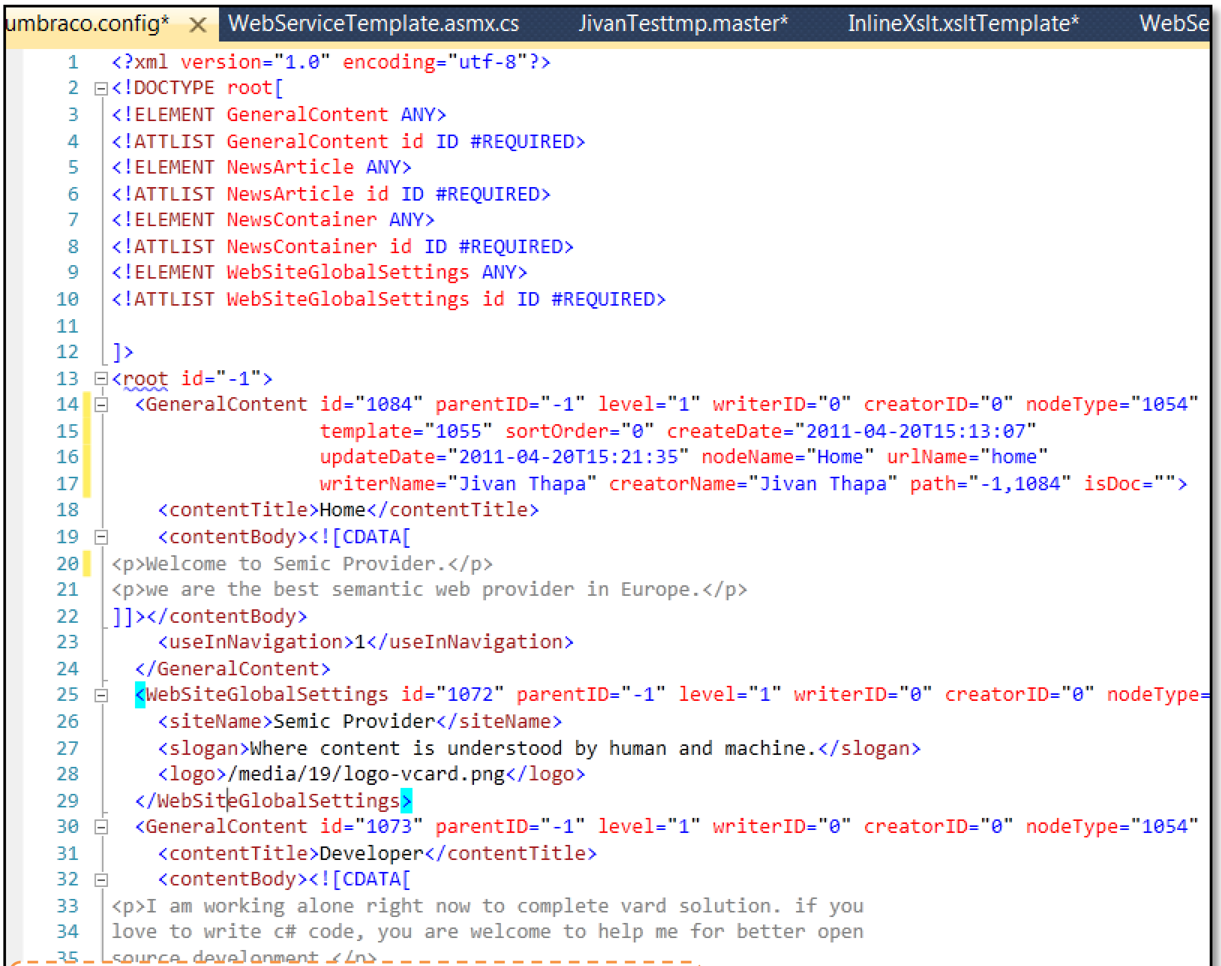
A- Title of Content: first time it must be empty.
B- Delete: To delete selected Content. It will not delete content from Content Section, it will only delete from Visualize Xslt service.
C- Choose..: you can easily change or alter Content to visualize XSLT and see different output based on XSLT code.
D- Visualize Xslt Button: Press to visualize XSLT
E- Encode: Checked if you want to see xml formatted data, uncheck to see only content.
F- Output of XSLT: it's a result; it's an output of your Xslt code, its output currentPage value. Same value will insert into templates.
G- Press "x" to close Visualize XSLT service.

Xml File umbraco.config

I mentioned Xslt use to fetch data from xml file. You must wonder about what xml file? Where is that xml file? Why it's in xml file? And more questions.

- Umbraco generates Umbraco.config temporary file it's structured in xml format, every time when you made change on content or document type. Umbraco remind you to publish content. It means, when you made change on content Umbraco made change only on Database. Umbraco does not fetch data from database to present data to end users unless you extend it using other programming language. One reason is it is slow process to fetch data from database as compared to fetching data from xml file. Database may store in separate machine or separate location in server that need to travel data that cause slower process. Umbraco use great idea generating temporary xml files, that xml files cached in memory. Using temporary xml file it's dramatically faster to fetch data using Xslt. To see temporary xml generated file you need to locate directory “App_Data” in website root level and find umbraco.config file. “~/App_data/umbraco.config”

- Open umbraco.config file using Note pad or other software, if you open with Visual studio it will easy to navigate. See captured screen. Your version can be difference.



```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE root[
<!ELEMENT GeneralContent ANY>
<!ATTLIST GeneralContent id ID #REQUIRED>
<!ELEMENT NewsArticle ANY>
<!ATTLIST NewsArticle id ID #REQUIRED>
<!ELEMENT NewsContainer ANY>
<!ATTLIST NewsContainer id ID #REQUIRED>
<!ELEMENT WebSiteGlobalSettings ANY>
<!ATTLIST WebSiteGlobalSettings id ID #REQUIRED>

]>
<root id="-1">
  <GeneralContent id="1084" parentID="-1" level="1" writerID="0" creatorID="0" nodeType="1054"
                  template="1055" sortOrder="0" createDate="2011-04-20T15:13:07"
                  updateDate="2011-04-20T15:21:35" nodeName="Home" urlName="home"
                  writerName="Jivan Thapa" creatorName="Jivan Thapa" path="-1,1084" isDoc="">
    <contentTitle>Home</contentTitle>
    <contentBody><![CDATA[
<p>Welcome to Semic Provider.</p>
<p>we are the best semantic web provider in Europe.</p>
]]></contentBody>
    <useInNavigation>1</useInNavigation>
  </GeneralContent>
  <WebSiteGlobalSettings id="1072" parentID="-1" level="1" writerID="0" creatorID="0" nodeType=
    <siteName>Semic Provider</siteName>
    <slogan>Where content is understood by human and machine.</slogan>
    <logo>/media/19/logo-vcard.png</logo>
  </WebSiteGlobalSettings>
  <GeneralContent id="1073" parentID="-1" level="1" writerID="0" creatorID="0" nodeType="1054"
    <contentTitle>Developer</contentTitle>
    <contentBody><![CDATA[
<p>I am working alone right now to complete vard solution. if you
love to write c# code, you are welcome to help me for better open
source development.</p>
```

Image: Umbracoconfig.jpg

Relationship between Document Type and Umbraco.config Xml

See Image: Umbracoconfig.jpg

Line 1: xml declaration

Line 2 to line 10: DTD schema information

Line 13: Umbraco Root Element. All Elements go inside root node. Root is parent element for all.

Line 14: Umbraco starts generating xml elements and attributes based on our document type and content. Did you notice Alias that used in Document type is Elements here?

- GeneralContent is alias of Document Type “General Content”. All attributes inside “GeneralContent” element are inserted by Umbraco automatically.
- Now let’s look child element of “GeneralContent” Element. Two more element “contentTitle” and “contentBody” also created. If you notice that we added two new property “contentTitle” and “contentBody” in Document Type “General Content”.
- Here in umbraco.config xml these new property structured as child of Document Type element tags.
- Probably you need time to understand, you have to go back to Document Type, see carefully all alias. Come back to umbraco.config xml file and analyze yourself that helps you to understand better.

Line 14: attributes: Umbraco generates few attributes automatically. Most attributes are self explanatory so easier to understand. I feel some attributes need to describe here for Umbraco New developers.

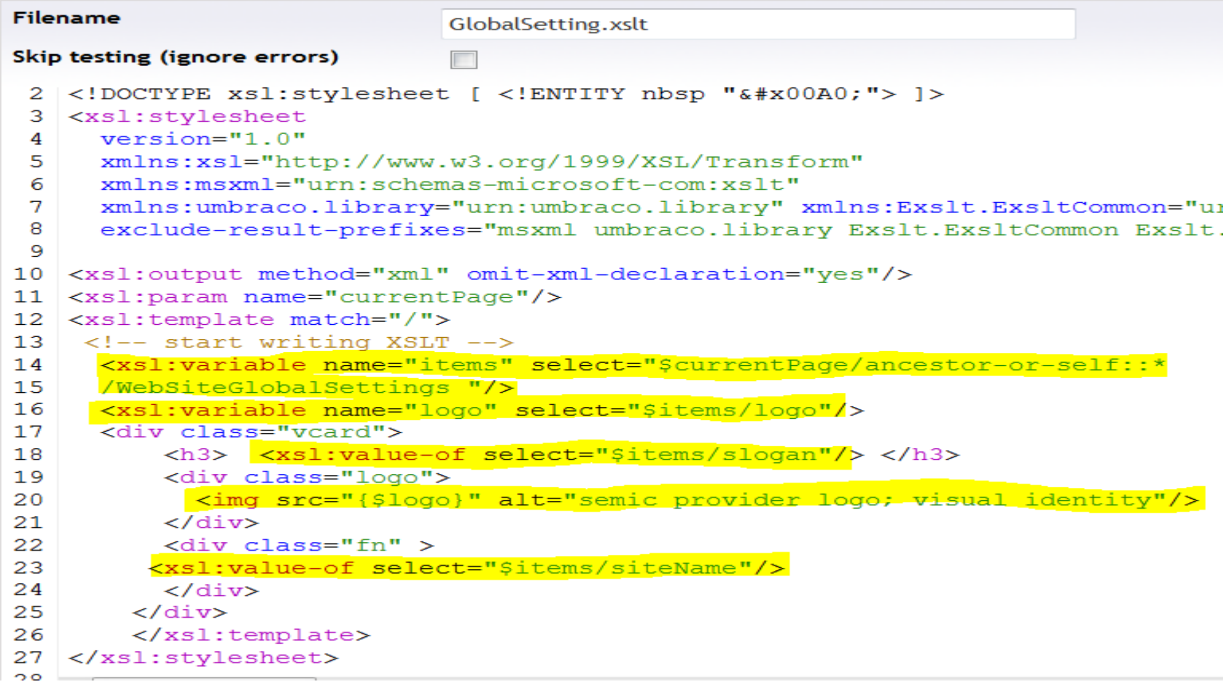
- **id** : it’s a node id of content each content has different node id
- **parentID**: node id of parent element. If you notice in Content API chapter we pass parent id -1 value. Means create content at root level in content but actually it means create content as child of root element. Umbraco assign root element id = -1 see line 13 Image: “Umbracoconfig.jpg;
- **level**: level of content. 1 means at root level if you create child content it will be level 2
- **nodeName**: It’s not a contentTitle that you provide on document type content title property. it’s a title of Content that appear at Content Tree Section.
- **urlName**: name of url, content access using urlName. Example /home.aspx will load contain id 1084. See line 14 Image: “Umbracoconfig.jpg;
- **nodeType**: node id of Document Type
- **How do you know Document Type Node id?**

  It’s a tricky. Just mouse over Document Type at left bottom of browser , browser show JavaScript link like this “javascript:openNodeType(1054);” number between () is node id of Document Type.

- **creatorID**: id of users who created content. 0 means administrator and always exist in Umbraco user, it has root level of access[15].

Fetching data:

Let's take real example for Semic provider site and be clear about use of Xslt. Semic provider has logo, title and slogan as global items. For global items[16], we need to fetch value from siteName, slogan and logo element *see line at 26, 27 and 28 respectively at Image:Umbracoconfig.jpg"*. Open GlobalSetting.xslt file and add highlighted code into GlobalSetting.xslt. See captured screen and Save Xslt.



Filename: GlobalSetting.xslt

Skip testing (ignore errors) □

```
<!DOCTYPE xsl:stylesheet [ <!ENTITY nbsp "&#x00A0;"> ]>
<xsl:stylesheet
  version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:msxml="urn:schemas-microsoft-com:xslt"
  xmlns:umbraco.library="urn:umbraco.library" xmlns:Exslt.ExsltCommon="u
  exclude-result-prefixes="msxml umbraco.library Exslt.ExsltCommon Exslt.

<xsl:output method="xml" omit-xml-declaration="yes"/>
<xsl:param name="currentPage"/>
<xsl:template match="/">
 <!-- start writing XSLT -->
  <xsl:variable name="items" select="$currentPage/ancestor-or-self::*
  /WebSiteGlobalSettings "/>
  <xsl:variable name="logo" select="$items/logo"/>
  <div class="vcard">
      <h3> <xsl:value-of select="$items/slogan"/> </h3>
      <div class="logo">
        <img src="{$logo}" alt="semic provider logo; visual identity"/>
      </div>
      <div class="fn" >
     <xsl:value-of select="$items/siteName"/>
      </div>
    </div>
    </xsl:template>
</xsl:stylesheet>
```

All html tags are not important, only important is highlighted line.

Description:

- `<xsl:variable name="items" select="$currentPage/ancestor-or-self::* /WebSiteGlobalSettings "/>`
  - Here $ represent variable in XSLT. variable name items will hold data from "WebSitesGlobalSettings" Element see line at 25 Image: "Umbracoconfig.jpg" page no: xx
  - currentPage will select current page.
  - ancestor-or-self:: move pointer back until root level.
  - * select all that match
  - / move one step further to child element.
  - WebsitesGlobalSettings is element name.
- `<xsl:variable name="logo" select="$items/logo"/>`
  - $items/logo will select logo element and hold data
- `<xsl:value-of select="$items/slogan"/>`
  - $items/slogan: select value from slogan element
  - `<img src="{$logo}" alt="semic provider logo; visual identity"/>`
  - Insert value of logo variable. Using $ and followed by name means use of variable, encapsulated data with {}. {} means code inside {} will executed on server by Umbraco CMS.

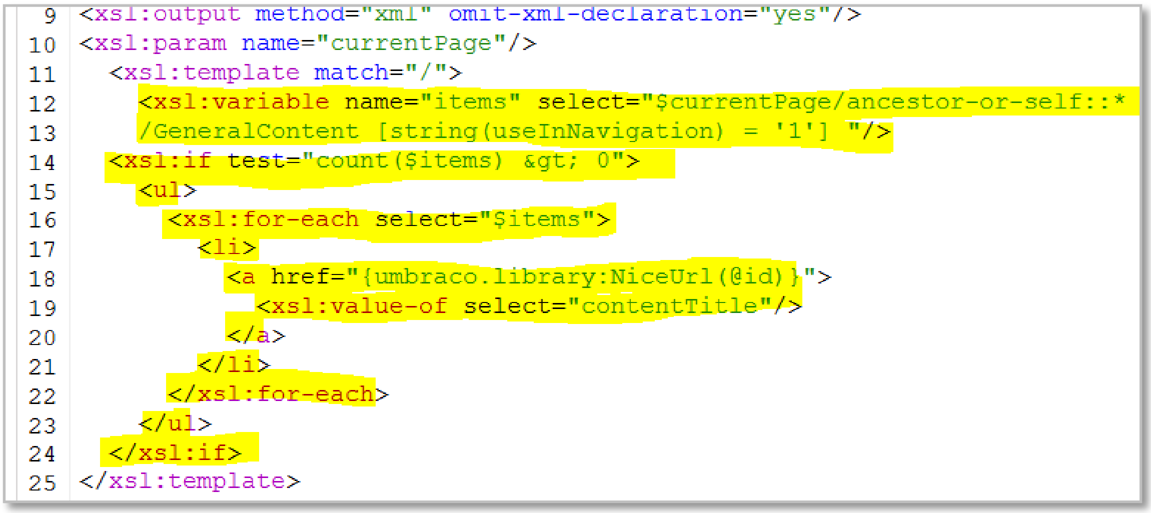
[15] Root level of access: access that can CRUD ; C-create, R-read, U-update, D-delete

[16] Items those are common for all. For example logo, website name.

- `<xsl:value-of select="$items/siteName"/>`
    - This line will select value of siteName element and output.

Note: alt tag in img element is not compulsory but I strongly recommend having alt tag. It will help for web accessibility and SEO.

Xslt for Navigation:



```
<xsl:output method="xml" omit-xml-declaration="yes"/>
<xsl:param name="currentPage"/>
  <xsl:template match="/">
    <xsl:variable name="items" select="$currentPage/ancestor-or-self::*
    /GeneralContent [string(useInNavigation) = '1'] "/>
  <xsl:if test="count($items) &gt; 0">
    <ul>
      <xsl:for-each select="$items">
        <li>
          <a href="{umbraco.library:NiceUrl(@id)}">
            <xsl:value-of select="contentTitle"/>
          </a>
        </li>
      </xsl:for-each>
    </ul>
  </xsl:if>
</xsl:template>
```

This is another Xslt for Navigation bar. Create new Xslt. Provide name navigation and Checked Create Macro. Copy highlighted text. Note that html is not that much important these are for CSS and layout purpose.

Description:

- [string(useInNavigation) = '1'] condition match means select all GeneralContent Element from ~/App_Data/umbraco.config file where useInNavigation checked boxed should be checked. Remember that we added new property at "General Content" Document Type. If editor wants to use general content in navigation bar. Editor should Checked mark on "useInNavigation" when s/he creates content.

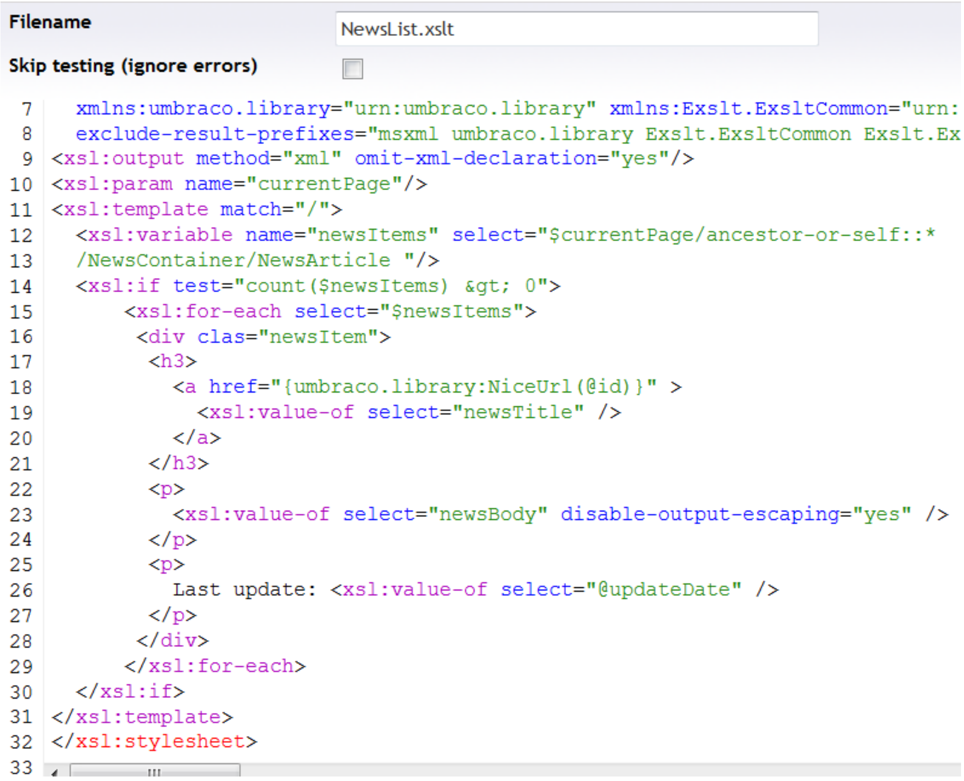
- Nice Url: umbraco.library:NiceUrl(@id)
    - NiceUrl(@node id) will generate user friendly nice Url. Remember that urlName are used example: home.aspx
    - @id: it's an attribute of selected element at ~/App_Data/umbraco.config xml file. In my case it's a node id of content based on "General Content" document type. @ sysmbole represent get value from attributes. See Xslt, Xpath and xml for details.

- <xsl:value-of select="contentTitle"/>
    - Select value from contentTitle element
- <xsl:for-each select="$items">
    - Used to loop all existing content
- <xsl:if test="count($items) &gt; 0">
    - Here &gt; means greater than. Test: means execute block of code only if result is more than zero. If items are less than zero no need to loop items.

Note: there is missing code in captured screen means missing code are generated by Umbraco Default and removed from here to reduce space.

New Xslt file for News List items.

It will be exercise for you I create NewsList.xslt file, here I just include captured screen without description try to describe yourself. It's not big but need to knowledge of Xpath. Main purpose of this Xslt is to fetch content from 2nd level menu contain. It will fetch News articles from ~/App_data/umbraco.config xml file. Captured screen are bellow.



Filename NewsList.xslt

Skip testing (ignore errors) □

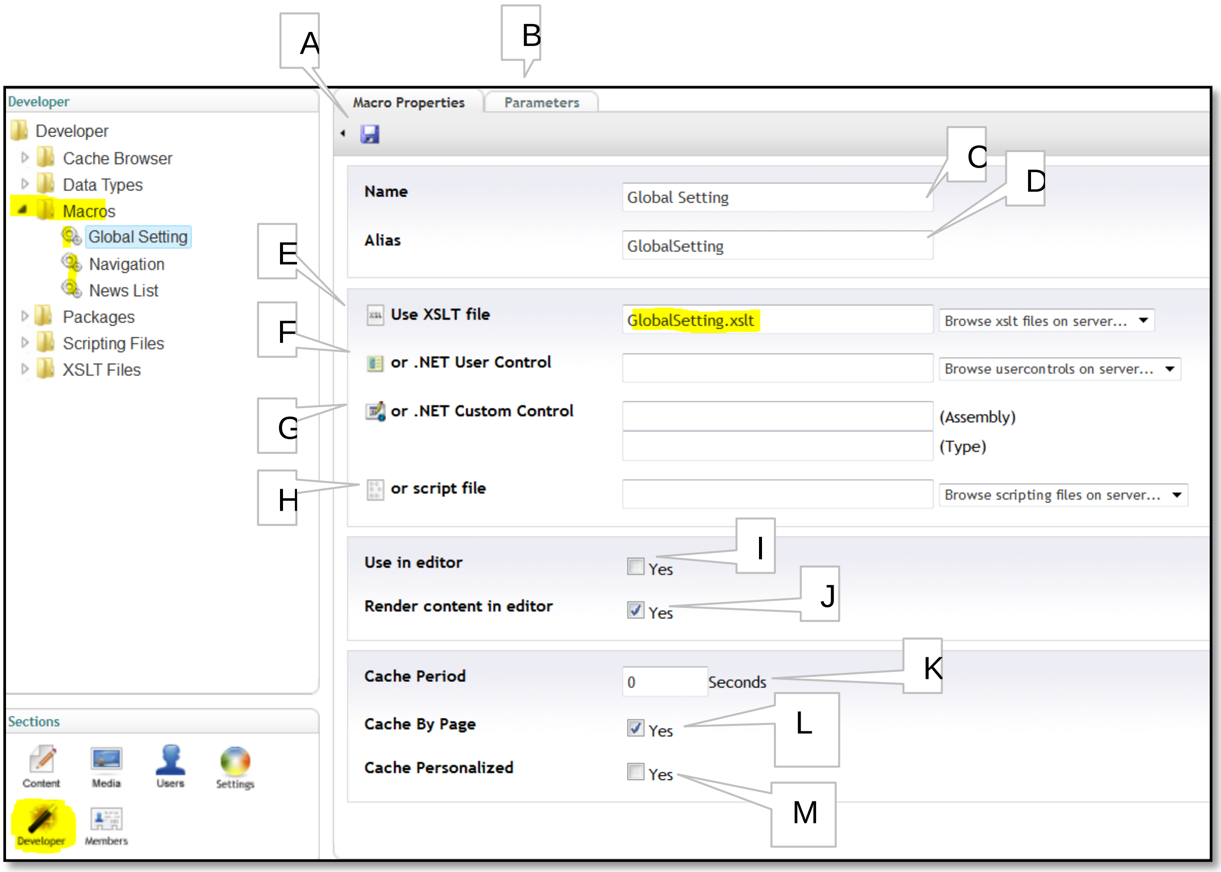
```
  xmlns:umbraco.library="urn:umbraco.library" xmlns:Exslt.ExsltCommon="urn:
  exclude-result-prefixes="msxml umbraco.library Exslt.ExsltCommon Exslt.Ex
<xsl:output method="xml" omit-xml-declaration="yes"/>
<xsl:param name="currentPage"/>
<xsl:template match="/">
  <xsl:variable name="newsItems" select="$currentPage/ancestor-or-self::*
  /NewsContainer/NewsArticle "/>
  <xsl:if test="count($newsItems) &gt; 0">
      <xsl:for-each select="$newsItems">
       <div clas="newsItem">
        <h3>
          <a href="{umbraco.library:NiceUrl(@id)}" >
            <xsl:value-of select="newsTitle" />
          </a>
        </h3>
        <p>
          <xsl:value-of select="newsBody" disable-output-escaping="yes" />
        </p>
        <p>
          Last update: <xsl:value-of select="@updateDate" />
        </p>
       </div>
      </xsl:for-each>
  </xsl:if>
</xsl:template>
</xsl:stylesheet>
```

Chapter 07: Macro

Macro:

Marco is use as bridge between xslt, user control or other languages and templates, using macro we can render content or data from data source to templates and templates are responsible to present data to end users. Macro is very important part in Umbraco.

Macro working Environments:



Name: Global Setting

Alias: GlobalSetting

Use XSLT file: GlobalSetting.xslt

or .NET User Control:

or .NET Custom Control: (Assembly) (Type)

or script file:

Use in editor □ Yes

Render content in editor ☑ Yes

Cache Period 0 Seconds

Cache By Page ☑ Yes

Cache Personalized □ Yes

Let me say something here before I start talking about marco working environments. I did not create macro here instead I checked "Create macro" option when I created three Xslt Files. If you followed me you should have same macro as me. If you did not checked "Create marco" option when you created Xslt, do not worry you can right click at Macro menu and Press "Create" it will open same windows as you are viewing now and start proving similar information like as in captured screen. it will be fine.

A- Save button to save your work or settings

B- Parameters: macro allows you to insert parameter into macro. Whenever you use that macro you should provide value to parameter based on parameter type you created. In Xslt you can use that parameters using <xsl:value-of

select=/macro/parameterAlis />. I will provide you reference to look more about Macro Parameters[17].

C- Name of Macro: you provide name, if you checked “Create Macro” during Xslt creation time Umbraco will automatically assign name.

D- Alias of macro, alias are available to insert macro into templates and content.

E- Currently all three macro I have will associated with xslt. Here you can select your Xslt file that you want to use with macro

F- If you create user control. You should select user control. I will show you later

G- Use .NET custom control assembly and type.

H- Or Use other script files for example Razor Engine.

I- Checked if you want to use in Content Editor

J- Show preview fetch data on Content Editor.

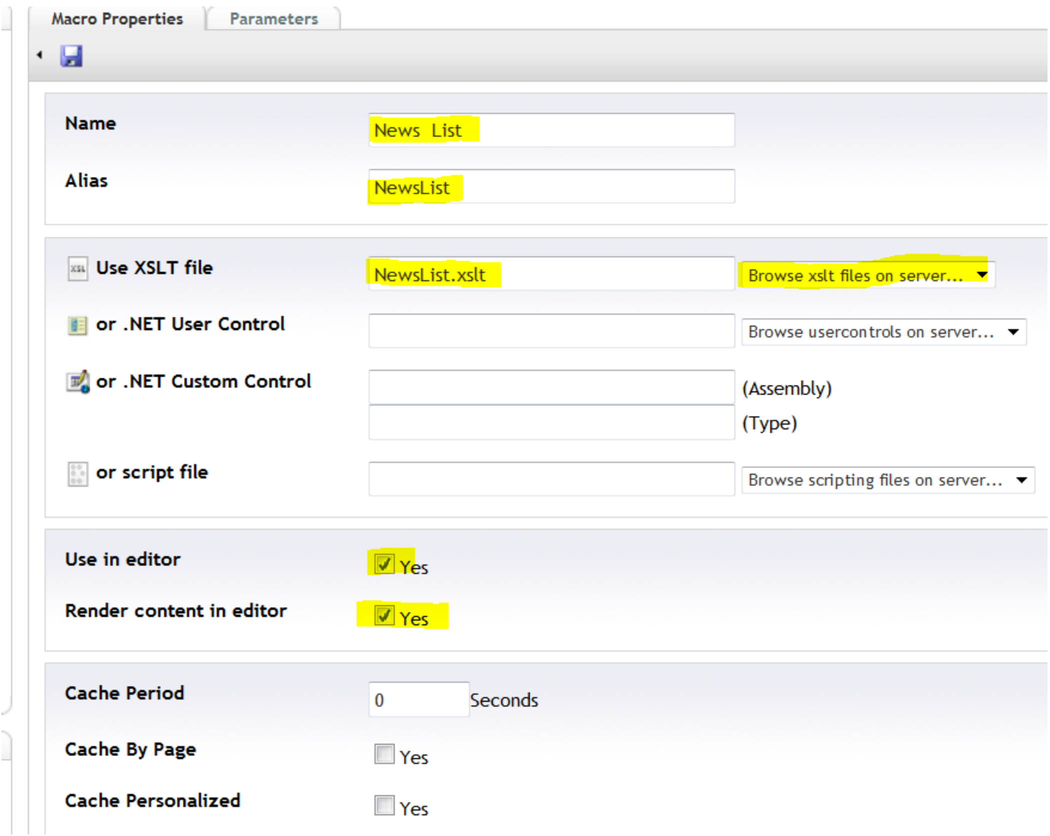
K- Cache for performance issue. Changing value will cause Umbraco to cache fetch data into server. It’s duration of cache time. 10 second means cache macro output for 10 second

L- Cached per page: cached page based on requested page

M- Cached per personalized: checked if you want to cache Macro output based on members or person who visiting web pages.

Let’s see another macro “News List”: it’s not difference but I checked in Use in Editor. *See captured screen, found in next page*. If you do not have macro created you can easily create by right click on macro and press “Create” provide following captured screen information and Press “Save”

[17] http://our.umbraco.org/wiki/reference/templates/umbracomacro-element/macro-parameters/advanced-macro-parameter-syntax



Macro Properties | Parameters

Name	News List
Alias	NewsList
Use XSLT file	NewsList.xslt — Browse xslt files on server...
or .NET User Control	Browse usercontrols on server...
or .NET Custom Control	(Assembly) (Type)
or script file	Browse scripting files on server...
Use in editor	☑ Yes
Render content in editor	☑ Yes
Cache Period	0 Seconds
Cache By Page	□ Yes
Cache Personalized	□ Yes

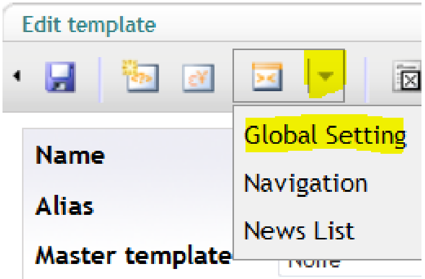
- *Have you notice I checked in "Use in Editor" and I unchecked in "Cache By Page".*
  - *I want to use news list macro in content editor so I checked "Use in Editor"*
  - *News list macro are responsible to list all resent news and news means should be delivered instantly without delay so that I do not want to cache news items. If I cache news item, recently news will be delay to display to end users.*

Let's get fun by inserting macro to templates and see how our web page looks likes, If you followed me. We need to insert Two Macro "Global Setting" and "Navigation" at "Web Site Global Settings" templates and One Macro "News List" should be inserted into Content title News.

Using Macro into templates:

Instruction:

- Open "Web Site Global Settings" Templates
- click on between opening and closing Div class="Title" tags.
- Click on Insert macro. Select "Global Setting" Umbraco will insert automatically. see Image right->

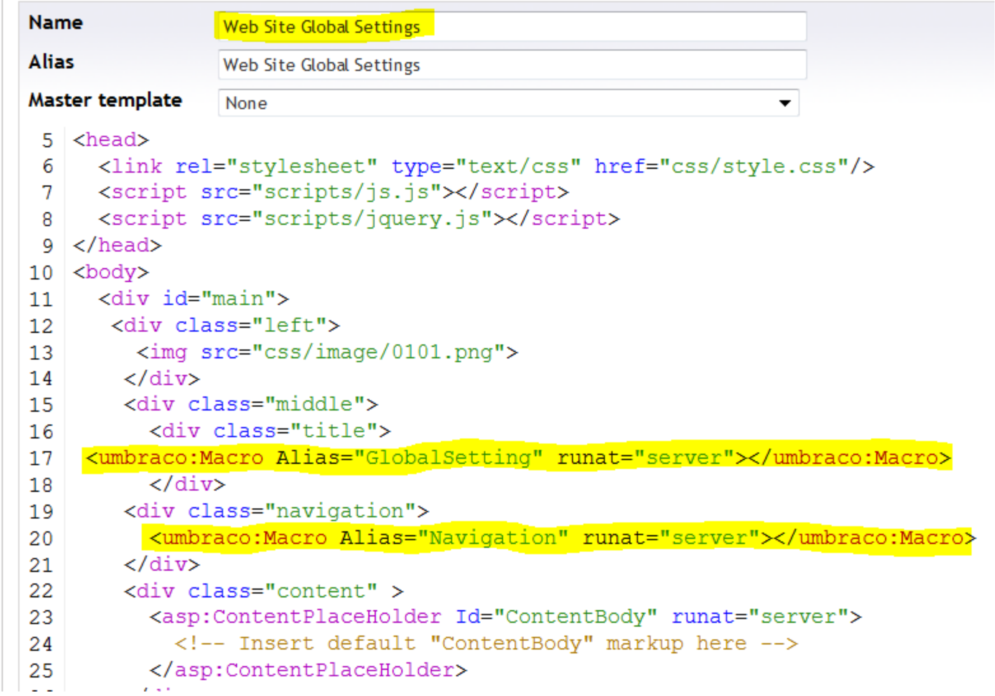


Edit template

Global Setting
Navigation
News List

Name
Alias
Master template None

- Repeat same process to insert navigation macro. Make sure you have to insert macro inside opening and closing div class="navigation"
- If you followed me "WebSite Global Settings" Template should look like this see highlighted area in captured screen bellow.



Name: Web Site Global Settings

Alias: Web Site Global Settings

Master template: None

```
<head>
  <link rel="stylesheet" type="text/css" href="css/style.css"/>
  <script src="scripts/js.js"></script>
  <script src="scripts/jquery.js"></script>
</head>
<body>
  <div id="main">
   <div class="left">
     <img src="css/image/0101.png">
    </div>
    <div class="middle">
      <div class="title">
 <umbraco:Macro Alias="GlobalSetting" runat="server"></umbraco:Macro>
      </div>
    <div class="navigation">
      <umbraco:Macro Alias="Navigation" runat="server"></umbraco:Macro>
    </div>
    <div class="content" >
      <asp:ContentPlaceHolder Id="ContentBody" runat="server">
        <!-- Insert default "ContentBody" markup here -->
      </asp:ContentPlaceHolder>
```

-

- *<umbraco:Macro>:* means, Macro starting tags
- *Alias:* alias of Macro we defined in Macro Section
- *Runat="server":* Means execute that block of code at server
- *</umbraco:Macro>:* closing of macros tags

Lets insert Macro into Content Editor as well and get knowledge of use of macro in different sections. Open content Section. I already create content "News" based on "General Content" Document Type. If you did not create, create first and continue to insert Macro in Content Editor. Using Macro in Content Editor is another way to fetch data in Umbraco CMS. When you create content make sure you checked "Use In Navigation" check box for Content "News" otherwise "News" Content would not show up in navigation bar.

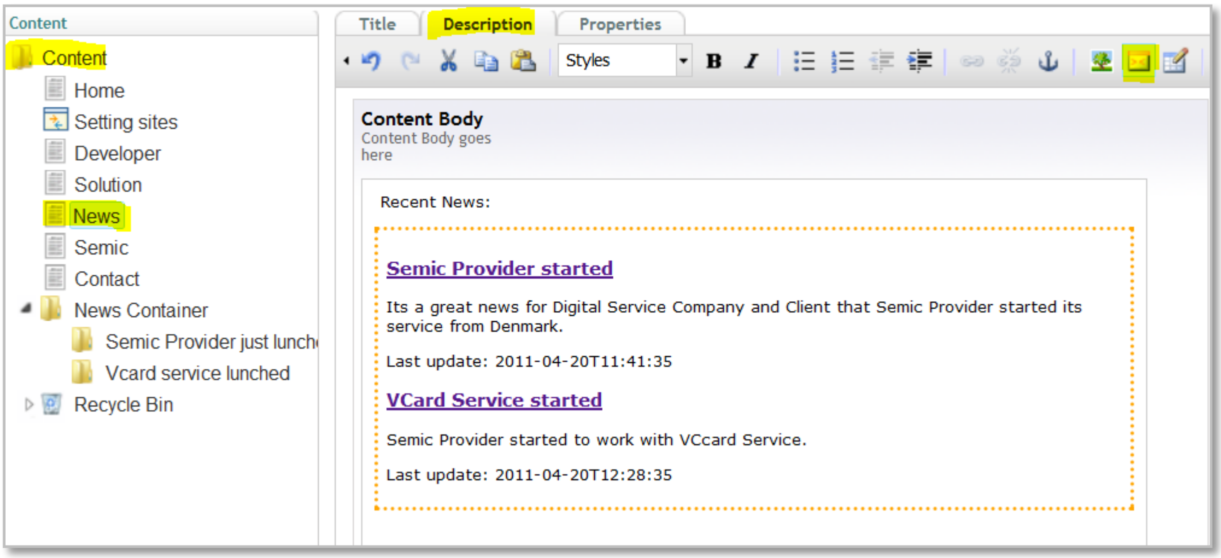
Instruction to Insert Macro in Content Area:

- Open content "News"
- Click Tabs "Description". It will show up Content body Area.
- Navigate Insert Macro at top-right corner of Richtext editor tools[18]
- Select Marco and Press Insert.

[18] Rich text editor tools. It means your Document Type property should have "Rich Text Editor" data type selected. Insert marco is not available for data type "simple text editor".

- Make sure to press “Save and Publish” Button.

- If you followed me. See highlighted area at captured screen bellow as reference.



Content Body
Content Body goes here

Recent News:

Semic Provider started

Its a great news for Digital Service Company and Client that Semic Provider started its service from Denmark.

Last update: 2011-04-20T11:41:35

VCard Service started

Semic Provider started to work with VCcard Service.

Last update: 2011-04-20T12:28:35

- All red dotted area denotes Macro areas.
- Did you see Two content are fetched at Content Editor? It is because we checked “Render in Content Editor” in News List Macro settings.
- If you see Empty “Insert Macro” menu means you did not checked “Use in Editor” check box in “News List” Macro settings. So go back to “News List” Macro and checked “use in Editor” After this macro should be available to insert at Content Editor.

Macro API (Application Programming Interface)

- `Macro macro = Macro.MakeNew(name);`
    - Passing name will made new macro. Alias will be assign automatically. You can change it later if you want to change
- `macro.UseInEditor = true;`
    - set value True if you want to use Macro into Content Editor. Else set False
- `macro.RenderContent = true;`
    - Set True if you want to render content into Content Editor. Else set Flase
- `macro.Xslt = "/xslt/NewsList.xslt";`
- provide location of xslt file to use.
- In above example macro is linking to “NewsList.xslt” xslt file.
- `macro.Save();`

When you press Save Button, Umbraco execute Save() void method.

Chapter 08:

Extending Umbraco CMS

Extending Umbraco CMS

Default Umbraco CMS may not contain enough methods or function that your website requires, in such case you need to extend Umbraco CMS website. Extending Umbraco CMS is not difficult if you already work with ASP.NET application. There are many ways to extend Umbraco CMS. Some are listed below.

User control
.NET Class Library
Razor Engine
XSLT
Python

Here I will discuss top two lists to extend Umbraco CMS. XSLT I already covered.

Razor Engine is based on MVC and not covered in this book.

Python is also not covered in this book.

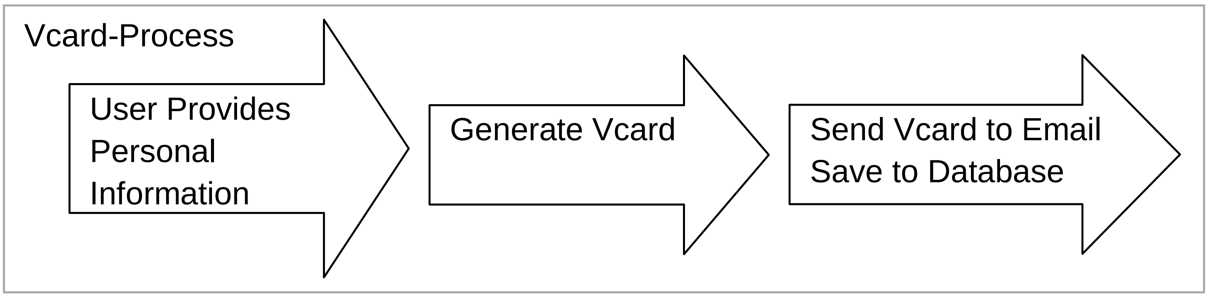
User Control:

User control is not difficult, if you already work with user control in ASP.NET websites, you will feel comfortable. The way you write the code is same as you write in ASP.NET websites. There is no difference writing code for Umbraco. Let's see image diagram bellow as references and make concept for User control.

In our Vcard Web sites, main production is to provide Online Vcard Generator So that user can create Vcard including image.

Assume that now you are not creating UserControl to Umbraco sits instead you are creating for ASP.NET websites. Make a plan same way that you would do in ASP.NET websites using Usercontrol. My approach to create Vcard generator in ASP.NET is follows

1- Add New User Control
2- Add Input textbox to type text, Add Upload to upload Image in View Page (".ascx")
3- Add Button To Upload and Create Vcard Methods
4- Writhe c# Code in Code Behind page (".ascx.cs")


Vcard-Process
User Provides Personal Information
Generate Vcard
Send Vcard to Email Save to Database

Before start coding:

You get short idea to create usercontrol for ASP.NET web site. It's a similar way for Umbraco also. We need to create Usercontrol first. Before start coding or programming I have to say something about how to start for VCARD.

Vcard is personal information that understands by machine. Personal information contains Name, Email, Phone, Image, Address, Title etc. To generate Vcard we should have personal information of user. User visit website so we need to provide some sort of input environment where use can type using keyboard or other assistive technology devices and retrieve data from client side. We can process to generate Vcard when we get enough information. Here what does it mean "enough Information"?

It can be different answers, in our Vcard concept. Vcard should be send to Email address means Email must have and should be valid otherwise we will not able to send Vcard through email. In our Vcard website Name is compulsory and other information are optional.

Never trust Client:

Clients are not controllable. Consciously or unconsciously clients may send invalid, harmful and dangerous data because we are providing input environment to end users. There is some security [19]an issue.

For example: XSS[20] (cross site script) and SQL Injection[21] are the major security issues[22]. To prevent from these attacks programmer needs to consider security issues like validation and remove special characters. Validation means validate input field like email must contain "identity", symbol "@", Domain Name and Domain Extension example ".com" or sub domain also acceptable. Special Characters[23] means those character are preserved by programming languages and used on program. Example ";" that used to end block of code.

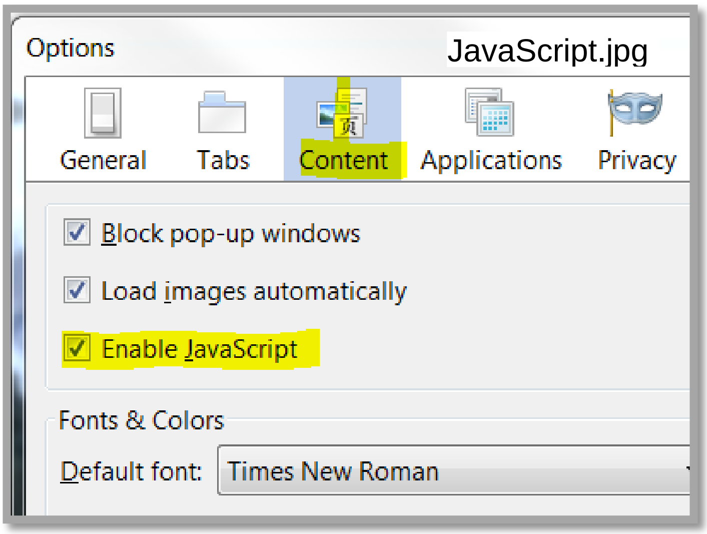
Client side validation

This is fast way to validate input data on client side using JavaScript or Jquery languages. Using client side validation, browser do not need to transfer data from Client to Server to Client. Browser will execute client side program and validate before submit data to server. Using client side validate we can provide rich validate environment. It's a fast way but we should not trust to client because client easily disable or escape these validation steps if s/he knows about program. For example if I program email validation using JavaScript. Let's open Firefox and Open "Option". You will see "Enable JavaScript" option in "Content" Section. If you uncheck it and Press OK, it will disable JavaScript. See Captured Screen in next page "JavaScript.jpg"

[19] Security Issues based on 2nd semester IT security class by Tue Becher.
[20] http://en.wikipedia.org/wiki/Cross-site_scripting
[21] http://en.wikipedia.org/wiki/Sql_injection
[22] Based on http://sanjevsharma.blogspot.com/2008/05/aspnet-top-security-issues-checklist.html
[23] Special characters like , ; "" '' {} / etc.

Did you notice any problem here when you disable JavaScript?

I explain more and it will be clear. Email is validating using JavaScript Code, but client can easily disable JavaScript on their browser when Client submit data all JavaScript Validation does not work because JavaScript is disable by Client in that case email validation is fail. So that to provide more secure we should also implement Server side validation.



JavaScript.jpg

☑ Block pop-up windows

☑ Load images automatically

☑ Enable JavaScript

Fonts & Colors

Default font: Times New Roman

Note: I will not cover “Client Side Validation” in this “How To Umbraco” basic.

Server Side Validation:

Server where client does not have control but you have control over server, make validation on server site will make site more secure that input data is in Valid format and not harmful. I cover shortly about server side validation here if you need more about Server Side Validation you can look up yourself.

Here I will cover:

- Name validation. For example it must have alphabet characters no numbers and Name must not be empty.
- Phone: it will validate Danish phone number format, its 8 digital characters without country code. You can easily implement phone number validation for other country code. Only difference is number of digital character. For example in Nepal mobile number has Ten digital format and Land line phones has eight digital characters in that case you need to provide two different mobile or phone validation rules.
- Email: I will use third party validation, someone already coded to validate email I will use it, of course I will explain what does it means.
- Replace Especial characters: I will show you how to replace special characters using Regex.
- “Regex” means Regular Expression.

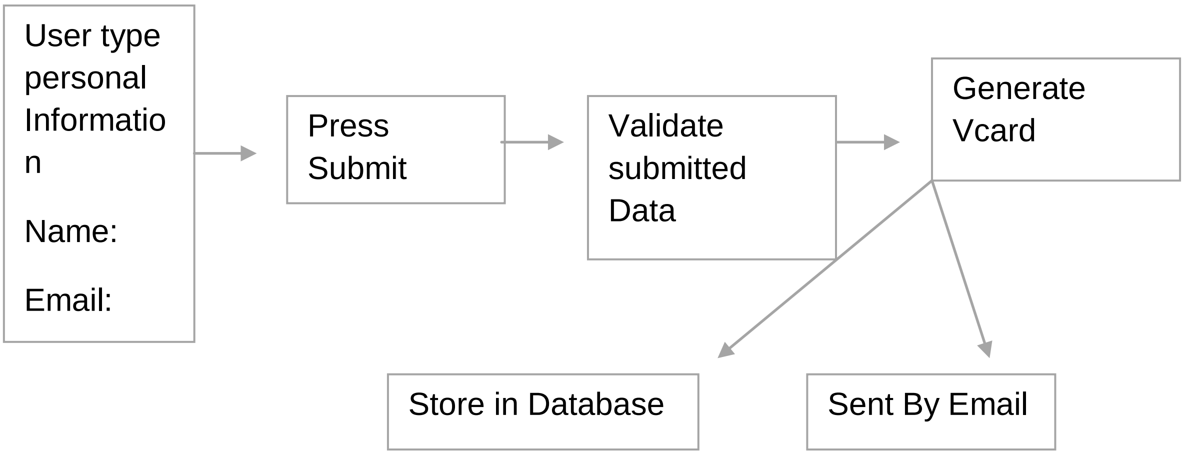
Submit Button:

After filling personal information user should able to submit form. For this purpose we need to create Button.

Storing Personal Data:
After user submitting data, we do not only generate Vcard but also Vcard is sent to Email. We also save to Database for later purpose[24]. For example if users want to edit some information later or User wants to generate Vcard later on, it's not good idea to provide same information more than once to create Vcard so we need to save data to Database, and retrieve whenever we need it.

Generate Vcard:
When user submits personal information then we need to process to generate Vcard.

Send Email:
As Vcard case, Vcard should be sent to Email.
Let's generate Visual Diagram how these will be done.



These are visual diagram before start coding let's reviews so it will be easier to code.

Needs to do in "User Control Design View"

- Create Seven Text Box for
    - Name, street, postal code, country, phone, email and website,
- Create Upload button for Image or Photo
- Create Button
    - To submit data

Needs to do in "Code behind Page"

- Create Class
    - To save data to Database
- Write Code to process following jobs.
    - Validate input methods
    - Go Generate Vcard
    - Send Email

Store Data to Database

[24] Later purpose means if you want to reuse email, name or location etc after user submitted form.

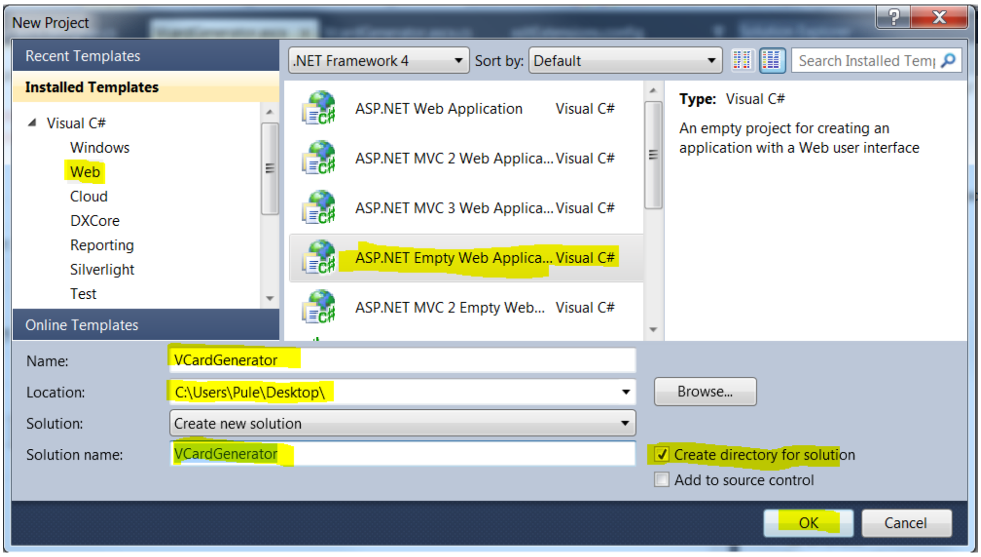
Let’s Begin to code:

Follow the steps To Create Project first

Open Visual Studio

- If You have not installed yet, please back to “Requirements” and install Visual Studio

- File -> New --> Project
- Select Web --> ASP.NET Empty Web Application… Visual C#
  - You can choose VB if you know how to code in VB but here I will use C#
- Name: Vcard Generator
- Location: choose different location if you want. I am using “Desktop”
- Solution name: leave default, if you want you can change
- Checked : Create Director for solution
  - It will Create new directory
- Press OK. See captured screen bellow



ASP.NET Web Application	Visual C#
ASP.NET MVC 2 Web Applica...	Visual C#
ASP.NET MVC 3 Web Applica...	Visual C#
ASP.NET Empty Web Applica...	Visual C#
ASP.NET MVC 2 Empty Web...	Visual C#

Type: Visual C#

An empty project for creating an application with a Web user interface

Name: VCardGenerator

Location: C:\Users\Pule\Desktop\

Solution: Create new solution

Solution name: VCardGenerator

☑ Create directory for solution

□ Add to source control

It should create new solution called “VCardGenerator”.

Now you need to add user Control

Right Click on Solution. → Add → New Items.

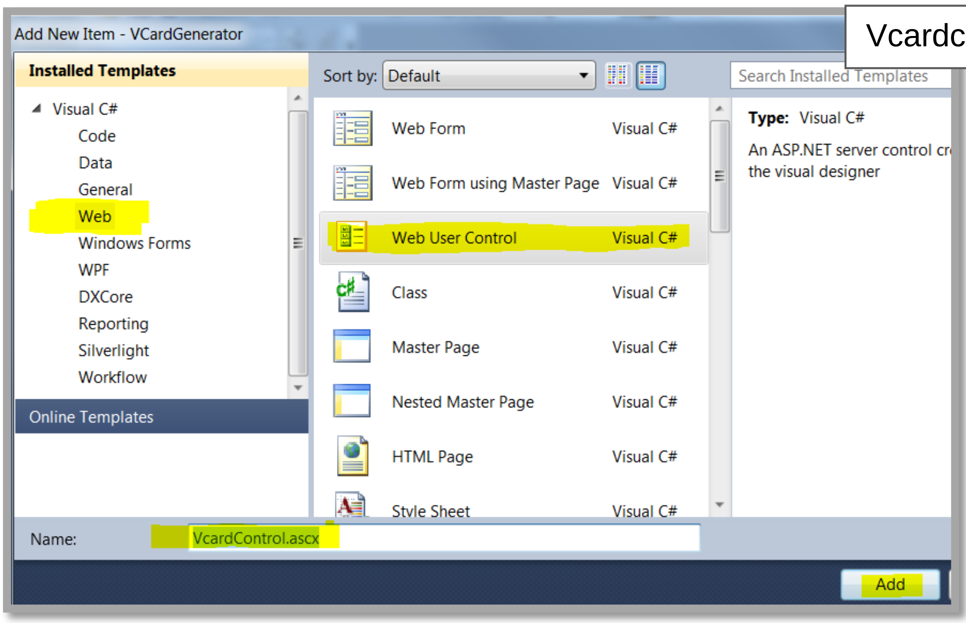
It will open new windows

Select → Web → Web User Control

Name “VcardControl.ascx” and Press Add.

You can change name if you want.

See captured screen “vcardcontrol.jpg”.



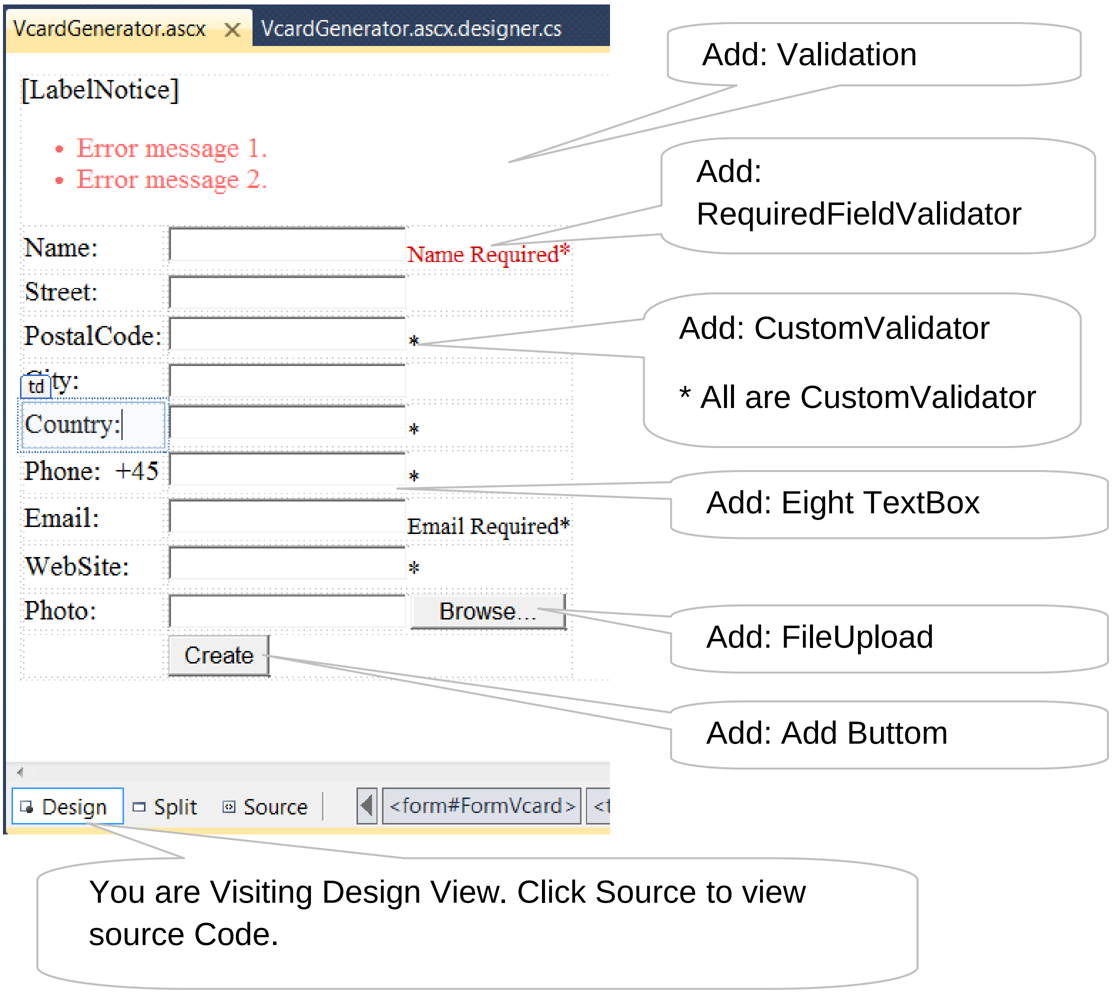
Vcardcontrol.jp

Add New Item - VCardGenerator

Name: VcardControl.ascx

-
-
-
- It must create three files end with “.ascx”, “ascx.cs” and “ascx.designer.cs”.
- Open VcardGenerator.ascx and Create Textbox, Button, Required Field validate and Custom Validate. See captured screen bellow

-



[LabelNotice]

- Error message 1.
- Error message 2.

Name:		Name Required*
Street:		
PostalCode:		*
City:		
Country:		*
Phone: +45		*
Email:		Email Required*
WebSite:		*
Photo:		Browse...
	Create	

Add: Validation

Add: RequiredFieldValidator

Add: CustomValidator

* All are CustomValidator

Add: Eight TextBox

Add: FileUpload

Add: Add Buttom

You are Visiting Design View. Click Source to view source Code.

I placed in table format; you can remove these table formats using div if you like.

- ValidationSummary: Validation Summary is used to provide users validation summary, if there is any validation fail, it will show up list of validation error message, error message are customizable.

```
<asp:ValidationSummary ID="ValidationSummary" runat="server" ForeColor="#FF6666" />
```

- Make sure you have runat="server", otherwise code does not execute at server.

- Add: TextBox

```
<asp:TextBox ID="TextBoxName" runat=server></asp:TextBox>
```

- This is example for textbox Name. Repeat same process for Street, PostalCode, City, Country, Phone, Email and WebSite.

- Each Textbox should have unique id so if you choose id that is self-explanatory it's easier to refer later on for example choose Textbox id "TextboxEmail" for email input textbox.

- Add: Required Filed Validator

In Vcard website Name an Email must provide so we need to add Required Field Validator. Go to Toolbox → Validation → click on “RequiredFieldValidator”

```
<asp:RequiredFieldValidator ID="RequiredFieldValidatorName" runat="server"
 ControlToValidate="TextBoxName" ErrorMessage="Name Required"> </asp:RequiredFieldValidator>
```

- RequiredFieldValidator start and end with asp:RequiredFieldValidator tags.
- Important thing in RequiredFiledValidator is “ContorlToValidate”, here ControlToValidate is equal to “TextBoxName”. Please look carefully “TextBoxName” is the id of Input “TextBox” that we created for Name Input Data that means we are now making Input Filed Name is required Field. ErrorMessage is the message that will appear if client tries to send empty data.
- Repeat same process “RequiredFieldValidator” for Email as well.
- Add: Custom Validator:
- input field should be validate on server side. To add custom validator,
- Go to Toolbox → Validation → click on “CustomValidation”

```
<asp:CustomValidator ID="CustomValidatorName" runat="server"
ControlToValidate="TextBoxName" ErrorMessage="Name is not Valid"
onservervalidate="CustomValidatorName_ServerValidate">*</asp:CustomValidator>
```

- Custom Validator start and end with asp:CustomValidator tags.
- controlToValidate=”TextBoxName” is we are validating Input TextBox Name.
- ErrorMessage. Message show up in Validation Summary list if input data is not valid.
- * text or character between asp:CustomValidator start and end tags will show up where you placed asp:CustomValidator tags if Data is not Valid.
- Onservervalidate here we specify which methods should we run or is relevant. it is linking methods called “CustomValidatorName_ServerValidate” this method must be create at code page. Explained later one.
- Repeat similar process for Email, street, postal code, country, city, phone, email and website.
- For references see code at next page.

VcardControl.ascx

```
<form id="FormVcard" runat="server">
<div class="errorMessage">       <asp:Label ID="LabelNotice" runat="server" Text=""></asp:Label>
<asp:ValidationSummary ID="ValidationSummary" runat="server" ForeColor="#FF6666" /></div>
<table >
  <tr class="name">
    <td> Name: </td><td><asp:TextBox ID="TextBoxName" runat=server></asp:TextBox>
    <asp:RequiredFieldValidator ID="RequiredFieldValidatorName" runat="server"
      ControlToValidate="TextBoxName" ErrorMessage="Name Required" Font-Size="Small"
ForeColor="#CC0000"></asp:RequiredFieldValidator>
    <asp:CustomValidator ID="CustomValidatorName" runat="server"
      ControlToValidate="TextBoxName" ErrorMessage="Name is not Valid"
      ForeColor="#CC0000"
onservervalidate="CustomValidatorName_ServerValidate">*</asp:CustomValidator>
    </td>
  </tr>
  <tr class="street">
    <td> Street:</td><td> <asp:TextBox ID="TextBoxStreet" runat="server"></asp:TextBox>
    </td>
  </tr>
  <tr class="postalCode">
    <td> PostalCode:</td><td> <asp:TextBox ID="TextBoxPostalCode"
runat="server"></asp:TextBox>
    <asp:CustomValidator ID="CustomValidatorPostalCode" runat="server" Font-Size="Small"
ErrorMessage="Postal Code 4 digit" ControlToValidate="TextBoxPostalCode"
onservervalidate="CustomValidatorPostalCode_ServerValidate">*</asp:CustomValidator>
    </td>
  </tr>
  <tr class="city">
    <td> City:</td><td> <asp:TextBox ID="TextBoxCity" runat="server"></asp:TextBox>
  </tr>
  <tr class="country">
    <td> Country:</td><td> <asp:TextBox ID="TextBoxCountry" runat="server"></asp:TextBox>
    <asp:CustomValidator ID="CustomValidatorCountry" runat="server" Font-Size="Small"
      ErrorMessage="Country Not Valid" ControlToValidate="TextBoxCountry"
onservervalidate="CustomValidatorCountry_ServerValidate">*</asp:CustomValidator>
    </td>
  </tr>
  <tr class="phone">
    <td> Phone:&nbsp; +45</td><td><asp:TextBox ID="TextBoxPhone"
runat="server"></asp:TextBox>
    <asp:CustomValidator ID="CustomValidatorPhone" runat="server" Font-Size="Small"
      ErrorMessage="Phone must be 8 digit, Without +45 "
      ControlToValidate="TextBoxPhone"
onservervalidate="CustomValidatorPhone_ServerValidate">*</asp:CustomValidator>
    </td>
  </tr>
  <tr class="email">
    <td> Email: </td><td><asp:TextBox ID="TextBoxEmail" runat="server"
      ontextchanged="TextBoxEmail_TextChanged"></asp:TextBox>
    <asp:RequiredFieldValidator ID="RequiredFieldValidatorEmail" runat="server" Font-Size="Small"
      ErrorMessage="Email Required" ControlToValidate="TextBoxEmail">Email
Required</asp:RequiredFieldValidator>
    <asp:CustomValidator ID="CustomValidatorEmail" runat="server" Font-Size="Small"
      ErrorMessage="Email Not Valid" ControlToValidate="TextBoxEmail"
      onservervalidate="CustomValidatorEmail_ServerValidate">*</asp:CustomValidator>
    </td>
  </tr>
  <tr class="website">
```

```
        <td> WebSite:</td><td> <asp:TextBox ID="TextBoxWebsite" runat="server"></asp:TextBox>
        <asp:CustomValidator ID="CustomValidatorWebSite" runat="server"
            ErrorMessage="Website not valid" ControlToValidate="TextBoxWebsite"
            onservervalidate="CustomValidatorWebSite_ServerValidate">*</asp:CustomValidator>
        </td>
     </tr>
    <tr class="photo">
        <td> Photo:</td><td> <asp:FileUpload ID="FileUploadPhoto" runat="server" />
        </td>
    </tr>
    <tr >
        <td>
            &nbsp;</td> <td> <asp:Button ID="ButtonCreate" runat="server" Text="Create"
            onclick="ButtonCreate_Click" /> </td>
    </tr>

</table>
</form>
```

If you want to follow me, copy above code and paste it into your User Control Design View.

General description of “VcardControl.ascx” if you are new to ASP.NET user control

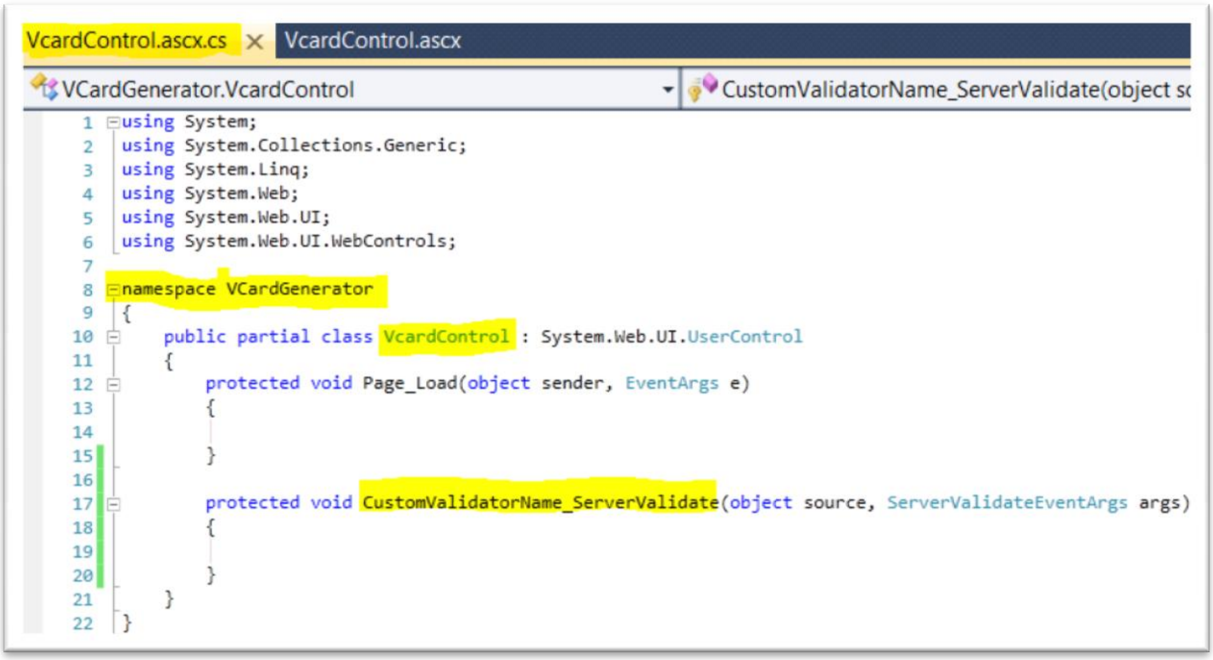
Description: It start with form tags, you must include runat=”server” because we want to execute code at server side. And I added div class that used for style sheet and I used table to nicely format input fields.

<asp:Label ID="LabelNotice" runat="server" Text=""> it’s a label that will use to show up conformation message to end users or client.

- <table> is starting table
- <tr> it’s a table row. One row contains multiple data cells so that we can use multiple <td> between opening and closing tags of table row <tr>. TD stands for “Table Data”.
- &nbsp; means white space.
- Closing tags of table row, table and form tags.
- All tags contain forward slash “/” means closing tags.
- Example </tr> </td>, </table> tags.
- Onclick is events that will executed when user clicks on Button. In our Vcard we crate button id “ButtonCreate” and called “ButtonCreat_Click” methods that is in code behind page.

Code behind Page "VcardControl.ascx.cs" page.

To open code behind page you can double click on button. For example let's double click at CustomValidator for Name. It will create "CustomValidatorName_ServerValidate" methods at code behind page and links automatically at design view for you. Or you can manually add methods to code behind page and link manually at Design view. See captured screen bellow of code behind page.

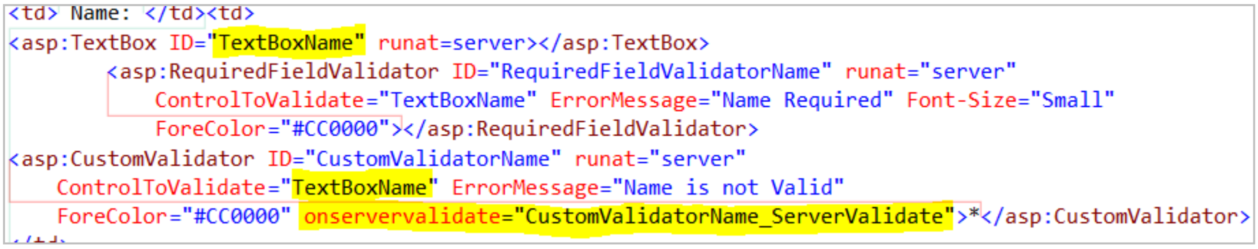

VcardControl.ascx.cs
VcardControl.ascx
VCardGenerator.VcardControl
CustomValidatorName_ServerValidate(object sc

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace VCardGenerator
{
    public partial class VcardControl : System.Web.UI.UserControl
    {
        protected void Page_Load(object sender, EventArgs e)
        {

        }

        protected void CustomValidatorName_ServerValidate(object source, ServerValidateEventArgs args)
        {

        }
    }
}
```

- Using is used to references other namespace
- *Namespace* "VCardGenerator" namespace name.
- *Public* means public, available from outside
- *Colon* ":" means left side of class is inherited from right side of class
- *Partial Class*: it's not a single class; C# compiler will group all the partial classes together and treat as a single class.
- *VcardControl* is name of class and it is inheritance from System.Web.Ui.Usercontrol
- *Protected* means, member is accessible within its class and by derived classes.
- *Void*: it means do not return anything, just execute block of code that inside open curly and close curly brackets.
- *Page Load*: it's a method that executed when page load at server, All code inside Page_Load will executed automatically when page loads.
- Be care full you should not write your code outside open and close curly brackets of Class. You have to place your code inside open and close curly brackets of Class. In our case "VcardControl class", otherwise C# compiler will failed.

CustomValidator for Name:

Let's Create CustomValidator for Name, Open Design View "VcardControl.ascx" and double click on CustomValidator that link to Name Textbox. If you follow me your code behind page should looks like this



```
<td> Name: </td><td>
<asp:TextBox ID="TextBoxName" runat=server></asp:TextBox>
        <asp:RequiredFieldValidator ID="RequiredFieldValidatorName" runat="server"
            ControlToValidate="TextBoxName" ErrorMessage="Name Required" Font-Size="Small"
            ForeColor="#CC0000"></asp:RequiredFieldValidator>
<asp:CustomValidator ID="CustomValidatorName" runat="server"
    ControlToValidate="TextBoxName" ErrorMessage="Name is not Valid"
    ForeColor="#CC0000" onservervalidate="CustomValidatorName_ServerValidate">*</asp:CustomValidator>
```

- Above captured screen is part of "VcardContro.ascx"
- See second line of code, its TextBox and has id "TextBoxName"
- See seventh line of code, ControlToValidate ="TextBoxName" . it means CustomValidator is linked to TextboxName.
- See last line of onservervalidate="CustomValidatorName_ServerValidate"

onservervalidate is important, in above case it's equal to "CustomValidator_ServerValidate". It must be same name methods that you specify between double quote in your code behind page. Let's create methods "CustomValidatorName_ServerValidate".

```
protected void CustomValidatorName_ServerValidate(object source, ServerValidateEventArgs args)
    {
        args.IsValid = RemoveSpecialCharacter(args.Value).Length > 0;
    }
```

- This is simple validation, here I get serverValidateEventArgs and stored in variable args.
- I called bool methods IsValid, if provided value's length is greater than zero it will set validate value equal to true otherwise it will set up false.
- Here I pass values to RemoveSpeicalCharacter() methods because I want to replace all special character to secure websites.

RemoveSpeicalCharacter() methods

```
public string RemoveSpecialCharacter(string input)
        {
            return Regex.Replace(input, "[^0-9a-zA-Z. ]+", "");
        }
```

- It is written in Regular expression.
- Regex is regular expression can be use using Regex.
- Make sure to add `using System.Text.RegularExpression before Namespace starts.`

- RemoveSpeicalCharacter() Method take one parameter string, it removes all character but not from 0 to 9 digits, from A to Z upper and lower case characters and period.

Email Validation:

- Email Validation:
- I created methods ValidateEmail()[25]

ValidateEmail () methods:

```
public bool ValidateEmail(string email)
   { return Regex.IsMatch(email,
              @"^(?("")("".+?""@)|(([0-9a-zA-Z]((\.(?!\.))|[-
!#\$%&'\*\+/=\?\^`\{\}\|~\w])*)(?<=[0-9a-zA-Z])@))" +
               @"(?(\[)(\[(\d{1,3}\.){3}\d{1,3}\])|(([0-9a-zA-Z][-\w]*[0-9a-zA-
Z]\.)+[a-zA-Z]{2,6}))$"); }
```

- This method has one parameter email; here input string email is compared with pattern. Pattern is “valid email character id” + “@” + “Domain or sub Domain name”. And finally returns true if pattern is match otherwise return false.
- Example:
  - *Jeevan_2np@yahoo.com* will return true
  - *Jeevan_2np.at.the.rate.yahoo.com* will return false- because “@” is missing here. Email id must contain “@” symbol

- Here is CustomEmailValidator Methods

```
protected void CustomValidatorEmail_ServerValidate(object source,
ServerValidateEventArgs args)
  {        args.IsValid = ValidateEmail(args.Value.ToString());    }
```

Description: args.Value.ToString() is holding value that user typed in Email input Textbox in Browser, passing that value to ValidateEmail() method will decide whether passing value is in valid email format or not. If it is valid it will set args.IsValid = true, if it is not valid it will set args.IsValid = false;

Phone Validation:

CustomValidatorPhone_ServerValidate:

```
protected void CustomValidatorPhone_ServerValidate(object source,
ServerValidateEventArgs args)
        {
            if (args.Value.Length > 0)
            {
                args.IsValid = ValidatePhone(args.Value);
            }
            else
            {
                args.IsValid = true;
            }
        }
```

[25] Methods originally created by http://msdn.microsoft.com/en-us/library/01escwtf.aspx

Description: “CustomValidatorPhone_ServerValidate”

First we check if user typed some value in Phone Number text box or not. If user typed some character on input text box its length must be greater than zero. If input value is great than zero, we need to validate whether it is valid phone number or not. We pass value to ValidatePhone() methods, ValidatePhone() method return true or false. If it is valid args.IsValid value is set to true, if not value is set to false.

Else statement execute if input value is empty or less than zero character length. It means user do not want to provide phone number on their Vcard so that args.IsValid is set to true.

```
ValidatePhone() methods:
public bool ValidatePhone(string input)
        {
            if (input.Length != 8) { return false; }
            Regex num = new Regex("[^0-9]");
            return !num.IsMatch(input);
        }
Description:
It is not new, only logic inside method block of code is new. It will
compare input value to pattern. First it will check if input value is not
equal to 8 characters length, it’s not valid mobile or phone number in
Denmark so return false. Here I will only include validating Danish phone
number. If you need for other countries look yourself. Process is similar
only number pattern is difference. If input value is equal to 8 characters
but characters contain alphabets, it’s not valid number because phone number
does not contain alphanumeric character. It must be numeric character. So
Regex will compare input value with pattern. If input value is not from 0 to
9 digits, it will return false because it’s not valid phone number.
```

Validation for rest of Input Fields Street, PostalCode, City, Country, and Website are listed at appendix as complete code or putting all code together “Vcardcontrol.ascx.cs” page title. Please check appendix page for more details.

Create Button Events.

End user submits data by clicking Create Button; here we have to do all processing steps. I have listed steps that need to do after end user clicks on “Create Button”.

- Step 1: Hold or Store submitted data in a variable
- Step 2: Validate and Remove special characters from submitted data
- Step 3: Upload Image to server and save it in server location if user choose image files.
- Step 4: Generate Vcard on the fly
- Step 5: Send attach Email with Vcard
- Step 6: Store data to Database for later use.

Step 2 is already done for Email, Name and Phone. Rest of validation and remove special character are available at appendix page as complete code so that please check out appendix page.

Step 1: Hold or Store Submitted Data

-it's a very easy. See example for holding or retrieving value for Name from client side. string name = RemoveSpecialCharacter(TextBoxName.Text); I have created string variable name, name is equal to TextBoxName.Text, here TextBoxName is id of TextBoxName and .Text means text value from input textbox. I passed text value from input textbox to method "RemoveSpecialCharacter()" to remove special character. At the end string name will be equal to data or value that user typed in textbox other then special characters.

Repeat same process for rest of input data. See example as references.

```
string streetaddress = RemoveSpecialCharacter(TextBoxStreet.Text);
string country = RemoveSpecialCharacter(TextBoxCountry.Text);
string zip = RemoveSpecialCharacter(TextBoxPostalCode.Text);
string city = RemoveSpecialCharacter(TextBoxCity.Text);
string email = TextBoxEmail.Text;
string web = RemoveSpecialCharacter(TextBoxWebsite.Text);
string phone = RemoveSpecialCharacter(TextBoxPhone.Text);

Note: email is without RemoveSpecialCharater methods, it is because if user
submits especial characters in email, Email validation will set false, in that
case further process will not continue to generate Vcard.
```

Step 2: Validate Data

Double click on "Create" button, visual studio will automatically add event and open code behind page. If you follow me it must be looks like bellow

```
protected void ButtonCreate_Click(object sender, EventArgs e)
    {
        if (Page.IsValid)
        {          // processing code goes here……..          }
         }
```

Visual studio creates ButtonCreate_Click() method events automatically. I added if(Page.IsValid) statement. It is very important to check all validation must be true, Page.IsValid will check whether page is valid or not.

```
For example
protected void CustomValidatorEmail_ServerValidate(object source,
ServerValidateEventArgs args)
  {        args.IsValid = ValidateEmail(args.Value.ToString());    }

In highlighted text above, if valid email address is provided by user then args.IsValid
= true; in order to be Page.IsValid = true, args.IsValid must be equal to true in
highlighted text above. This is an example for email. In our Vcard website there is
more than email validation so that all validation should be true in order to execute if
(Page.IsValid) { } block of code. Remember that Validation for Name, Street,
PostalCode, City, Country, Phone, Email, Website and photo should have be true.
```

If there is any validation fails or has false value, error message will show up at error summary and will not execute if(Page.IsValid) {} block of code.

- Note: Double back slash inside code is used to comments in c# programming language.

Step 3- Image Upload and Save to Server:

In Vcard, user is allowed to have Vcard with image, in order to generate vcard with image user should able to upload image using browse button. Add Upload Button, make Id is naming convention so upload button will be easy to recognize on code behind page. if you added Button it must look like bellow.

```
VcardControl.ascx
       <asp:FileUpload ID="FileUploadPhoto" runat="server" />
       Description: FileUpload allows to upload file
       ID= "FileUploadPhoto" it's an id of file upload used in code behind page.
       Runat = "server" will execute that code at server.
```

In code behind page: using posted.saveas() methods, image will be uploaded to server. See code bellow as references. Your logic can be difference.

```
if (FileUploadPhoto.PostedFile != null && FileUploadPhoto.PostedFile.ContentLength
> 0)
{     string guid = id.ToString("N");
      string imageDirectory = Server.MapPath("vcardimage");
      if (Directory.Exists(imageDirectory))
      {
       FileUploadPhoto.PostedFile.SaveAs(
       Server.MapPath("/vcardimage/" + guid + FileUploadPhoto.FileName));
        }
        else
           {       Directory.CreateDirectory(Server.MapPath("vcardimage"));

FileUploadPhoto.PostedFile.SaveAs(Server.MapPath("/vcardimage/" + guid +
FileUploadPhoto.FileName));
  }
 }

Description:
   -   Fist we ensure that user browse or select file and Content Length of file
       is greater than zero that can be achieve  by using
       FileUploadPhoto.PostedFile != null (means posted files should not equal to
       null or empy) and FileUploadPhoto.PostedFile.ContentLength > 0 means length
       of file should be greater than zero. if file name and content length is
       greater than zero we go to next step.

   -   Second: we make new guid, its unique id that system generates
       automatically, in theory guid is never be same or do not match each other.
       You may have raise questions here why do we need guid? We need because when
       user upload files we do not know what file name user has, if two user
       upload different file but same image file name and file type, system will
       overwride old files. To prevent from this we add extra string to file name
       so that file name will not be same even user upload more than one files on
       same name and file format.
           o   To get guid id we should call Guid id = Guid.NewGuid(); and string
               guid = id.ToString("N"); "N" means guid will formatted without "-"
               symbol. By default guid looks like in that format 0000-0000-0000.
               Using "N" will remove "-"so that guid will be in format of 123456.
```

- Third: we check whether directory “vcardimage” exist or not. It is very important to make sure there must be directory to save files in Vcard case, all images goes insides “vcardimage” directory. You can choose different name if you like. If directory exist we save file
    - Directory.Exists(directory name) will check whether passed director exist or not.
    - FileUploadPhoto.PostedFile.SaveAs(Server.MapPath("/vcardimage/" + guid + FileUploadPhoto.FileName));
    - PostedFile.SaveAs() methods used to save posted file. Posted files means user submitted page to server by clicking buttons.
    - FileUploadPhoto is id of asp:FileUpload that we created at design view or at VcardControl.ascx page
- Forth: if director does not exist “else” statement code will be executed so we create directory first and save image after that.
    - Directory.CreateDirectory(directory name) methods is used to make directory.

Step 4: Generate Vcard:

Now we have required information, validation and image. It’s time to generate Vcard. Writing Vcard is no difference from writing file and adding text line to file. See bellow code as reference. Only difference is structure and data. If you want to know more about Vcard structure you can visit microformat official web sites at microformat.org. Also see references[26]. Your idea may be difference than mine idea.

Methods Name: GenerateVcard() Parameters: string firstName, string streetAddress, string zip, string city, string country, string email, string phone, string web and string image

```
Code:        var vcardStringBuilder = new StringBuilder();
           vcardStringBuilder.AppendLine("BEGIN:VCARD");
           vcardStringBuilder.AppendLine("VERSION:2.1");
           vcardStringBuilder.AppendLine("FN;LANGUAGE=en-us:" + firstName);
           vcardStringBuilder.Append("ADR;HOME;PREF:;;");
           vcardStringBuilder.Append(streetAddress + ";");
           vcardStringBuilder.Append(zip + ";");
           vcardStringBuilder.Append(city + ";");
           vcardStringBuilder.AppendLine(country + " ");
           vcardStringBuilder.AppendLine("TEL;HOME;VOICE:" + phone);
           vcardStringBuilder.AppendLine("URL;type=pref:" + url);
           vcardStringBuilder.AppendLine("EMAIL:" + email);
             vcardStringBuilder.AppendLine("PHOTO;ENCODING=BASE64;TYPE=" +
             image.GetType().ToString().ToUpper() + ":");
             vcardStringBuilder.AppendLine(Convert.ToBase64String(ReadImage(image))
             );
                        vcardStringBuilder.AppendLine(string.Empty);
        vcardStringBuilder.AppendLine("END:VCARD");
             return vcardStringBuilder.ToString();
```

[26] http://en.wikipedia.org/wiki/VCard for vcard example

Methods Name: GenerateVcard()
Description:

- Start writing with line "BEGIN:VCARD" using StringBuilder.AppendLine() methods.
- I append rest of information after "BEGIN:VCARD" before "END:VCARD". Its Vcard structure we must follow otherwise Vcard will not work as expected.
- Other structure I will not explain here it's not hard to understand if you know Vcard structure. For more details about structure see reference link.
- Adding image to Vcard may be difficult you to understand so I will explain here. In Vcard image should be added as characters so that first we need to read all images as byte array see ReadImage(image path) methods bellow.

    -
      ```
      public byte[] ReadImage(string image)
      {            return File.ReadAllBytes(image); }
      ```
    - Here I pass image file location, using File.ReatAllBytes() methods, it will read image as bytes and return it than I need to convert to string using Convert.ToBase64String() methods.
- Make sure to specify "PHOTO;ENCODING=BASE64;TYPE= file type" before appending actual image string.
    - Here PHOTO: used to indicate Vcard contains image
    - Encoding: types of encode, in our case Base64
    - Type: type of image or format. Example .jpg, .png .gif etc.
- Finally Vcard is return in format of String for further process.

Step 5: Send attach Email with Vcard

After generating Vcard, Vcard needs to be delivered by Email attachment. Take a look bellow code how to send email using SMTP. I use gmx[27] as an example.

```
private void SendVcard(string subject, string body, string email, string file)
      {       byte[] byteArray = Encoding.ASCII.GetBytes(file);
              MemoryStream stream = new MemoryStream(byteArray);
              MailMessage message = new MailMessage();
              message.Subject = subject;
              message.Body = body;
              message.From = new MailAddress("sender id ");
              message.To.Add(email);
              SmtpClient client = new SmtpClient();
              Attachment vcard = new Attachment(stream, "PersonalVCard.vcf");
              message.Attachments.Add(vcard);
              client.Host = host name example "mail.gmx.com";
              client.Port = SMTP port number. Example 587 for gmx smtp service.
                                   client.Credentials = new
                   NetworkCredential("user id ", " password");
              client.EnableSsl = true;
              client.Send(message);
     }
```

Sender id= replace with your sender id

- User id = replace with real user id that has access to use SMTP service
- Password = replace with real valid password to access SMTP service
- Host = replace with real host name. For more details contact to your web hosting provider company.

[27] www.gmx.com is free smtp provider as gmail.com, yahoo.com and hotmail.com need upgrade to use their smtp service using API.

Description:

```
  Description: input string subject, string body, string email and string file are
parameters, required to send email.
   -  byte[] byteArray = Encoding.ASCII.GetBytes(file);
         o  Read all bytes that are Vcard, We created Vcard in step 4.
         o  MemoryStream is used to write bytes into memory
   -  MailMessage message = new MailMessage();
         o  instantiate MailMessage
         o  added receiver address, subject, sender
   -  using Attachment we can attach files into email
         o  Attachment vcard = new Attachment(stream, "PersonalVCard.vcf");
   -  Instantiate SmtpClient service, add Host, Port, Credentials
      (authentication), and finally call Send() method to send email.
   -  EnalbeSsl= true means use secure socket layer.
```

Here Vcard is sent to Email address, it make sure that email address should be valid and right person, but it also cause user may need to check spam email.

What if server or SMTP fails to send email?

Lets create option for incase SMTP fails to send email. See code bellow

```
  /// <summary>
        /// Downloads the vcard.
        /// </summary>
        /// <param name="vcardFile">The vcard file.</param>
        public void DownloadVcard(byte [] vcardFile)
        {
            Response.ClearHeaders();
            Response.Clear();
            Response.Buffer = true;
            Response.ContentType = "application/vcard";
           Response.AddHeader("content-disposition", "attach;filename =
      YourVcard.vcf");
            Response.Charset = "";
            Response.BinaryWrite(vcardFile);
            HttpContext.Current.ApplicationInstance.CompleteRequest();
            Response.End();
        }

Description: I will describe very shortly.
   -  created methods DownloadVcard() and took one parameters it's a Vcard File
   -  start clearing headers if there is any by calling ClearHeaders() methods
   -  clear any content if there is any by calling Clear() methods.
   -  Set buffer true
   -  Content type is Vcard
   -  Add headers and File name
   -  Start writing Vcard by using BinaryWriter() methods
   -  Finall complete writing and httpcontent by using CompleteRequest() and End()
      methods.
Call DownloadVcard() methods at Catch Block of code in SendVcard() methods.
See complete code at appendix page.
```

Step 6: Store data to Database for later use

In this step we will store Vcard data into Database for later use. To store data, first we need to ensure that table and fields are exits into database. Here is over view of logic how I am going to do this job done.

1) Create table and fields into Database
2) Check table before insert data.
3) Insert data into table and fields using Umbraco system

Create Table and Fields into Database[28]

There are many way to create table and fields into database. You may create using Sql Server Management Studio or you can create using C# code. Here I will explain sustainable way to program.

Creating Table and Fields:

```
string queryString =     @" CREATE TABLE [vcardimage](
                        [id] [int] IDENTITY(1,1) NOT NULL,
                         [name] [nvarchar] (250) NULL,
                        [street] [nvarchar](250) NULL,
                        [postalcode] [int] NULL,
                        [country] [nvarchar](250) NULL,
                        [phone] [nvarchar](250) NULL,
                        [email] [nvarchar](250) NULL,
                        [website] [nvarchar](250) NULL,
                        [date] [datetime] NULL,
                        [imageguid] [nvarchar](250) NULL)";
```

I have created queryString first. CREATE TABLE [table name] is used to create table. Fields are added inside open and close brackets.

- [id] [int] IDENTITY(1,1) NOT NULL means: [id] is field name, [int] short form of integer is its property, (1,1) means increase automatically by +1, NOT NULL means field id should not be null it must contains data.
- [name] [nvarchar] (250) NULL, [name]: field name is name, its property is nvarchar and (250) means its maximum capacity, Null means name field may be null or empty
- Street, country, phone, email, website and imageguid are similar as Name field
- postalCode and DateTime has different properties int (integer) and Datetime respectively.
- For more details check Sql query languages.

[28] Inspired by http://umbraco.com/follow-us/blog-archive/2009/12/7/blog-4-umbraco-205-%E2%80%93-comments

Note: Given creating table and fields example are only for MsSql database, if you using MySql database code may be difference.

Executing queryString:

Follow the steps:

- Create sql command object
  - using `SqlCommand command = new SqlCommand(queryString, connection)`
  - pass queryString, we just created and pass connection string. Connection string is very important because it contains database location and authentication. To create table user should have admin access to Database.
- To reuse Umbraco connection string use folling code
  - `SqlConnection connection = new SqlConnection(GlobalSettings.DbDSN);`
  - Make sure to add reference using umbraco before use GlobalSettings.DbDSN.
- Open database connection using connection.Open();
- Use ExecuteNonQuery() Methods, command.ExecuteNonQuery();
- Why do we use ExecuteNonQuery() method, its because we just need to execute code, no need to retrieve any data back after executing code. It's a fast process and increase performance.
- connection.Close();
- Finally close connection by calling Close() methods.

Code all in one for CreateTable methods

```
private static bool CreateTable(string tableName)
        {
            try
            {
                SqlConnection connection = new
SqlConnection(GlobalSettings.DbDSN);
                string queryString =
                    @" CREATE TABLE [vcardimage](
                        [id] [int] IDENTITY(1,1) NOT NULL,
                         [name] [nvarchar] (250) NULL,
                        [street] [nvarchar](250) NULL,
                        [postalcode] [int] NULL,
                        [country] [nvarchar](250) NULL,
                        [phone] [nvarchar](250) NULL,
                        [email] [nvarchar](250) NULL,
                        [website] [nvarchar](250) NULL,
                        [date] [datetime] NULL,
                        [imageguid] [nvarchar](250) NULL)";

         SqlCommand command = new SqlCommand(queryString,
connection);
                connection.Open();
                command.ExecuteNonQuery();
                connection.Close();
                return true;
            }
            catch
            {
                return false;
            }
        }
```

Never forget to put code block inside try{} catch{}.

First it will try to execute code if there is nothing wrong it will return true. It means table and fields successfully created. If there is anything wrong while executing code it will catch those errors and return false. False means table and fields did not create successfully.

Using Try {} Catch{} page does not break on yellow dead screen if something goes wrong.

Check Table exist:

Using Umbraco system we can easily check table name see code bellow.

```
CheckTable() methods.
private static bool CheckTable(string tableName)
       {
           SqlServerHelper sqlHeplper = new SqlServerHelper(GlobalSettings.DbDSN);
           return sqlHeplper.Utility.CreateTableUtility().ContainsTable(tableName);
       }
Description: using GlobalSettings.DBDSN we get sql connection string,
SqlServerHelper object has containsTable() methods by passing table name it will
return bool value true or false. If there is table it will return true.

Make sure to add reference: using umbraco.DataLayer.SqlHelpers.SqlServer provided
by Umbraco CMS.
```

Intert Data:

If there is no table exist, we can create table using Create[29] Table() methods[30], it's time to insert data into

```
InsertTable() Methods:

public static bool InsertTable(string name, string street, string postalcode, string
country, string phone, string email, string website, string imageguid)
        {
           string tableName = "vcardimage";
            try
            {
                    if (!CheckTable(tableName))
                    {CreateTable(tableName); }
                    ISqlHelper sqlHelper = umbraco.BusinessLogic.Application.SqlHelper;
                    sqlHelper.ExecuteNonQuery(
                    @"insert into "+ tableName+ "(name,street,postalcode,country,phone,
email, website,imageguid, date)" +
                    "values(@name, @street, @postalcode, @country, @phone, @email,
@website, @imageguid, @date);",
                    sqlHelper.CreateParameter("@name", name),
                    sqlHelper.CreateParameter("@street", street),
                    sqlHelper.CreateParameter("@postalcode", postalcode),
                    sqlHelper.CreateParameter("@country", country),
                    sqlHelper.CreateParameter("@phone", phone),
                    sqlHelper.CreateParameter("@email", email),
                    sqlHelper.CreateParameter("@website", website),
                    sqlHelper.CreateParameter("@imageguid", imageguid),
                    sqlHelper.CreateParameter("@date", DateTime.Now));
                    return true;
            }
            catch(Exception)
            {
                return false;
            }

        }
```

[29] Inspired by http://umbraco.com/follow-us/blog-archive/2009/12/7/blog-4-umbraco-205-%E2%80%93-comments
[30] See createTable() methods on previous chapter.

Description of InsertTable() Methods:

- First we check if table exist or not using if(!CheckTable(tableName)) if not we call CreateTable() methods to create table by passing table name.
- Second we call ISqlHelper object of Umbraco System, ISqlHelper object has ExecuteNonQuery() methods, that requires query string and parameters.
- To add parameters we call another methods of ISqlHelper object called CreateParameters().
- Third if it's successfully inserted we return True means data inserted into table.
- Forth if not return false. Something happen wrong.
- Make sure to put code inside Try and Catch block to prevent yellow and death screen.

Note: make sure to reference using umbraco.DataLayer in order to use ISqlHelper object.

Putting all code together.

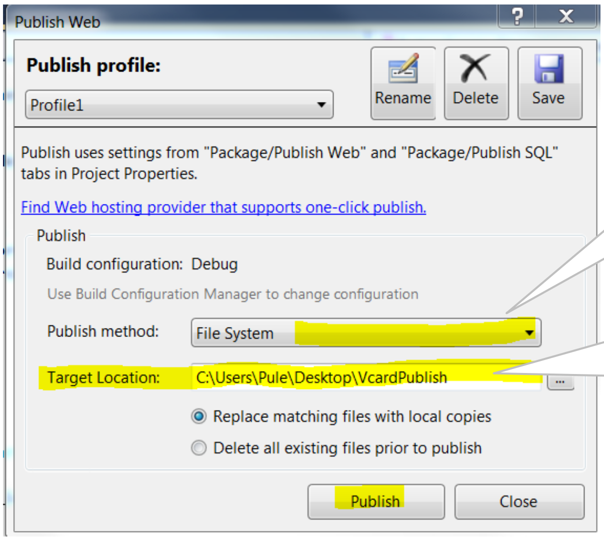
VcardControl.ascx.cs

For complete code, please visit appendix page and find page title "VcardControl.ascx.cs".

Implement User Control:

Until now we just finish creating usercontrol in Visual studio, its time to publish visual studio project, publishing visual studio project generate different types of files but we only need two types of files. Dot (.) dll file and .ascx file. Dll and .ascx (usercontrol) files should be copied into Umbraco site. DLL file goes inside bin directory of Umbraco site and .ascx file goes to "Usercontrol" directory of Umbraco sites. Follow the steps

- First publish visual project
- To publish project go to Build → Publish + your visual project name
- In my case publish VcardGenerator
- See captured screen for references
-



Publish profile: Profile1

Publish uses settings from "Package/Publish Web" and "Package/Publish SQL" tabs in Project Properties.

Find Web hosting provider that supports one-click publish.

Publish

Build configuration: Debug

Use Build Configuration Manager to change configuration

Publish method: File System

Target Location: C:\Users\Pule\Desktop\VcardPublish

● Replace matching files with local copies

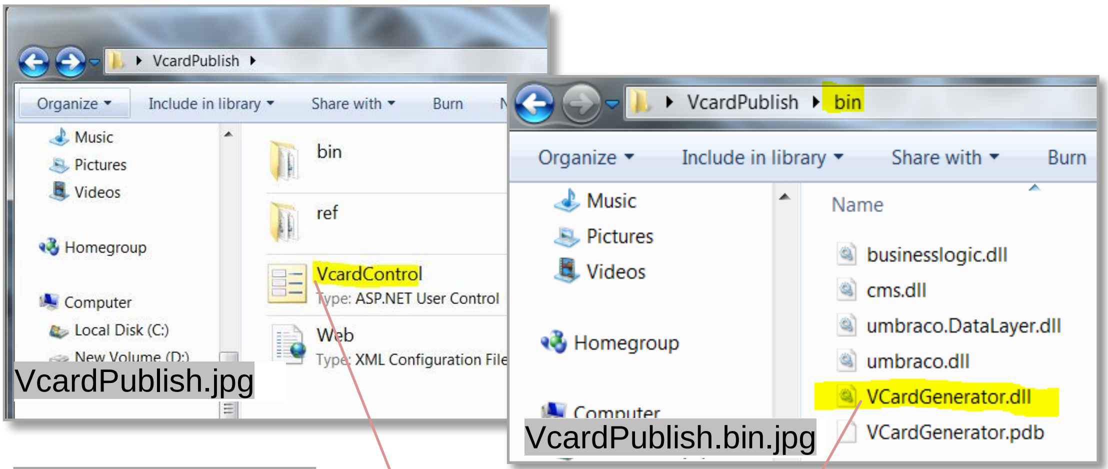
○ Delete all existing files prior to publish

Select "File System" from

Browse desire location to publish web site, in my case VcardPublish directory in Desktop.

Press

Let’s open published items director. In my case “VcardPublish” directory it must contain at least one directory “bin” and one user control “VcardControl” see “VcardPublish.jpg” and “VcardPublish.bin.jpg” image bellow



VcardPublish.jpg

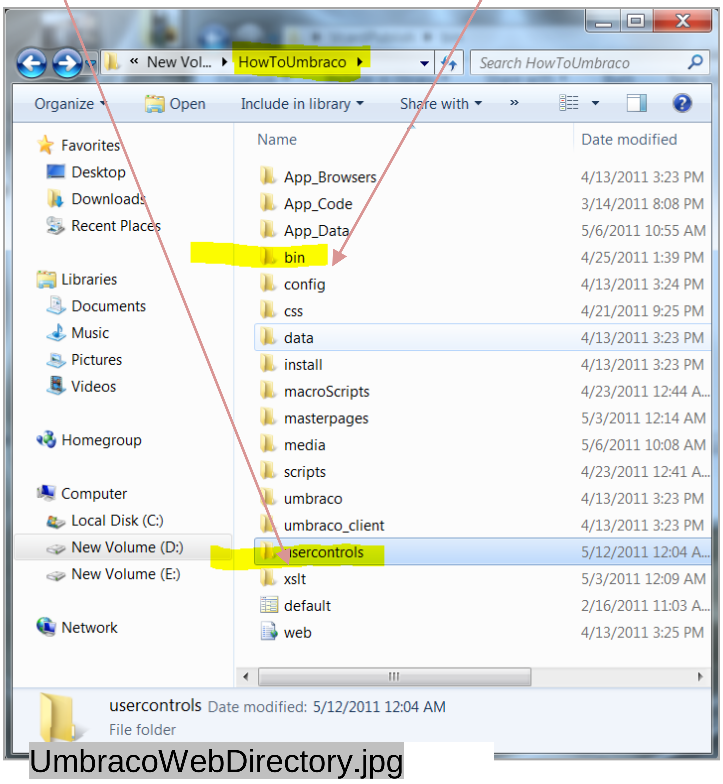
VcardPublish.bin.jpg

VcardPublish.jpg

It contains directory called “ref” I placed umbraco.dll, cms.dll, umbraco.DataLayer.dll file inside “ref” directory. Your case can be difference

VcardPublish.bin.jpg

It contains more than one dll file, you may get confuse. It’s because I added businesslogic.dll, cms.dll, umbraco.DataLayer.dll and umbraco.dll, all relevant dll file should be copied into Umbraco directory.



Name	Date modified
App_Browsers	4/13/2011 3:23 PM
App_Code	3/14/2011 8:08 PM
App_Data	5/6/2011 10:55 AM
bin	4/25/2011 1:39 PM
config	4/13/2011 3:24 PM
css	4/21/2011 9:25 PM
data	4/13/2011 3:23 PM
install	4/13/2011 3:23 PM
macroScripts	4/23/2011 12:44 A...
masterpages	5/3/2011 12:14 AM
media	5/6/2011 10:08 AM
scripts	4/23/2011 12:41 A...
umbraco	4/13/2011 3:23 PM
umbraco_client	4/13/2011 3:23 PM
usercontrols	5/12/2011 12:04 A...
xslt	5/3/2011 12:09 AM
default	2/16/2011 11:03 A...
web	4/13/2011 3:25 PM

usercontrols Date modified: 5/12/2011 12:04 AM
File folder

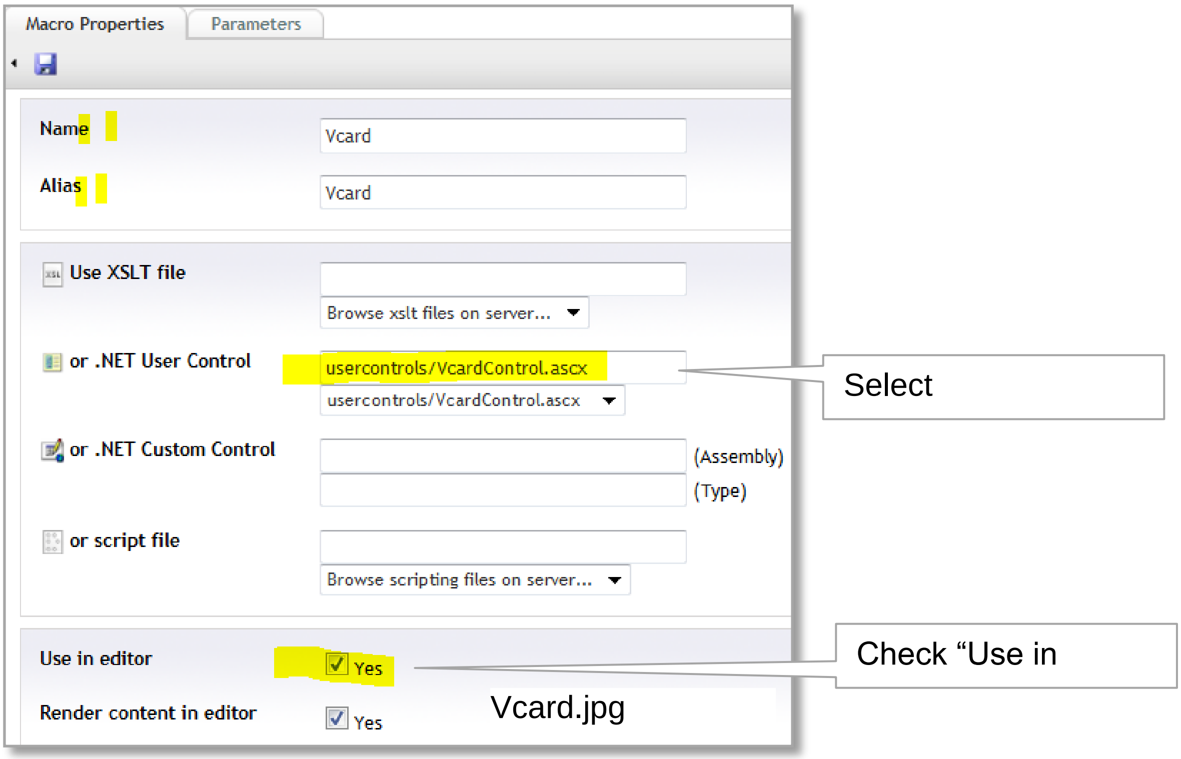
UmbracoWebDirectory.jpg

- **Asp.net User Control files should goes inside “usercontrols” directory**

- **VCardGenerator.DLL file goes inside “Bin” Directory see images “VcardPublish.jpg”, “VcardPublish.bin.jpg” and “UmbracoWebDirectory.jpg”.**

Create Macro to use User Control.

Copying Asp.net User Control to “usercontrol” directory and relevant DLL file into “bin” directory is not enough to use User Control, using User Control to Asp.net and using User Control to Umbraco site is different. In Umbraco site we can use User Control creating associated Marco. See image “vcard.jpg”



Name: Vcard

Alias: Vcard

Use XSLT file: Browse xslt files on server...

or .NET User Control: usercontrols/VcardControl.ascx

usercontrols/VcardControl.ascx

or .NET Custom Control: (Assembly) (Type)

or script file: Browse scripting files on server...

Use in editor: ☑ Yes

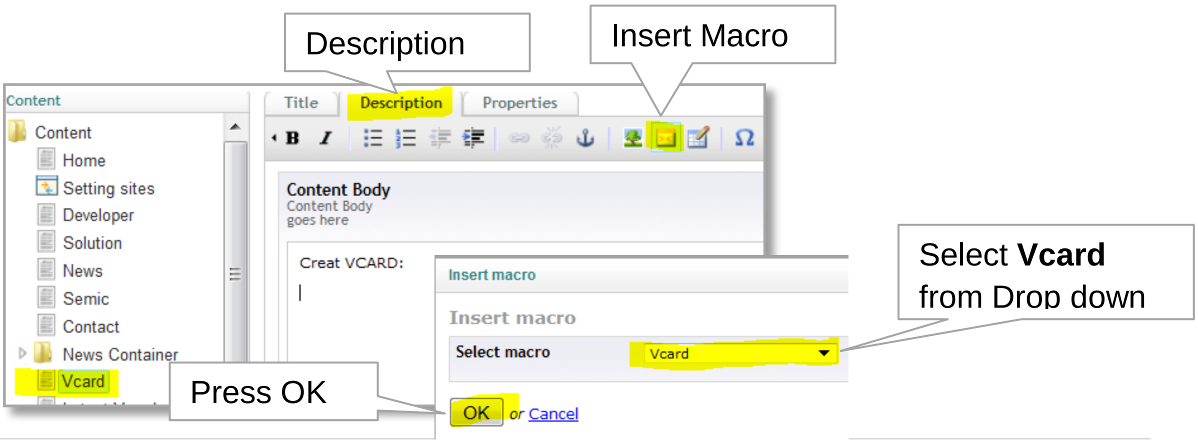
Render content in editor: ☑ Yes

Vcard.jpg

I created User Control “vcard”, selected User Control “VcardControl.ascx” Umbraco automatically add directory “usercontrol” and I checked “use in Editor”. Checked in “use in editor” will make macro available to use in Content Rich text editor.

Let’s create content and insert Vcard macro in content.

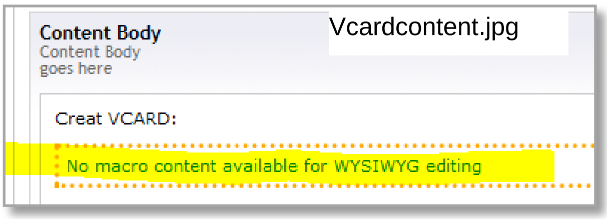
Open Umbraco back end also called Umbraco Back Office. Go to Section “Content” click “Create New. Choose “General Content” Document Type. Click “Description” Tab and navigate Insert Macro near insert image icon



Creat VCARD:

Select macro: Vcard

After pressing OK Button, you must see macro inserted into content; it looks like red dotted boarder line. See image bellow for reference.



Content Body
Content Body goes here

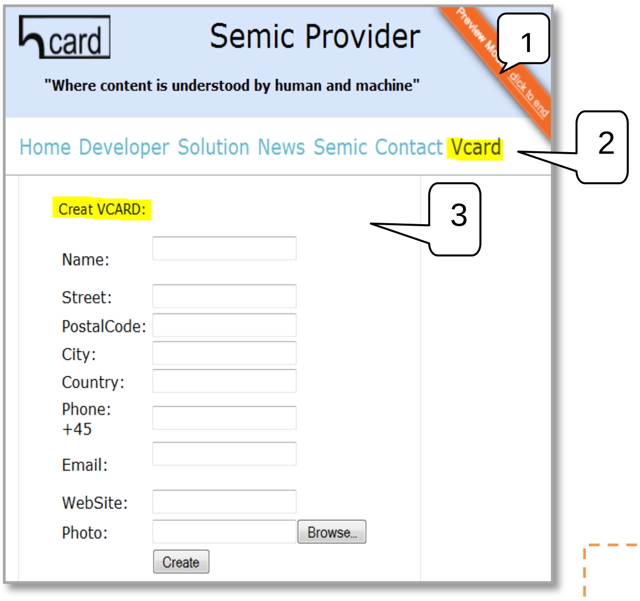
Creat VCARD:

No macro content available for WYSIWYG editing

A macro area is highlighted. See image “vcardcontent.jpg”

Lets Preview Vcard Content by clicking Preview Icon:

If you are following me, preview must look like this.

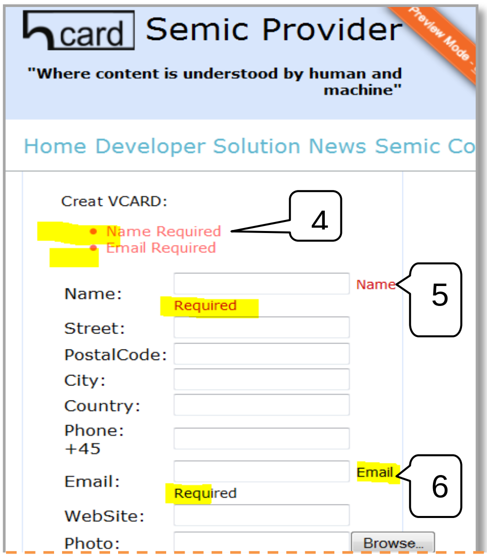


Semic Provider
"Where content is understood by human and machine"

Home Developer Solution News Semic Contact Vcard

Creat VCARD:

Name:
Street:
PostalCode:
City:
Country:
Phone: +45
Email:
WebSite:
Photo:



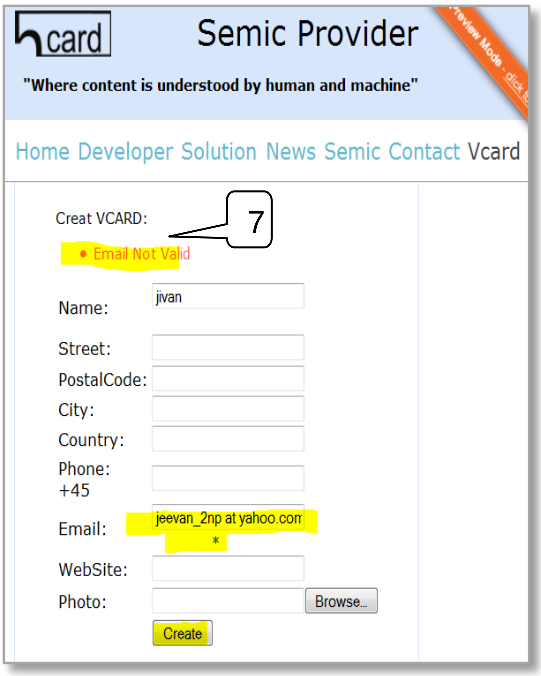
Semic Provider
"Where content is understood by human and machine"

Home Developer Solution News Semic Co

Creat VCARD:

- Name Required
- Email Required

Name: Required Name
Street:
PostalCode:
City:
Country:
Phone: +45
Email: Required Email
WebSite:
Photo:



Semic Provider
"Where content is understood by human and machine"

Home Developer Solution News Semic Contact Vcard

Creat VCARD:

- Email Not Valid

Name: jivan
Street:
PostalCode:
City:
Country:
Phone: +45
Email: jeevan_2np at yahoo.com *
WebSite:
Photo:

1- Red strip: preview mode

2- Vcard navigation bar show up

3- Input text box, Image upload, Create button environments

4- Error Summary. Press “Create” button without typing. It show up required field errors

5- Name: required error message

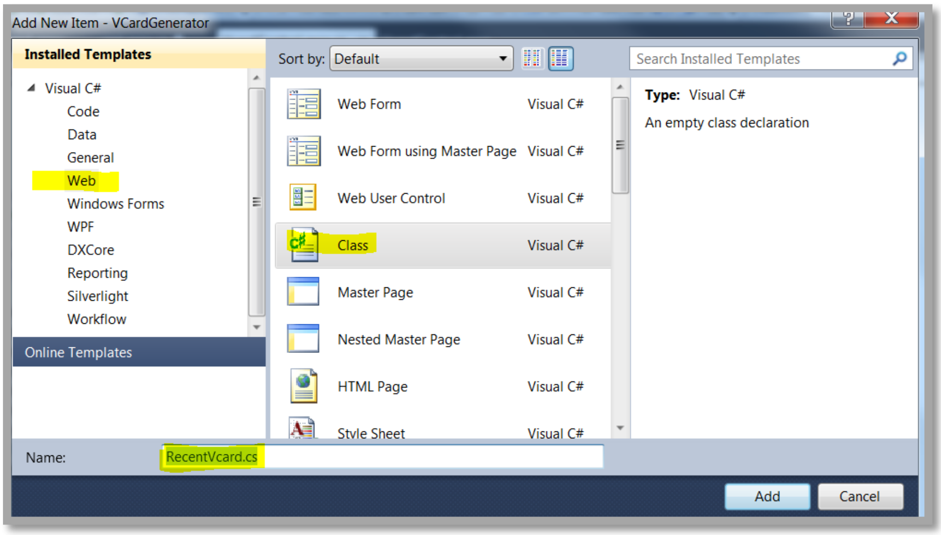
6- Email: Required error message

7- Validation: email validation works. I tried to input “jeevan_2np at yahoo.com” email id, which does not contain @ character, validation is not pass, it show up error message.

Extending Xslt Using .NET Class

In many cases Xslt may not be enough to fetch data or to process data in that case you can easily extend XSLT using .NET Class. Here I show how to use it, In Vcard website, people may interested to see recent Vcard. Name and address are fetched using .NET Class to extend XSLT.

Let's add Class. If you followed me, add new class in Visual studio VCardGenerator solution where we created User Controls. I choose class name "RecentVcard" and press Add. See captured screen bellow.



Add New Item - VCardGenerator

Installed Templates

- Visual C#
  - Code
  - Data
  - General
  - Web
  - Windows Forms
  - WPF
  - DXCore
  - Reporting
  - Silverlight
  - Workflow

Online Templates

Sort by: Default

Web Form	Visual C#
Web Form using Master Page	Visual C#
Web User Control	Visual C#
Class	Visual C#
Master Page	Visual C#
Nested Master Page	Visual C#
HTML Page	Visual C#
Style Sheet	Visual C#

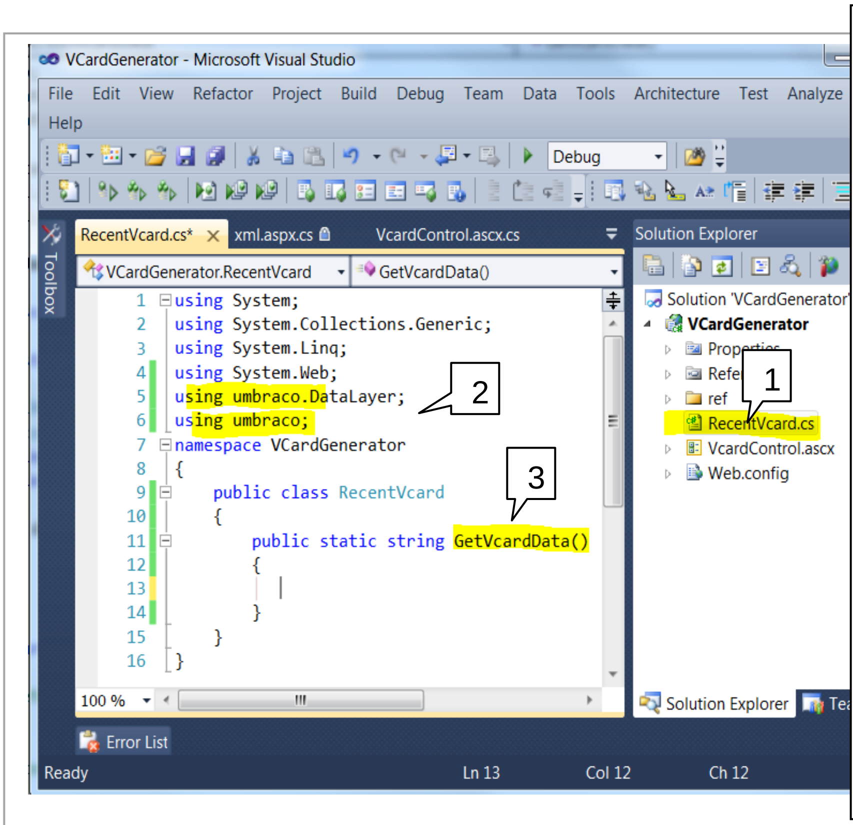
Search Installed Templates

Type: Visual C#

An empty class declaration

Name: RecentVcard.cs

Add Cancel



VCardGenerator - Microsoft Visual Studio

File Edit View Refactor Project Build Debug Team Data Tools Architecture Test Analyze Help
Debug
Toolbox
RecentVcard.cs* xml.aspx.cs VcardControl.ascx.cs
VCardGenerator.RecentVcard
GetVcardData()
Solution Explorer
Solution 'VCardGenerator'
VCardGenerator
Properties
References
ref
RecentVcard.cs
VcardControl.ascx
Web.config
Solution Explorer Tea
100 %
Error List
Ready Ln 13 Col 12 Ch 12
1
2
3

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using umbraco.DataLayer;
using umbraco;
namespace VCardGenerator
{
    public class RecentVcard
    {
        public static string GetVcardData()
        {
            |
        }
    }
}
```

1) Recently added Class "RecentVcard.cs"

2) I added two references, umbraco.DataLayer and Umbraco.

3) Method Name GetVcardData().

- **GetVcardData() method is going to use in XSLT so that it must be public access modifier and Static method declaration.**

Requirements: when you extend .NET Class you should have some basic knowledge which I did not cover here. My assumption is you already know basic things about

1) Adding References, in my case I am using Umbraco system to fetch data; you need to add Umbraco.DataLayer and Umraco references in your Visual Project.

2) Know how to select data from MsSql and sorting data (ascending or descending)

3) In Vcard web Recent Vcard will be selected top 10 lists from database and should be order by latest Vcard should be at top.

4) Looping. You should know how to loop data using while or foreach.

5) Know about static methods and where to use it. If you want to use methods in XSLT it must be declear as static and public access modifier.

Code inside GetVcardData() method.

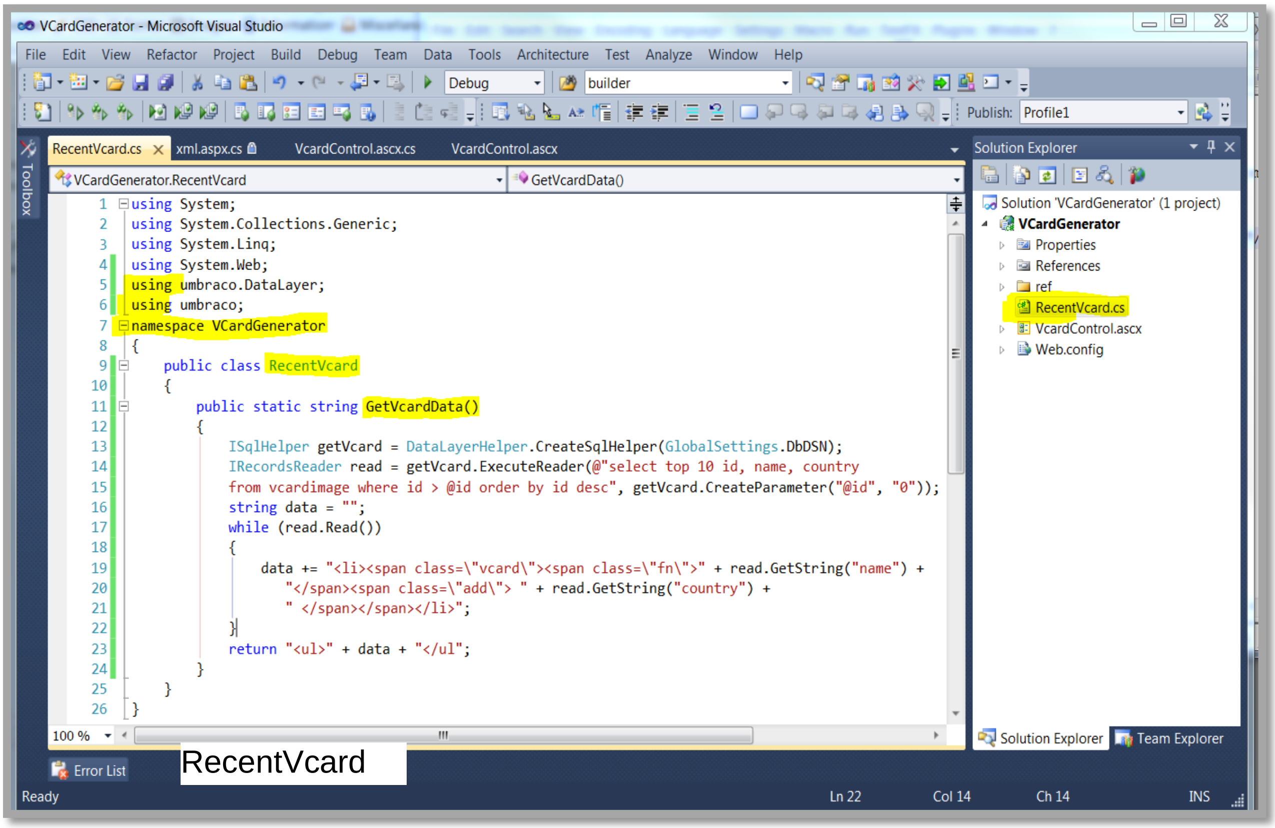
I am going use ISqlHelper class provided by Umbraco to fetch data from Database, but you can fetch using asp.net way if you like. Here I show you to fetch data from Database using Umbraco. See code bellow.

```
public static string GetVcardData()
        {
            ISqlHelper getVcard =
            DataLayerHelper.CreateSqlHelper(GlobalSettings.DbDSN);
            IRecordsReader read = getVcard.ExecuteReader(@"select top 10 id, name,
            country
            from vcardimage where id > @id order by id desc",
            getVcard.CreateParameter("@id", "0"));
            string data = "";
            while (read.Read())
            {
              data += "<li><span class=\"vcard\"><span class=\"fn\">" +
              read.GetString("name") +
              "</span><span class=\"add\"> " + read.GetString("country") +
              " </span></span></li>";
            }
            return "<ul>" + data + "</ul";
        }
```

Description:

- ISqlHelper object used to connect to database, GlobalSettings.DbDSN is use to retrieve database connection string from web.config file.
- IRecorderReader object contains method ExecuteReader(). ExecuteReater() has two parameters, first parameters is query string and second is parameters.
- “Select to 10 id, name, country” means select id, name and country from table name vcardimage and order by desc (desending) order.
- I use while loop to loop data.
- By calling GetString() methods Umbraco retrieve data as

If you are following me, RecentVcard class should looks like bellow, see captured screen. Now you need to publish your website. Go to *Build* → *Publish VCardGenerator*. Select *Publish Method* → *file System*. Select *Target Location* and press “*Publish*” button. It should generate *VCardControl.dll* file inside “Bin” Directory of your published directory. Copy that *VCardControl.dll* file to Umbraco website “*Bin*” Directory. If you are following me, you should already copied *VCardControl.dll* file into Umbraco website “*Bin*” Directory during User Control topics. You should replace old *VCardControl.dll* file with new *VCardControl.dll* file. We can register our *RecentVcard* class into Umbraco website *xsltExtensions.config* file.



```
 1  using System;
 2  using System.Collections.Generic;
 3  using System.Linq;
 4  using System.Web;
 5  using umbraco.DataLayer;
 6  using umbraco;
 7  namespace VCardGenerator
 8  {
 9      public class RecentVcard
10      {
11          public static string GetVcardData()
12          {
13              ISqlHelper getVcard = DataLayerHelper.CreateSqlHelper(GlobalSettings.DbDSN);
14              IRecordsReader read = getVcard.ExecuteReader(@"select top 10 id, name, country
15              from vcardimage where id > @id order by id desc", getVcard.CreateParameter("@id", "0"));
16              string data = "";
17              while (read.Read())
18              {
19                  data += "<li><span class=\"vcard\"><span class=\"fn\">" + read.GetString("name") +
20                     "</span><span class=\"add\"> " + read.GetString("country") +
21                     " </span></span></li>";
22              }
23              return "<ul>" + data + "</ul>";
24          }
25      }
26  }
```

RecentVcard

Overview of *VCardGenerator* visual studio solution

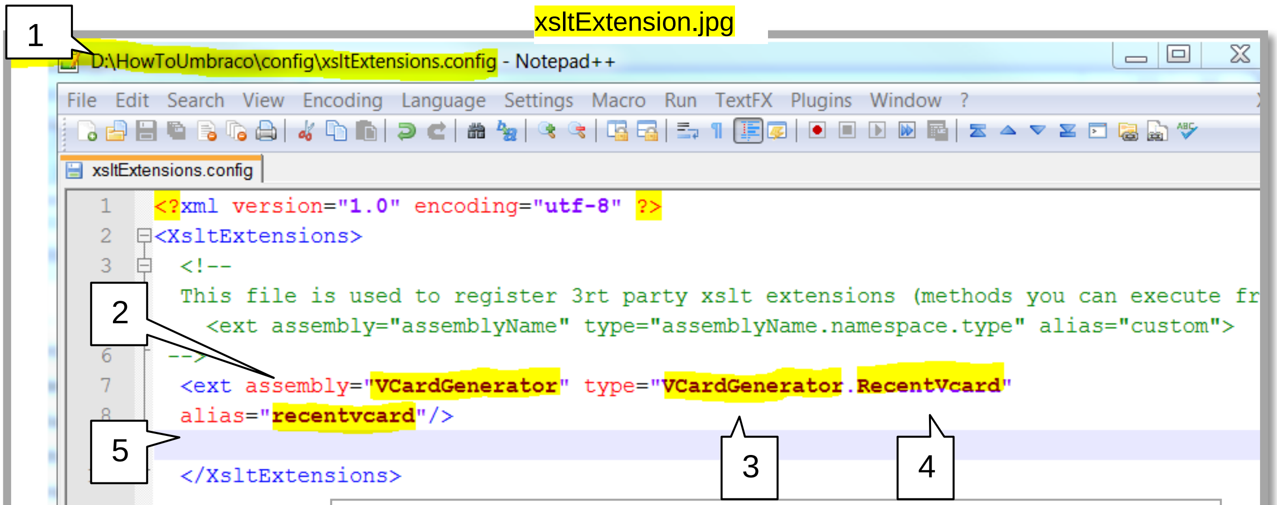
- One user Control “VcardControl.ascx” use to generate Vcard
- One Class “RecentVcard.cs” use to fetch data from Database
- Added two references
    1) Umbraco.DataLayer
    2) Umbraco

    See code line 5 and 6 in RecentVcard Class.

Registering .NET Class on config file

RecentVcard class returns latest ten list of recent vcard from database, before to use *RecentVcard* class, Umbraco requires registering first. We can use *RecentVcard* class only after registering *RecentVard* class into Umbraco *xsltExtensions.config* file. *xsltExtesion.config* file is xml file which can be found inside “config” directory in Umbraco website. Go To “config” directory and open *xsltExtension.config* file. You can either use visual studio or other software program, I am using Notepad ++[31] open source software. See captured screen bellow.

xsltExtension.jpg


1
D:\HowToUmbraco\config\xsltExtensions.config - Notepad++
File Edit Search View Encoding Language Settings Macro Run TextFX Plugins Window ?
xsltExtensions.config
2
5
3
4

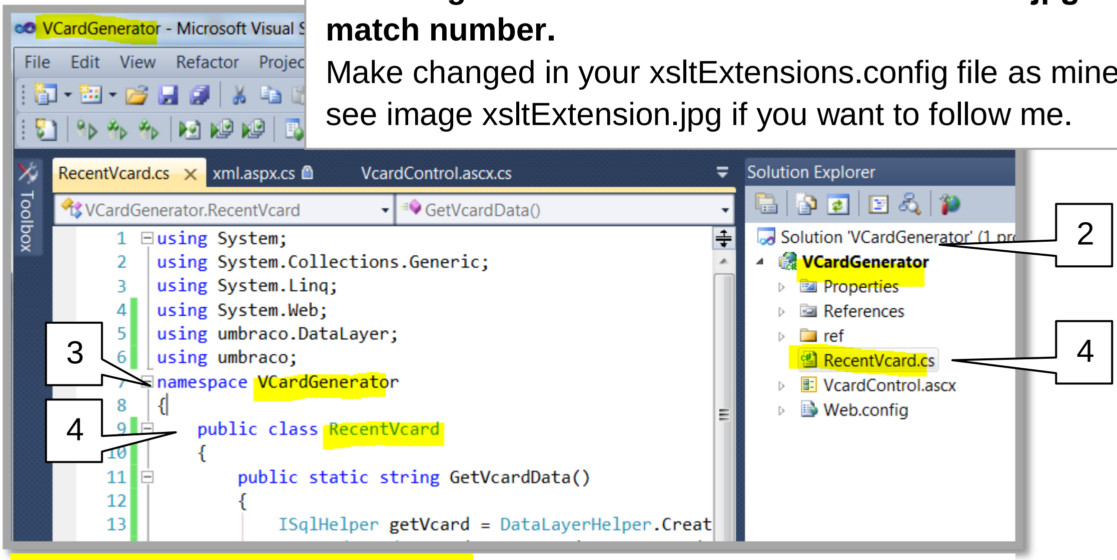
```
<?xml version="1.0" encoding="utf-8" ?>
<XsltExtensions>
  <!--
  This file is used to register 3rt party xslt extensions (methods you can execute fr
    <ext assembly="assemblyName" type="assemblyName.namespace.type" alias="custom">
  -->
  <ext assembly="VCardGenerator" type="VCardGenerator.RecentVcard"
  alias="recentvcard"/>

  </XsltExtensions>
```

1) File location of xsltExtensions.config
2) Assembly name “VcardGenerator”
3) Type = first one before dot is Namespace
4) Type = second after dot is Class name
5) Alias : it is use in Xslt to refer assembly

Did you get confuse?

See image bellow “VCardGenerator.extension.jpg” and match number.

Make changed in your xsltExtensions.config file as mine see image xsltExtension.jpg if you want to follow me.



```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using umbraco.DataLayer;
using umbraco;
namespace VCardGenerator
{
    public class RecentVcard
    {
        public static string GetVcardData()
        {
            ISqlHelper getVcard = DataLayerHelper.Creat
```

VCardGenerator.extension.jpg

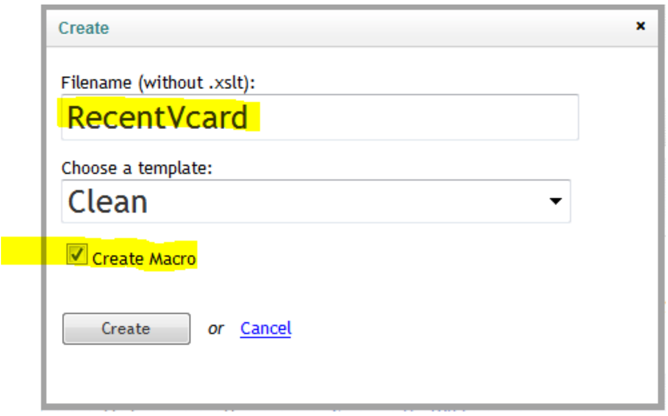
[31] http://notepad-plus-plus.org/

After made change on xsltExtensions.config file, make sure to save changed.

Extending XSLT using .net Class Practice

We have now VCardGenerator.dll file inside “Bin” directory and we also registered our Class on “xsltExtentions.config” file. Remember alias we specified at “xsltExtensions.config” file. If you are following me, alias is “recentvcard”.

Create Xslt:



Create ×

Filename (without .xslt):
RecentVcard

Choose a template:
Clean

☑ Create Macro

I have created new xslt for Recent Vcard, see captured screen left. Go to backend part of umbraco website, Select → Developer → XSLT Files → Right Click and click on “Create”.

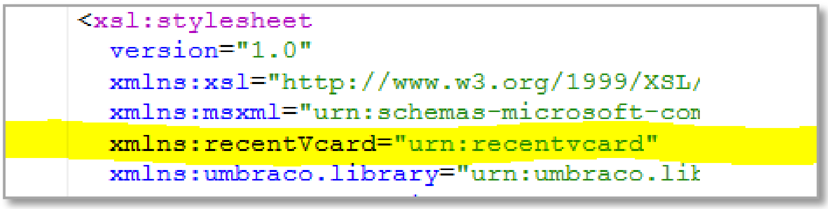
Filename: “ RecentVcard”

Choose a Template: “Clean”

Checked: “Create Macro” by checked in Umbraco will create matching macro for us automatically,

Press → *Create* Button.

Open “RecentVcard.xslt”

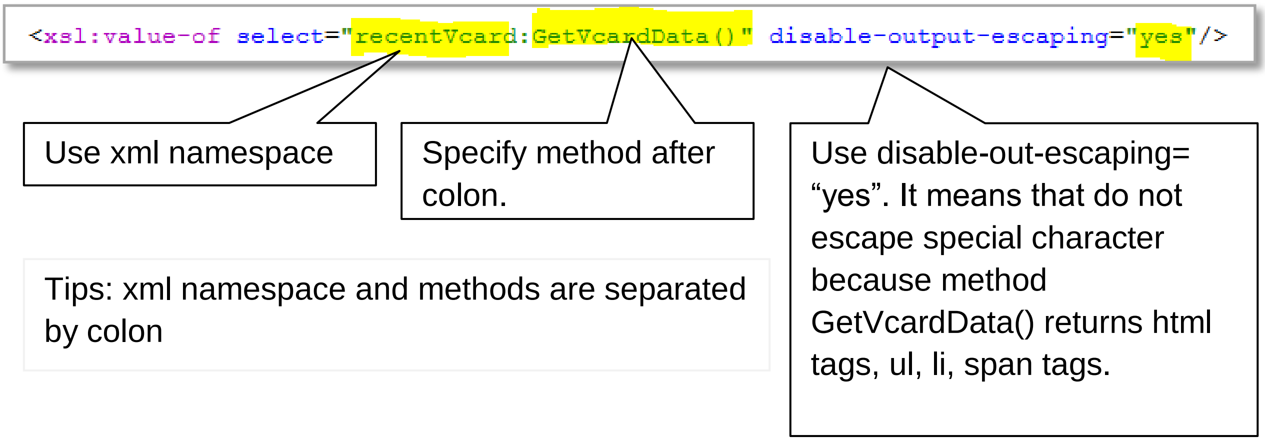


```
<xsl:stylesheet
  version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/
  xmlns:msxml="urn:schemas-microsoft-com
  xmlns:recentVcard="urn:recentvcard"
  xmlns:umbraco.library="urn:umbraco.lib
```

We can registered .net class inside xsl:stylesheet, using xslns. See the captured screen right. See highlighted area on right image, if you already worked with Xslt you may know xmlns. Xmlns is xml namespace for a document. After colon you provide namespace name, I choose “recentVcard”. You can choose different if you want. Use urn colon alias that you specified at xsltExtensions.config” see “xsltExtension.jpg” image, I my case alias is “recentvcard”. URN is Uniform Resource Name.

public static method inside .NET class can be access using xml namespace in XSLT. In RecentVcard Class has one static public methods called *GetVcardData()*. To access that public method we can use namespace colon and method name. If there are any parameters, we can easily pass. See example bellow how I accessed GetVcardData() method.

```
<xsl:value-of select="recentVcard:GetVcardData()" disable-output-escaping="yes"/>
```



```
<xsl:value-of select="recentVcard:GetVcardData()" disable-output-escaping="yes"/>
```

Use xml namespace

Specify method after colon.

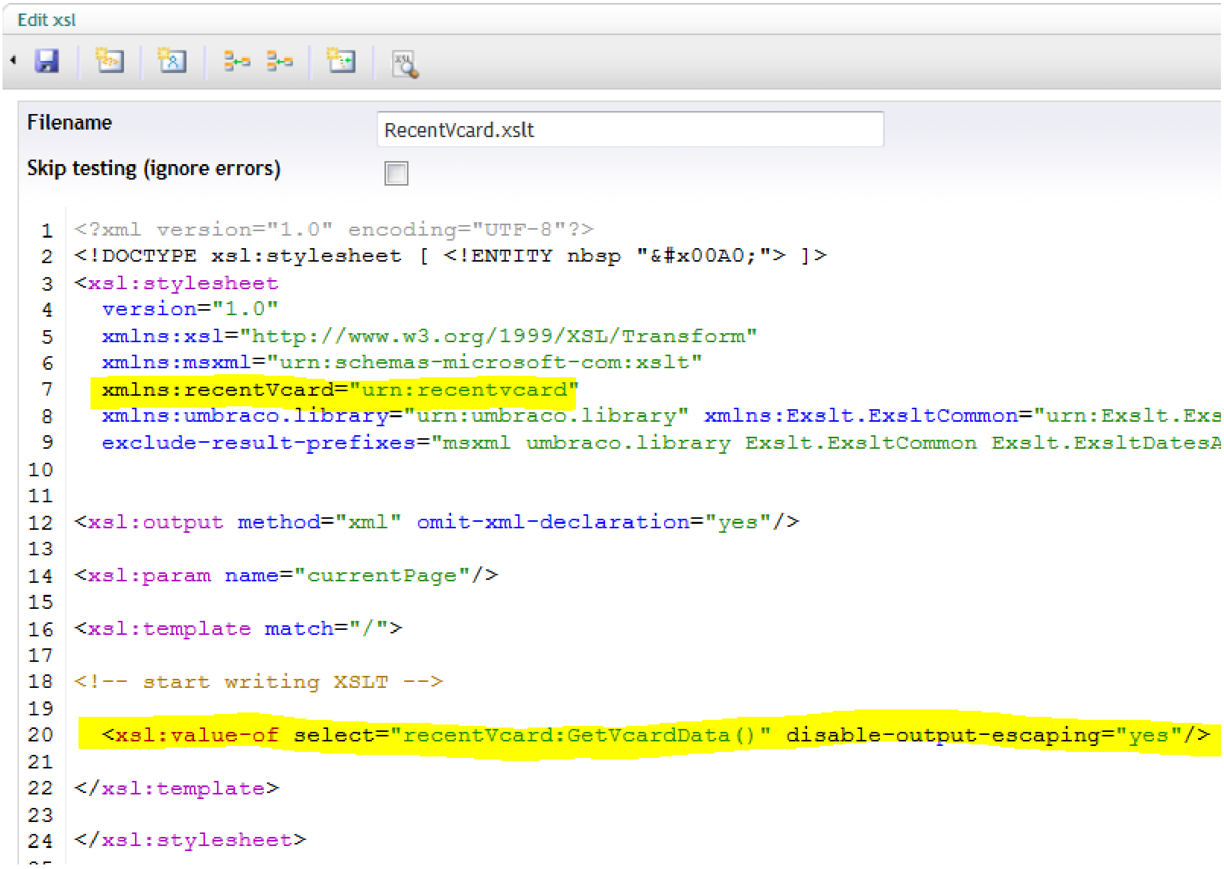
Use disable-out-escaping= “yes”. It means that do not escape special character because method GetVcardData() returns html tags, ul, li, span tags.

Tips: xml namespace and methods are separated by colon

Code all in one:

RecentVcard.xslt

Highlighted text are added, rest of code are default provided by Umbraco system when you create Xslt.



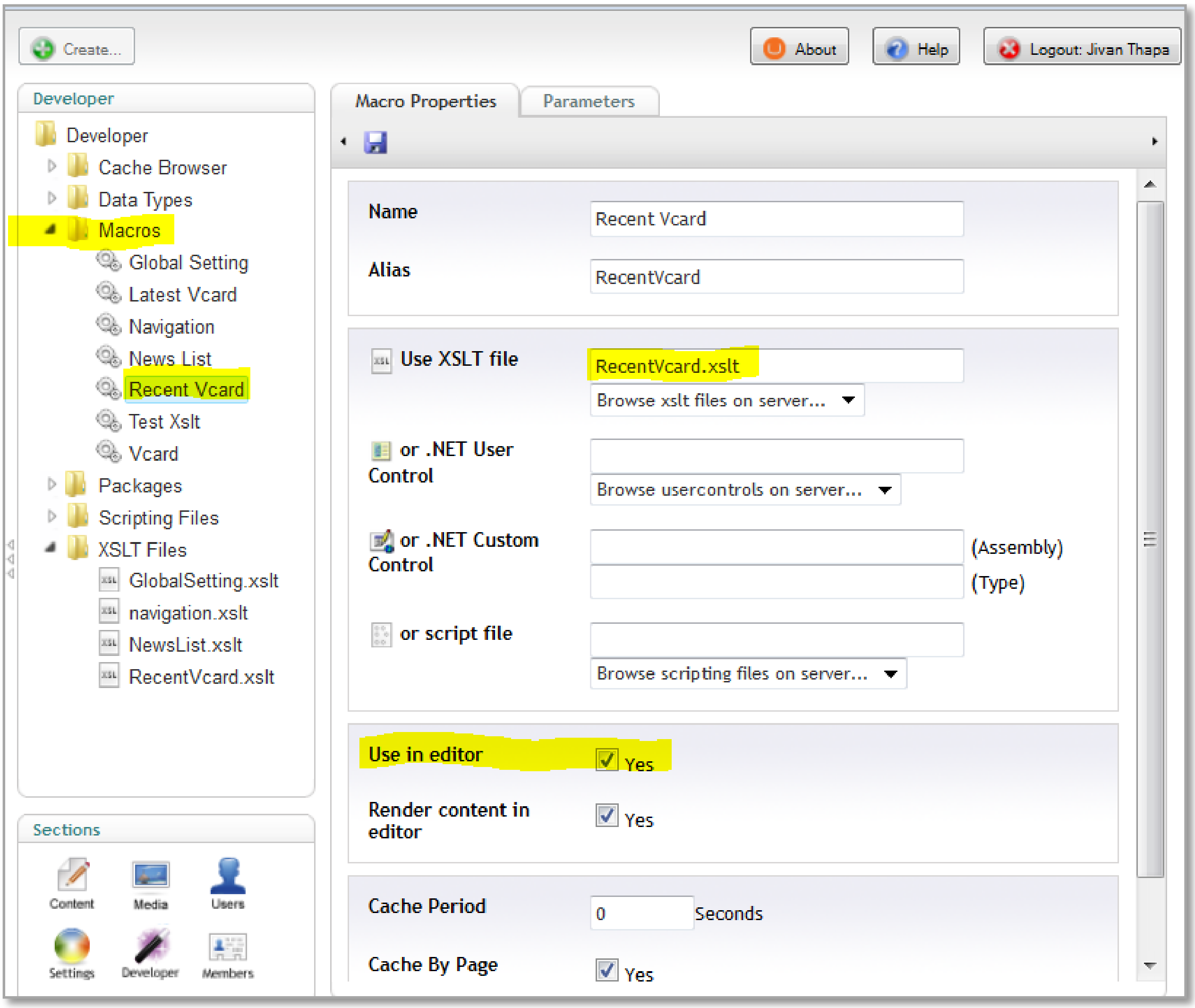
Filename: RecentVcard.xslt

Skip testing (ignore errors) □

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE xsl:stylesheet [ <!ENTITY nbsp "&#x00A0;"> ]>
<xsl:stylesheet
  version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:msxml="urn:schemas-microsoft-com:xslt"
  xmlns:recentVcard="urn:recentvcard"
  xmlns:umbraco.library="urn:umbraco.library" xmlns:Exslt.ExsltCommon="urn:Exslt.Exs
  exclude-result-prefixes="msxml umbraco.library Exslt.ExsltCommon Exslt.ExsltDatesA


<xsl:output method="xml" omit-xml-declaration="yes"/>

<xsl:param name="currentPage"/>

<xsl:template match="/">

<!-- start writing XSLT -->

  <xsl:value-of select="recentVcard:GetVcardData()" disable-output-escaping="yes"/>

</xsl:template>

</xsl:stylesheet>
```

Modify Macro:

Once you have XSLT, you can insert Xslt into templates, in my case I am going to insert into Content. So we need to check “Use in editor” option. Go to →Macros → Open “Recent Vcard” → checked on “Use in Editor” and Remember to “*Save”*. See captured screen bellow for references.



Name: Recent Vcard

Alias: RecentVcard

Use XSLT file: RecentVcard.xslt

or .NET User Control:

or .NET Custom Control: (Assembly) (Type)

or script file:

Use in editor ☑ Yes

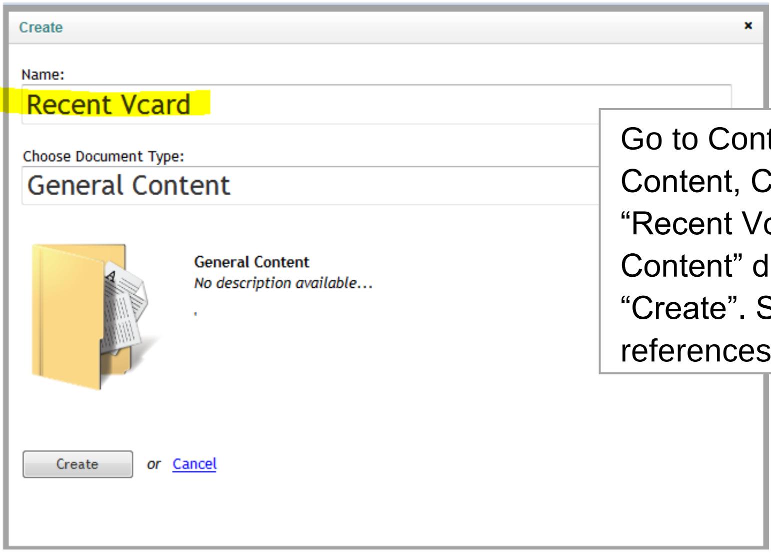
Render content in editor ☑ Yes

Cache Period 0 Seconds

Cache By Page ☑ Yes

By selecting “Use in editor” makes macro available to use in content section.

Create Content: “Recent Vcard”

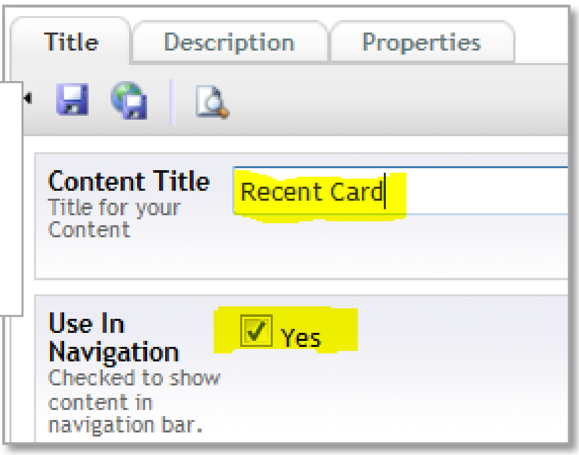


Name: Recent Vcard

Choose Document Type: General Content

General Content
No description available...

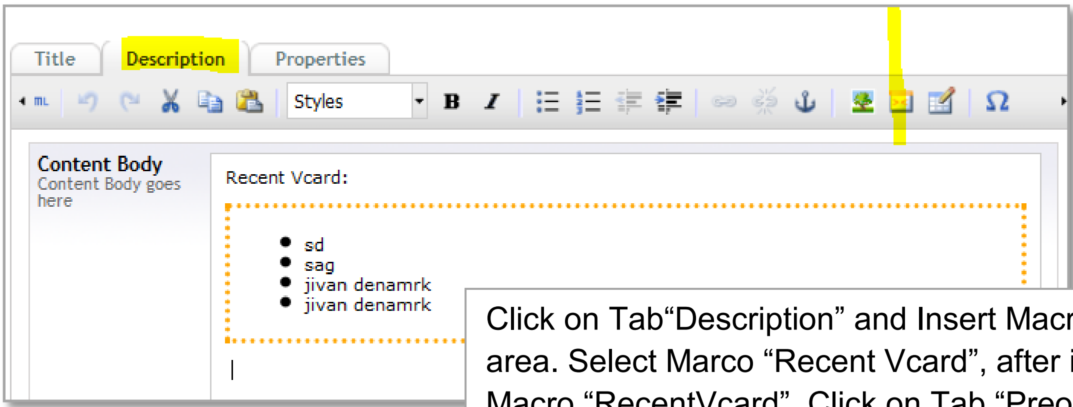
Go to Content Section → Right Click on Content, Click “Create” → Name “Recent Vcard” , Select “General Content” document type . Press button “Create”. See captured screen for references.



Content Title
Title for your Content: Recent Card

Use In Navigation
Checked to show content in navigation bar.: ☑ Yes

It should create Content “Recent Vcard” and visible at left side menu. Provide following information. “Content Title”: Recent Card and checked in use in Navigation so that “Recent Vcard” will show up in navigation bar



Content Body
Content Body goes here

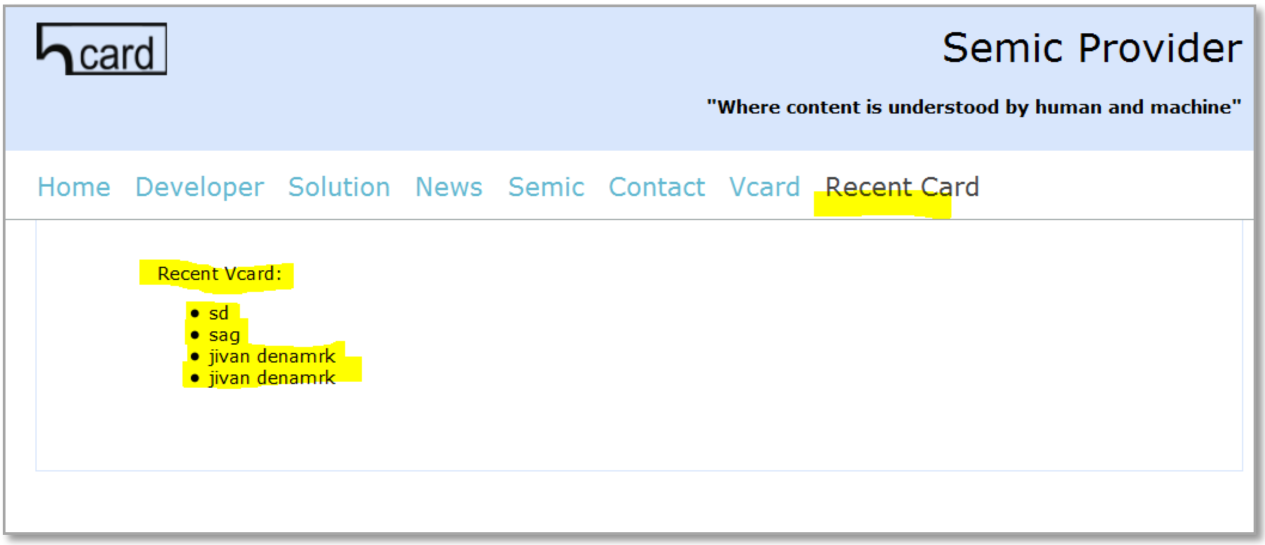
Recent Vcard:

- sd
- sag
- jivan denamrk
- jivan denamrk

Click on Tab“Description” and Insert Macro on content area. Select Marco “Recent Vcard”, after inserting Macro “RecentVcard”. Click on Tab “Preoperties” Select Template “General Content” and Press “Save and Publish Icon”. Create few Vcard and check. It should show up on web site.

Preview content by click on preview icon.

If you are following me, Vcard website should look like bellow captured screen.



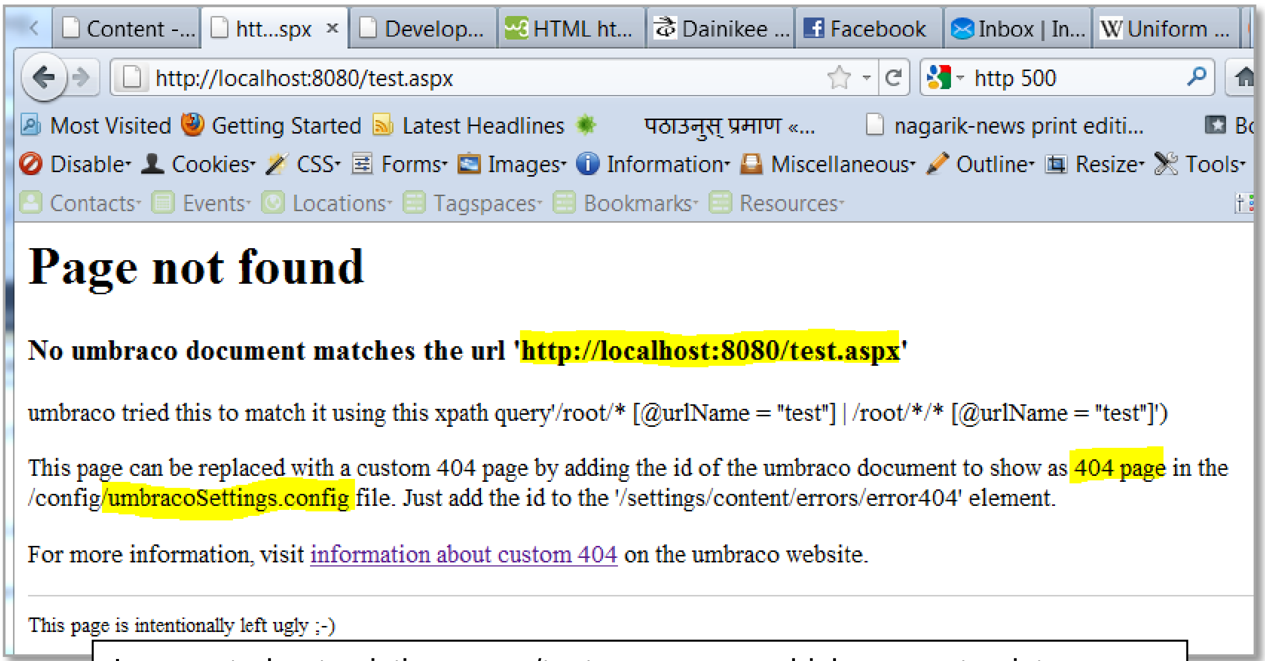
card
Semic Provider
"Where content is understood by human and machine"
Home Developer Solution News Semic Contact Vcard Recent Card

Recent Vcard:

- sd
- sag
- jivan denamrk
- jivan denamrk

Page not Found Error:

It is nice to replace default Umbraco error HTTP handler, there are many error handler 500, 404 and 403 etc. 404 error are page not found error, due to some reason, accidental file deletion, dead links and user request not existing page causes HTTP 404 errors. Umbraco handles 404 errors by showing its default error message. We can easily override 404 errors with custom page. First let's see Umbraco default error 404 page.



Page not found

No umbraco document matches the url 'http://localhost:8080/test.aspx'

umbraco tried this to match it using this xpath query'/root/* [@urlName = "test"] | /root/*/* [@urlName = "test"]')

This page can be replaced with a custom 404 page by adding the id of the umbraco document to show as 404 page in the /config/umbracoSettings.config file. Just add the id to the '/settings/content/errors/error404' element.

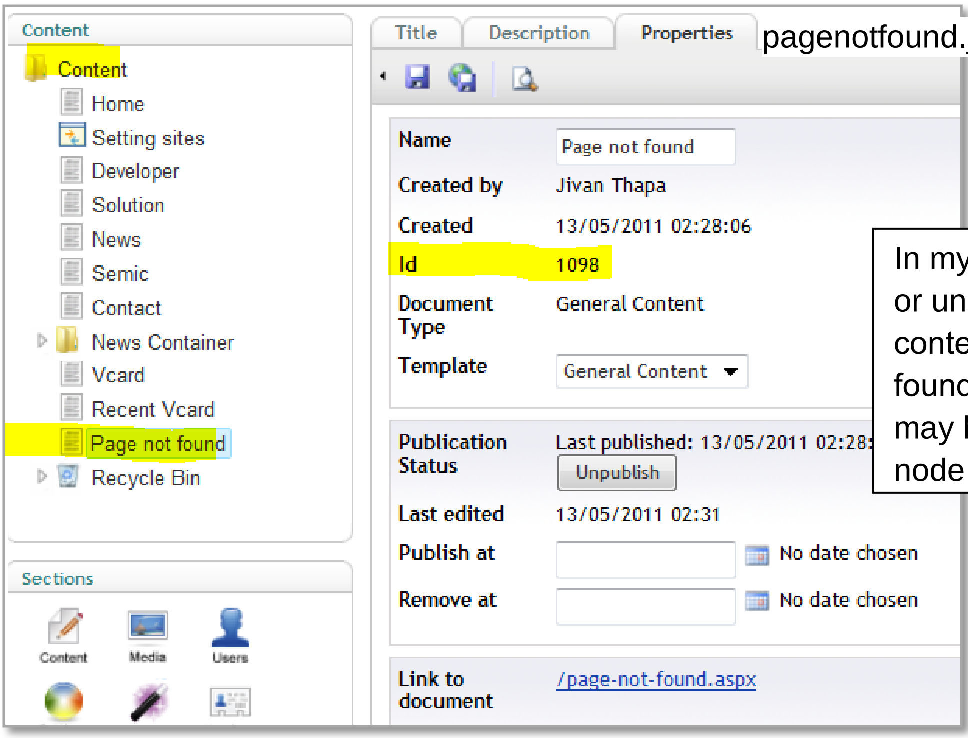
For more information, visit information about custom 404 on the umbraco website.

This page is intentionally left ugly ;-)

I requested not existing page /test.aspx page, which was not exist. Umbraco handle 404 error. See highlighted text, it explain well, what to do. Sometimes it may not enough so need to extra work or needs to add extra code in web.config file.

Create Custom 404 error page.

I am creating simple 404 error page. If you want to follow me go →Content → Right click and click on "Create". Name: "Page not Found" and Select "General Content" Document Type. Do not check "Use in Navigation" because I don't want to show this page into navigation. This page should be displayed only if 404 errors occur. Go to Tab "Title" provide content title "Page Not Found", Go to Tab "Description" and type "Requested page not found, temporarily unavailable or an out-of-date bookmark/favorite" Go to Tab "Properties" Template → "General Content". Finally press "Save and Publish Icon". Hold a second on Properties tab and look id of content because Umbraco needs content id. See "pagenotfound.jpg" image as references. Each content item gets unique id. We need to specify unique id of items into umbracoSettings.config files.



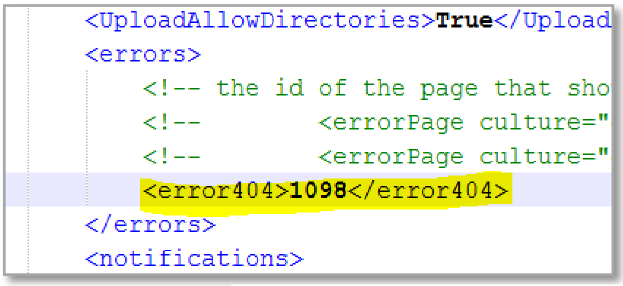
Name	Page not found
Created by	Jivan Thapa
Created	13/05/2011 02:28:06
Id	1098
Document Type	General Content
Template	General Content
Publication Status	Last published: 13/05/2011 02:28:
Last edited	13/05/2011 02:31
Publish at	No date chosen
Remove at	No date chosen
Link to document	/page-not-found.aspx

pagenotfound.jpg

In my case node id or unique id for content “Page not found” is 1098. You may have different node or unique id

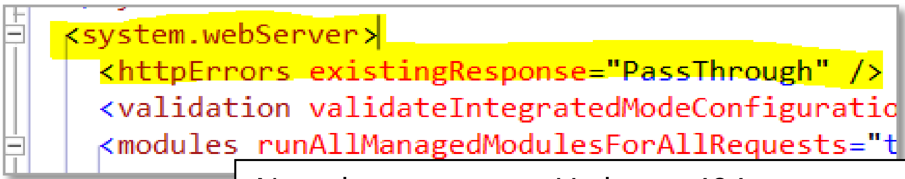
Modifying umbracoSettings.config file:

umbracoSettings.config file can be found inside “config” directory, open umbracoSettings.config file using notepad or other program. It’s xml document, find <errors> elements inside umbracoSettings.config. *errors* element contain child *error404* element. Set its value to node id of content, in my case 1098 and Save umbracoSettings.config file. See captured screen bellow for references.



```
<UploadAllowDirectories>True</Upload
<errors>
    <!-- the id of the page that sho
    <!--        <errorPage culture="
    <!--        <errorPage culture="
    <error404>1098</error404>
</errors>
<notifications>
```

Umbraco error404.jpg



```
<system.webServer>
  <httpErrors existingResponse="PassThrough" />
  <validation validateIntegratedModeConfiguratio
  <modules runAllManagedModulesForAllRequests="t
```

Note: in some cases Umbraco 404 error may override by IIS (internet Information Service) or IIS may not hand over 404 error to Umbraco to handle in such case 404 error page will end up with ugly yellow dead screen page. To prevent from this we can specify by telling IIS not to handle, instead hand over all 404 error to Asp.net web site in our case Umbraco site by adding few lines of code into web.config file. Web.config file can be found in root directory of Umbraco System. Find system.webserver elements and add new element <httpErrors existingResponse="PassThrough" /> inside system.webserver elements. Using existingRespone= “passThrough” means IIS does not override custom errors.

Conclusion:

Learning process is not difficult if we start from basic, every complex system begins with basic and simple structure. Here I provided basic concept of Umbraco using examples rather than theory. I use Vcard as a case to demonstrate practical examples of basic concept of Umbraco. Vcard was virtual website it does not exist on live but concept that I used in Vcard was easy to understand how to start building Umbraco website from scratch. I started to show installation of Umbraco cms, most of new developer faces many problems to install. Developing Umbraco site from empty Umbraco installation is great challenge to new developer but it’s a way to learn Umbraco from inside, I started to build Vcard site from scratch, empty Umbraco installation. Creating DocumentType, Template and Content is core of Umbraco CMS. If you reading books from beginning without skipping chapter you should be able to understand connection between DocumentType, Template and Content. I provided information about fetching data from xml using xslt and extending xslt using .net class provided basic concept of how to extend Umbraco. All these concepts and examples can be used for your own purpose and needs. You are familiar with how to fetch and insert data from database to Umbraco and Umbraco to Database, extending Umbraco using Xslt, user control and .net class. In last chapter I provided how to solve umbraco 404 error default page problem with custom 404 errors page in case you get errors. I also touch about security on website which will provide knowledge to secure Umbraco site even though security is not basic on Umbraco sites.

References:

Taken date: May-17-2011

Xslt and Xpath

http://www.w3schools.com/xsl/

Umbraco to Database and Database to Umbraco

http://our.umbraco.org/forum/developers/api-questions/12248-Get-value-from-base-into-database

http://umbraco.com/follow-us/blog-archive/2009/12/7/blog-4-umbraco-205---comments

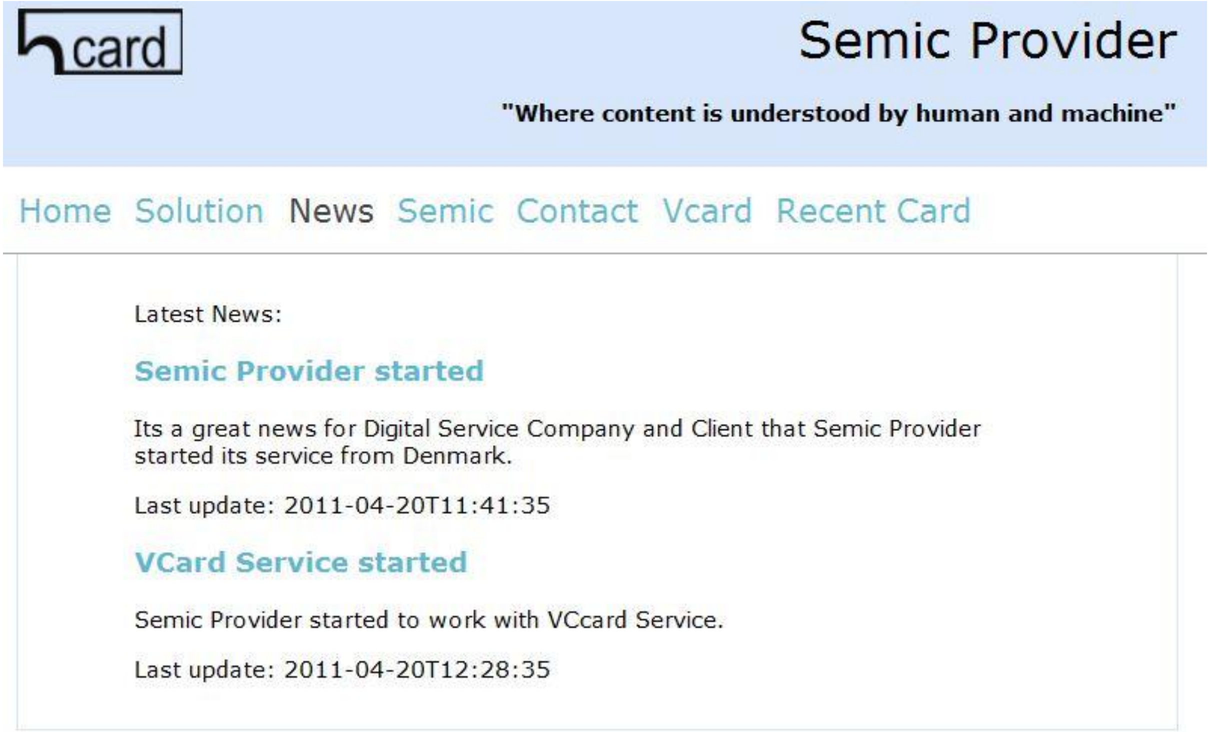
Wiki

http://our.umbraco.org/wiki

http://our.umbraco.org/wiki/recommendations/recommended-reading-for-net-developers

Extra files:

Web grid.



Semic Provider

"Where content is understood by human and machine"

Latest News:

Semic Provider started

Its a great news for Digital Service Company and Client that Semic Provider started its service from Denmark.

Last update: 2011-04-20T11:41:35

VCard Service started

Semic Provider started to work with VCcard Service.

Last update: 2011-04-20T12:28:35

File “VcardControl.ascx”

```
<%@ Control Language="C#" AutoEventWireup="true" CodeBehind="VcardControl.ascx.cs"
Inherits="VCardGenerator.VcardControl" %>

<form id="FormVcard" runat="server">
<div class="errorMessage">
            <asp:Label ID="LabelNotice" runat="server" Text=""></asp:Label>
    <asp:ValidationSummary ID="ValidationSummary" runat="server" ForeColor="#FF6666"
/>
</div>
<table >
    <tr class="name">
        <td> Name: </td><td><asp:TextBox ID="TextBoxName" runat=server></asp:TextBox>
        <asp:RequiredFieldValidator ID="RequiredFieldValidatorName" runat="server"
            ControlToValidate="TextBoxName" ErrorMessage="Name Required" Font-
Size="Small"
            ForeColor="#CC0000"></asp:RequiredFieldValidator>
        <asp:CustomValidator ID="CustomValidatorName" runat="server"
            ControlToValidate="TextBoxName" ErrorMessage="Name is not Valid"
            ForeColor="#CC0000"
onservervalidate="CustomValidatorName_ServerValidate">*</asp:CustomValidator>
        </td>
    </tr>
    <tr class="street">
        <td> Street:</td><td> <asp:TextBox ID="TextBoxStreet"
runat="server"></asp:TextBox>
        </td>
    </tr>
    <tr class="postalCode">
        <td> PostalCode:</td><td> <asp:TextBox ID="TextBoxPostalCode"
runat="server"></asp:TextBox>
        <asp:CustomValidator ID="CustomValidatorPostalCode" runat="server" Font-
Size="Small"
            ErrorMessage="Postal Code 4 digit" ControlToValidate="TextBoxPostalCode"

onservervalidate="CustomValidatorPostalCode_ServerValidate">*</asp:CustomValidator>
        </td>
    </tr>
    <tr class="city">
        <td> City:</td><td> <asp:TextBox ID="TextBoxCity"
runat="server"></asp:TextBox>
    </tr>
    <tr class="country">
        <td> Country:</td><td> <asp:TextBox ID="TextBoxCountry"
runat="server"></asp:TextBox>
        <asp:CustomValidator ID="CustomValidatorCountry" runat="server" Font-
Size="Small"
            ErrorMessage="Country Not Valid" ControlToValidate="TextBoxCountry"

onservervalidate="CustomValidatorCountry_ServerValidate">*</asp:CustomValidator>
        </td>
    </tr>
    <tr class="phone">
        <td> Phone:&nbsp; +45</td><td><asp:TextBox ID="TextBoxPhone"
runat="server"></asp:TextBox>
        <asp:CustomValidator ID="CustomValidatorPhone" runat="server" Font-
Size="Small"
            ErrorMessage="Phone must be 8 digit, Without +45 "
            ControlToValidate="TextBoxPhone"

onservervalidate="CustomValidatorPhone_ServerValidate">*</asp:CustomValidator>
        </td>
    </tr>
    <tr class="email">
        <td> Email: </td><td><asp:TextBox ID="TextBoxEmail" runat="server"
            ontextchanged="TextBoxEmail_TextChanged"></asp:TextBox>
        <asp:RequiredFieldValidator ID="RequiredFieldValidatorEmail" runat="server"
Font-Size="Small"
            ErrorMessage="Email Required" ControlToValidate="TextBoxEmail">Email
Required</asp:RequiredFieldValidator>
        <asp:CustomValidator ID="CustomValidatorEmail" runat="server" Font-
Size="Small"
            ErrorMessage="Email Not Valid" ControlToValidate="TextBoxEmail"

onservervalidate="CustomValidatorEmail_ServerValidate">*</asp:CustomValidator>
        </td>
    </tr>
    <tr class="website">
        <td> WebSite:</td><td> <asp:TextBox ID="TextBoxWebsite"
runat="server"></asp:TextBox>
        <asp:CustomValidator ID="CustomValidatorWebSite" runat="server"
            ErrorMessage="Website not valid" ControlToValidate="TextBoxWebsite"

onservervalidate="CustomValidatorWebSite_ServerValidate">*</asp:CustomValidator>
        </td>
     </tr>
    <tr class="photo">
        <td> Photo:</td><td> <asp:FileUpload ID="FileUploadPhoto" runat="server" />
        </td>
    </tr>
    <tr >
        <td>
            &nbsp;</td> <td> <asp:Button ID="ButtonCreate" runat="server"
Text="Create"
            onclick="ButtonCreate_Click" /> </td>
    </tr>

</table>
</form>
```

File VcardControl.ascx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using umbraco.DataLayer.SqlHelpers.SqlServer;
using umbraco.DataLayer;
using umbraco;
using System.Data.SqlClient;
using System.Text.RegularExpressions;
using System.Web.UI.WebControls;
using System.IO;
using System.Text;
using System.Net.Mail;
using System.Net;

namespace VCardGenerator
{
    public partial class VcardControl : System.Web.UI.UserControl
    {
        /// <summary>
        /// Handles the Load event of the Page control.
        /// </summary>
        /// <param name="sender">The source of the event.</param>
        /// <param name="e">The <see cref="System.EventArgs"/> instance containing the
event data.</param>
        protected void Page_Load(object sender, EventArgs e)
        {

        }

        protected void ButtonCreate_Click(object sender, EventArgs e)
        {
            if (Page.IsValid)
            {
                Guid id = Guid.NewGuid();
                string imgFile = "noimage";
                string name = RemoveSpecialCharacter(TextBoxName.Text);
                string streetaddress = RemoveSpecialCharacter(TextBoxStreet.Text);
                string country = RemoveSpecialCharacter(TextBoxCountry.Text);
                string zip = RemoveSpecialCharacter(TextBoxPostalCode.Text);
                string city = RemoveSpecialCharacter(TextBoxCity.Text);
                string email = TextBoxEmail.Text;
                string web = RemoveSpecialCharacter(TextBoxWebsite.Text);
                string phone = RemoveSpecialCharacter(TextBoxPhone.Text);
                if (FileUploadPhoto.PostedFile != null &&
FileUploadPhoto.PostedFile.ContentLength > 0)
                {
                    string guid = id.ToString("N");
                    string imageDirectory = Server.MapPath("vcardimage");
                    if (Directory.Exists(imageDirectory))
                    {
                        FileUploadPhoto.PostedFile.SaveAs(
                            Server.MapPath("/vcardimage/" + guid +
FileUploadPhoto.FileName));
                    }
                    else
                    {
                        Directory.CreateDirectory(Server.MapPath("vcardimage"));
FileUploadPhoto.PostedFile.SaveAs(Server.MapPath("/vcardimage/" + guid +
FileUploadPhoto.FileName));
                    }

                    imgFile = imageDirectory + "/" + guid + FileUploadPhoto.FileName;
                    SendVcard("Your Vcard:",
                              "Your vcard is ready. Please open Attachment. \n Thank
you very much \n Semic Provider",
                              email,
                              GenerateVcard(name, streetaddress, zip, city, country,
email, phone, web, imgFile));
                    LabelNotice.Text = "VCard has been sent to " + email;
                }
                else
                {
                    SendVcard("Vcard:", "Your vcard is ready. Please open Attachment.
\n Thank you very much \n Semic Provider",
                              email,
                    GenerateVcard(name, streetaddress, zip, city, country, email,
phone, web, imgFile));
                    LabelNotice.Text = "VCard has been sent to " + email;
                }
                InsertTable(name, streetaddress, zip, country, phone, email, web,
imgFile);
            }
        }


        /// <summary>
        /// Generates the vcard.
        /// </summary>
        /// <param name="firstName">The first name.</param>
        /// <param name="streetAddress">The street address.</param>
        /// <param name="zip">The zip.</param>
        /// <param name="city">The city.</param>
        /// <param name="country">The country.</param>
        /// <param name="email">The email.</param>
        /// <param name="phone">The phone.</param>
        /// <param name="web">The web.</param>
        /// <param name="image">The image.</param>
        /// <returns> vcard string </returns>
        public string GenerateVcard(string firstName, string streetAddress, string
zip, string city, string country, string email, string phone, string web, string
image)
        {
            string url = "";
            if (web.Length > 0)
            {
                url = "http://" + web.Replace("http", "");
            }
            var vcardStringBuilder = new StringBuilder();
            vcardStringBuilder.AppendLine("BEGIN:VCARD");
            vcardStringBuilder.AppendLine("VERSION:2.1");
            vcardStringBuilder.AppendLine("FN;LANGUAGE=en-us:" + firstName);
            vcardStringBuilder.Append("ADR;HOME;PREF:;;");
            vcardStringBuilder.Append(streetAddress + ";");
            vcardStringBuilder.Append(zip + ";");
            vcardStringBuilder.Append(city + ";");
            vcardStringBuilder.AppendLine(country + " ");
            vcardStringBuilder.AppendLine("TEL;HOME;VOICE:" + phone);
            vcardStringBuilder.AppendLine("URL;type=pref:" + url);
            vcardStringBuilder.AppendLine("EMAIL:" + email);
            if (image != "noimage")
            {
                vcardStringBuilder.AppendLine("PHOTO;ENCODING=BASE64;TYPE=" +
image.GetType().ToString().ToUpper() + ":");

vcardStringBuilder.AppendLine(Convert.ToBase64String(ReadImage(image)));
                vcardStringBuilder.AppendLine(string.Empty);
            }
            vcardStringBuilder.AppendLine("END:VCARD");
            return vcardStringBuilder.ToString();
        }

        /// <summary>
        /// Reads the image.
        /// </summary>
        /// <param name="image">The image.</param>
        /// <returns> image bytes </returns>
        public byte[] ReadImage(string image)
        {
            try
            {
                return File.ReadAllBytes(image);
            }
            catch (Exception ex)
            {
                return File.ReadAllBytes(Server.MapPath("media/logo-vcard.png"));
            }
        }

        /// <summary>
        /// Sends the vcard.
        /// </summary>
        /// <param name="subject">The subject.</param>
        /// <param name="body">The body.</param>
        /// <param name="email">The email.</param>
        /// <param name="file">The file.</param>
        private void SendVcard(string subject, string body, string email, string file)
        {
            byte[] byteArray = Encoding.ASCII.GetBytes(file);
            try
            {

                MemoryStream stream = new MemoryStream(byteArray);
                MailMessage message = new MailMessage();
                message.Subject = subject;
                message.Body = body;
                message.From = new MailAddress("sender id example:
jeevan_2np@yahoo.com");
                message.To.Add(email);
                SmtpClient client = new SmtpClient();
                Attachment vcard = new Attachment(stream, "PersonalVCard.vcf");
                message.Attachments.Add(vcard);
                client.Host = "smtp address";
                client.Port = smtp port number;
                client.Credentials = new NetworkCredential("user id ", "password");
                client.EnableSsl = true;
                client.Send(message);
            }
            catch
            {
                LabelNotice.Text = "Internal Error";
                DownloadVcard(byteArray);
            }

        }

        protected void CustomValidatorName_ServerValidate(object source,
ServerValidateEventArgs args)
        {
            args.IsValid = RemoveSpecialCharacter(args.Value).Length > 0;
        }

        protected void CustomValidatorPostalCode_ServerValidate(object source,
ServerValidateEventArgs args)
        {
            try
            {
                if (args.Value.Length > 0)
                {
                    args.IsValid = RemoveSpecialCharacter(args.Value).Length == 4;
                }
                else
                {
                    args.IsValid = true;
                }
            }
            catch { args.IsValid = false; }
        }

        protected void CustomValidatorCountry_ServerValidate(object source,
ServerValidateEventArgs args)
        {

        }

        protected void CustomValidatorPhone_ServerValidate(object source,
ServerValidateEventArgs args)
        {
            if (args.Value.Length > 0)
            {
                args.IsValid = ValidatePhone(args.Value);
            }
            else
            {
                args.IsValid = true;
            }
        }

        protected void CustomValidatorEmail_ServerValidate(object source,
ServerValidateEventArgs args)
        {
            args.IsValid = ValidateEmail(args.Value.ToString());
        }

        protected void CustomValidatorWebSite_ServerValidate(object source,
ServerValidateEventArgs args)
        {

        }

        /// <summary>
        /// Validates the phone.
        /// </summary>
        /// <param name="input">The input.</param>
        /// <returns> true if it valid </returns>
        public bool ValidatePhone(string input)
        {
            if (input.Length != 8) { return false; }
            Regex num = new Regex("[^0-9]");
            return !num.IsMatch(input);
        }

        /// <summary>
        /// Validates the email.
        /// </summary>
        /// <param name="email">The email.</param>
        /// <returns> true on valid email </returns>
        public bool ValidateEmail(string email)
        {
            return Regex.IsMatch(email,
             @"^(?("")("".+?""@)|(([0-9a-zA-Z]((\.(?!\.))|[-
!#\$%&'\*\+/=\?\^`\{\}\|~\w])*)(?<=[0-9a-zA-Z])@))" +
             @"(?(\[)(\[(\d{1,3}\.){3}\d{1,3}\])|(([0-9a-zA-Z][-\w]*[0-9a-zA-Z]\.)+[a-
zA-Z]{2,6}))$");
        }

        /// <summary>
        /// Removes the special character.
        /// </summary>
        /// <param name="input">The input.</param>
        /// <returns> secure string </returns>
        public string RemoveSpecialCharacter(string input)
        {
            return Regex.Replace(input, "[^0-9a-zA-Z. ]+", "");
        }

        protected void TextBoxEmail_TextChanged(object sender, EventArgs e)
        {

        }

        /// <summary>
        /// Creates the table.
        /// </summary>
        /// <param name="tableName">Name of the table.</param>
        /// <returns> true if table created</returns>
        private static bool CreateTable(string tableName)
        {
            try
            {
                SqlConnection connection = new SqlConnection(GlobalSettings.DbDSN);
                string queryString =
                    @" CREATE TABLE [vcardimage](
                        [id] [int] IDENTITY(1,1) NOT NULL,
                         [name] [nvarchar] (250) NULL,
                        [street] [nvarchar](250) NULL,
                        [postalcode] [int] NULL,
                        [country] [nvarchar](250) NULL,
                        [phone] [nvarchar](250) NULL,
                        [email] [nvarchar](250) NULL,
                        [website] [nvarchar](250) NULL,
                        [date] [datetime] NULL,
                        [imageguid] [nvarchar](250) NULL)";

                SqlCommand command = new SqlCommand(queryString, connection);
                connection.Open();
                command.ExecuteNonQuery();
                connection.Close();
                return true;
            }
            catch
            {
                return false;
            }
        }

        /// <summary>
        /// Checks the table.
        /// </summary>
        /// <param name="tableName">Name of the table.</param>
        /// <returns> true if table exist </returns>
        private static bool CheckTable(string tableName)
        {
            SqlServerHelper sqlHeplper = new SqlServerHelper(GlobalSettings.DbDSN);
            return sqlHeplper.Utility.CreateTableUtility().ContainsTable(tableName);
        }

        /// <summary>
        /// Inserts the table.
        /// </summary>
        /// <param name="name">The name.</param>
        /// <param name="street">The street.</param>
        /// <param name="postalcode">The postalcode.</param>
        /// <param name="country">The country.</param>
        /// <param name="phone">The phone.</param>
        /// <param name="email">The email.</param>
        /// <param name="website">The website.</param>
        /// <param name="imageguid">The imageguid.</param>
        /// <returns> true on successul data insert</returns>
        public static bool InsertTable(string name, string street, string postalcode,
string country, string phone, string email, string website, string imageguid)
        {
            string tableName = "vcardimage";
            try
            {
                if (!CheckTable(tableName))
                { CreateTable(tableName); }
                ISqlHelper sqlHelper = umbraco.BusinessLogic.Application.SqlHelper;
                sqlHelper.ExecuteNonQuery(
                @"insert into " + tableName + "(name,street,postalcode,country,phone,
email, website,imageguid, date)" +
                "values(@name, @street, @postalcode, @country, @phone, @email,
@website, @imageguid, @date);",
                sqlHelper.CreateParameter("@name", name),
                sqlHelper.CreateParameter("@street", street),
                sqlHelper.CreateParameter("@postalcode", postalcode),
                sqlHelper.CreateParameter("@country", country),
                sqlHelper.CreateParameter("@phone", phone),
                sqlHelper.CreateParameter("@email", email),
                sqlHelper.CreateParameter("@website", website),
                sqlHelper.CreateParameter("@imageguid", imageguid),
                sqlHelper.CreateParameter("@date", DateTime.Now));
                return true;
            }
            catch (Exception)
            {
                return false;
            }

        }

        /// <summary>
        /// Downloads the vcard.
        /// </summary>
        /// <param name="vcardFile">The vcard file.</param>
        public void DownloadVcard(byte [] vcardFile)
        {
            Response.ClearHeaders();
            Response.Clear();
            Response.Buffer = true;
            Response.ContentType = "application/vcard";
            Response.AddHeader("content-disposition", "attach;filename =
YourVcard.vcf");
            Response.Charset = "";
            Response.BinaryWrite(vcardFile);
            HttpContext.Current.ApplicationInstance.CompleteRequest();
            Response.End();
        }
    }
}
```

GlobalSetting.xslt

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE xsl:stylesheet [ <!ENTITY nbsp "&#x00A0;"> ]>
<xsl:stylesheet
  version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:msxml="urn:schemas-microsoft-com:xslt"
  xmlns:umbraco.library="urn:umbraco.library"
xmlns:Exslt.ExsltCommon="urn:Exslt.ExsltCommon"
xmlns:Exslt.ExsltDatesAndTimes="urn:Exslt.ExsltDatesAndTimes"
xmlns:Exslt.ExsltMath="urn:Exslt.ExsltMath"
xmlns:Exslt.ExsltRegularExpressions="urn:Exslt.ExsltRegularExpressions"
xmlns:Exslt.ExsltStrings="urn:Exslt.ExsltStrings"
xmlns:Exslt.ExsltSets="urn:Exslt.ExsltSets"
  exclude-result-prefixes="msxml umbraco.library Exslt.ExsltCommon
Exslt.ExsltDatesAndTimes Exslt.ExsltMath Exslt.ExsltRegularExpressions
Exslt.ExsltStrings Exslt.ExsltSets ">

<xsl:output method="xml" omit-xml-declaration="yes"/>
<xsl:param name="currentPage"/>
<xsl:template match="/">
 <!-- start writing XSLT -->
  <xsl:variable name="items" select="$currentPage/ancestor-or-self::*
  /WebSiteGlobalSettings "/>
  <xsl:variable name="logo" select="$items/logo"/>
  <div class="vcard">
    <div class="logo"><a href="/" title="start page">
        <img src="{$logo}"  alt="semic provider logo; visual identity"/>
         </a>
      </div>
      <div class="fn" >
     <xsl:value-of select="$items/siteName"/>
      </div>
      <h4>  <xsl:value-of select="$items/slogan"/> </h4>
    </div>
    </xsl:template>
</xsl:stylesheet>
```

Navigation.xslt

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE xsl:stylesheet [ <!ENTITY nbsp "&#x00A0;"> ]>
<xsl:stylesheet
  version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:msxml="urn:schemas-microsoft-com:xslt"
  xmlns:umbraco.library="urn:umbraco.library"
xmlns:Exslt.ExsltCommon="urn:Exslt.ExsltCommon"
xmlns:Exslt.ExsltDatesAndTimes="urn:Exslt.ExsltDatesAndTimes"
xmlns:Exslt.ExsltMath="urn:Exslt.ExsltMath"
xmlns:Exslt.ExsltRegularExpressions="urn:Exslt.ExsltRegularExpressions"
xmlns:Exslt.ExsltStrings="urn:Exslt.ExsltStrings"
xmlns:Exslt.ExsltSets="urn:Exslt.ExsltSets"
  exclude-result-prefixes="msxml umbraco.library Exslt.ExsltCommon
Exslt.ExsltDatesAndTimes Exslt.ExsltMath Exslt.ExsltRegularExpressions
Exslt.ExsltStrings Exslt.ExsltSets ">
<xsl:output method="xml" omit-xml-declaration="yes"/>
<xsl:param name="currentPage"/>
  <xsl:template match="/">
    <xsl:variable name="items" select="$currentPage/ancestor-or-self::*
/GeneralContent [string(useInNavigation) = '1'] "/>
  <xsl:if test="count($items) &gt; 0">
<ul>
  <xsl:for-each select="$items">
    <xsl:if test="$currentPage/@id = @id">
      <li class="current" >
          <xsl:value-of select="contentTitle"/>
      </li>
    </xsl:if>
     <xsl:if test="$currentPage/@id != @id">
      <li class="{contentTitle}" >
         <a href="{umbraco.library:NiceUrl(@id)}" title="{contentTitle}" >
          <xsl:value-of select="contentTitle"/>
        </a>
      </li>
    </xsl:if>
</xsl:for-each>
</ul>
    </xsl:if>
```

```
</xsl:template>
</xsl:stylesheet>
```

NewsList.xslt

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE xsl:stylesheet [ <!ENTITY nbsp "&#x00A0;"> ]>
<xsl:stylesheet
  version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:msxml="urn:schemas-microsoft-com:xslt"
  xmlns:umbraco.library="urn:umbraco.library"
xmlns:Exslt.ExsltCommon="urn:Exslt.ExsltCommon"
xmlns:Exslt.ExsltDatesAndTimes="urn:Exslt.ExsltDatesAndTimes"
xmlns:Exslt.ExsltMath="urn:Exslt.ExsltMath"
xmlns:Exslt.ExsltRegularExpressions="urn:Exslt.ExsltRegularExpressions"
xmlns:Exslt.ExsltStrings="urn:Exslt.ExsltStrings"
xmlns:Exslt.ExsltSets="urn:Exslt.ExsltSets"
  exclude-result-prefixes="msxml umbraco.library Exslt.ExsltCommon
Exslt.ExsltDatesAndTimes Exslt.ExsltMath Exslt.ExsltRegularExpressions
Exslt.ExsltStrings Exslt.ExsltSets ">
<xsl:output method="xml" omit-xml-declaration="yes"/>
<xsl:param name="currentPage"/>
<xsl:template match="/">
  <xsl:variable name="newsItems" select="$currentPage/ancestor-or-self::*
  /NewsContainer/NewsArticle "/>
<xsl:variable name="numberofnewsitems" select="/macro/numberofnewsitems"/>
  <xsl:if test="count($newsItems) &gt; 0">
        <xsl:for-each select="$newsItems" >
   <xsl:sort select="@createDate" order="ascending"/>
   <xsl:if test="position() &lt;=$numberofnewsitems" >      <div clas="newsItem">
        <h3>
          <a href="{umbraco.library:NiceUrl(@id)}" >
            <xsl:value-of select="newsTitle" />


          </a>
        </h3>
          <p>
          <xsl:value-of select="newsBody" disable-output-escaping="yes" />
        </p>
        <p>
          Last update: <xsl:value-of select="@updateDate" />
        </p>
       </div>
  </xsl:if>
      </xsl:for-each>
  </xsl:if>
</xsl:template>
</xsl:stylesheet>
```

RecentVcard.xslt

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE xsl:stylesheet [ <!ENTITY nbsp "&#x00A0;"> ]>
<xsl:stylesheet
  version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:msxml="urn:schemas-microsoft-com:xslt"
  xmlns:recentVcard="urn:recentvcard"
  xmlns:umbraco.library="urn:umbraco.library"
xmlns:Exslt.ExsltCommon="urn:Exslt.ExsltCommon"
xmlns:Exslt.ExsltDatesAndTimes="urn:Exslt.ExsltDatesAndTimes"
xmlns:Exslt.ExsltMath="urn:Exslt.ExsltMath"
xmlns:Exslt.ExsltRegularExpressions="urn:Exslt.ExsltRegularExpressions"
xmlns:Exslt.ExsltStrings="urn:Exslt.ExsltStrings"
xmlns:Exslt.ExsltSets="urn:Exslt.ExsltSets" xmlns:recentvcard="urn:recentvcard"
  exclude-result-prefixes="msxml umbraco.library Exslt.ExsltCommon
Exslt.ExsltDatesAndTimes Exslt.ExsltMath Exslt.ExsltRegularExpressions
Exslt.ExsltStrings Exslt.ExsltSets recentvcard ">
<xsl:output method="xml" omit-xml-declaration="yes"/>
<xsl:param name="currentPage"/>
<xsl:template match="/">
<!-- start writing XSLT -->
  <xsl:value-of select="recentVcard:GetVcardData()" disable-output-escaping="yes"/>
</xsl:template>
</xsl:stylesheet>
```

RecentVcard.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using umbraco.DataLayer;
using umbraco;
namespace VCardGenerator
{
    public class RecentVcard
    {
        public static string GetVcardData()
        {
            ISqlHelper getVcard =
DataLayerHelper.CreateSqlHelper(GlobalSettings.DbDSN);
            IRecordsReader read = getVcard.ExecuteReader(@"select top 10 id, name,
country
            from vcardimage where id > @id order by id desc",
getVcard.CreateParameter("@id", "0"));
            string data = "";
            while (read.Read())
            {
                data += "<li><span class=\"vcard\"><span class=\"fn\">" +
read.GetString("name") +
                    "</span><span class=\"add\"> " + read.GetString("country") +
                    " </span></span></li>";
            }
            return "<ul>" + data + "</ul>";
        }
    }
}
```

About Writer

Name: Jivan Thapa

Details: I am an international student at Erhvervsakademiet København Nord Lyngby Denmark, studying in Bsc. Web Development. “How to Umbraco” basic back is a part of final Examination Projects. My interest is in IT security and Object oriented program.

Web: www.jeevanthapa.com

Email: jeevan_2np@yahoo.com

“How to Umbraco” basic book is design for new asp.net developer who has knowledge of .net programming language and looking to build website using Umbraco CMS. Step by step and explanation through examples to understand core concept of Umbraco CMS is valuable assets of this book. “How to Umbraco” basic book is probably first written books to provide concept of Umbraco CMS and its structure. In General using this book new developer gets knowledge of Building website using Umbraco site from Scratch.

Here I covered:

- Document Type
- Template
- Content
- Xslt
- Extending Umbraco using .net class and User Control
- And more ….

Jivan Thapa