



# ASP.NET 3.5

## Website Programming

### Problem – Design – Solution

Chris Love



Programmer to Programmer™

# Get more out of **WROX.com**

## **Interact**

---

Take an active role online by participating in our P2P forums

## **Wrox Online Library**

---

Hundreds of our books are available online through [Books24x7.com](http://Books24x7.com)

## **Wrox Blox**

---

Download short informational pieces and code to keep you up to date and out of trouble!

## **Chapters on Demand**

---

Purchase individual book chapters in pdf format

## **Join the Community**

---

Sign up for our free monthly newsletter at [newsletter.wrox.com](http://newsletter.wrox.com)

## **Browse**

---

Ready for more Wrox? We have books and e-books available on .NET, SQL Server, Java, XML, Visual Basic, C#/ C++, and much more!

## **Contact Us.**

We always like to get feedback from our readers. Have a book idea?  
Need community support? E-mail us at [\*\*wrox-partnerwithus@wrox.com\*\*](mailto:wrox-partnerwithus@wrox.com)

# ASP.NET 3.5 Website Programming

<b>Introduction</b> . . . . .	<b>xix</b>
<b>Chapter 1: Introducing the Project: TheBeerHouse</b> . . . . .	<b>1</b>
<b>Chapter 2: Developing the Site Design</b> . . . . .	<b>9</b>
<b>Chapter 3: Planning an Architecture</b> . . . . .	<b>59</b>
<b>Chapter 4: Membership and User Profiling</b> . . . . .	<b>143</b>
<b>Chapter 5: News and Article Management</b> . . . . .	<b>247</b>
<b>Chapter 6: Opinion Polls</b> . . . . .	<b>327</b>
<b>Chapter 7: Newsletters</b> . . . . .	<b>365</b>
<b>Chapter 8: Forums</b> . . . . .	<b>397</b>
<b>Chapter 9: E-Commerce Store</b> . . . . .	<b>431</b>
<b>Chapter 10: Calendar of Events</b> . . . . .	<b>501</b>
<b>Chapter 11: Photo Gallery</b> . . . . .	<b>519</b>
<b>Chapter 12: Localizing the Site</b> . . . . .	<b>545</b>
<b>Chapter 13: Deploying the Site</b> . . . . .	<b>563</b>
<b>Index</b> . . . . .	<b>591</b>



# **ASP.NET 3.5 Website Programming**

**Problem–Design–Solution**



# **ASP.NET 3.5 Website Programming**

## **Problem–Design–Solution**

Chris Love



WILEY

Wiley Publishing, Inc.

# ASP.NET 3.5 Website Programming: Problem–Design–Solution

Published by  
Wiley Publishing, Inc.  
10475 Crosspoint Boulevard  
Indianapolis, IN 46256  
[www.wiley.com](http://www.wiley.com)

Copyright © 2010 by Wiley Publishing, Inc., Indianapolis, Indiana

Published simultaneously in Canada

ISBN: 978-0-470-18758-6

Manufactured in the United States of America

10 9 8 7 6 5 4 3 2 1

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, except as permitted under Sections 107 or 108 of the 1976 United States Copyright Act, without either the prior written permission of the Publisher, or authorization through payment of the appropriate per-copy fee to the Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923, (978) 750-8400, fax (978) 646-8600. Requests to the Publisher for permission should be addressed to the Permissions Department, John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, (201) 748-6011, fax (201) 748-6008, or online at <http://www.wiley.com/go/permissions>.

**Limit of Liability/Disclaimer of Warranty:** The publisher and the author make no representations or warranties with respect to the accuracy or completeness of the contents of this work and specifically disclaim all warranties, including without limitation warranties of fitness for a particular purpose. No warranty may be created or extended by sales or promotional materials. The advice and strategies contained herein may not be suitable for every situation. This work is sold with the understanding that the publisher is not engaged in rendering legal, accounting, or other professional services. If professional assistance is required, the services of a competent professional person should be sought. Neither the publisher nor the author shall be liable for damages arising herefrom. The fact that an organization or Web site is referred to in this work as a citation and/or a potential source of further information does not mean that the author or the publisher endorses the information the organization or Web site may provide or recommendations it may make. Further, readers should be aware that Internet Web sites listed in this work may have changed or disappeared between when this work was written and when it is read.

For general information on our other products and services please contact our Customer Care Department within the United States at (877) 762-2974, outside the United States at (317) 572-3993 or fax (317) 572-4002.

Wiley also publishes its books in a variety of electronic formats. Some content that appears in print may not be available in electronic books.

**Library of Congress Control Number:** 2009928740

**Trademarks:** Wiley, the Wiley logo, Wrox, the Wrox logo, and Wrox Programmer to Programmer are trademarks or registered trademarks of John Wiley & Sons, Inc. and/or its affiliates, in the United States and other countries, and may not be used without written permission. All other trademarks are the property of their respective owners. Wiley Publishing, Inc. is not associated with any product or vendor mentioned in this book.



## About the Author



**Chris Love** has more than 16 years experience in software design, development, and architecture. He has been the principal developer for more than 300 small- and medium-sized ASP and ASP.NET websites over the past 10 years. These projects have exposed him to a wide range of Microsoft-related technologies to solve real business problems for his clients. He has focused primarily on ASP.NET and VB.NET/C# to produce the majority of his websites. Recently he has been working on applying JQuery to bring more client-side life to his websites, and pursuing a reduction in the size and complexity of his ASP.NET applications by eliminating Web Controls and focusing more on pure HTML markup instead.

Chris's clients rely on his experience and expertise to develop online marketing strategies, including search engine optimization and pay-per-click campaigns. He has begun to leverage this experience along with his ASP.NET know-how to build his own Web properties and practice his technical and marketing theories first-hand.

Chris has been in a leadership role in the local user's group TRINUG for more than 5 years. He frequently presents and organizes Code Camps around the country.

His first book, *ASP.NET 2.0 Your Visual Blueprint for Developing Web Applications* (Wiley), was published in 2007. He has also written two eBooks for WROX on custom HttpHandlers and HttpModules.



# Credits

**Acquisitions Editor**

Paul Reese

**Project Editor**

Maryann Steinhart

**Technical Editor**

Cody Reichenau

**Production Editor**

Daniel Scribner

**Copy Editor**

Foxxe Editorial Services

**Editorial Director**

Robyn B. Siesky

**Editorial Manager**

Mary Beth Wakefield

**Production Manager**

Tim Tate

**Vice President and  
Executive Group Publisher**

Richard Swadley

**Vice President and Executive Publisher**

Barry Pruett

**Associate Publisher**

Jim Minatel

**Project Coordinator, Cover**

Lynsey Stanford

**Composer**

Craig Johnson, Happenstance Type-O-Rama

**Proofreader**

Nate Pritts, Word One

**Indexer**

Robert Swanson

**Cover Image**

© Getty Images



# Acknowledgments

I am so fortunate to have a great collection of friends and family who have been very supportive over the years. As my .NET experience and knowledge have matured I have been blessed with forming some great relationships in the community. I have to thank everyone who has been supportive in the Triangle .Net User Group over the years. Deciding one night what the group was ultimately changed my career so much for the positive. I encourage each and every reader of this book to seek out any user group in his area.

With that said, there are some specific people to thank. First, Tom, who has been a great friend to me both professionally and personally, and who is always excited when good things happen for me. Beth, you really helped me keep my composure and perspective in the past year with so many things coming at me.

I want to thank all the Microsoft team members with whom I have formed relationships over the last few years. You all have been so helpful to me and anyone else who asked.

Finally, I thank the staff at Wiley for offering me this opportunity. Nothing of this magnitude is without challenges and they have been helpful in solving the many challenges that arose. Producing a book is no small task, but ultimately rewarding. It is always an honor to have the opportunity to share knowledge and experience about something you love with others on the scale possible with a book like this. I know I have learned a lot and I hope to keep learning and sharing.

— *Chris Love*

The publisher gratefully acknowledges **Marco Bellinaso**'s contributions to this book. Marco was the author of *ASP.NET 2.0 Website Programming Problem–Design–Solution* as well as the original TheBeerHouse application. This ASP.NET 3.5 version of the book relies heavily on Marco's book and his application.



# Contents

<b>Introduction</b>	<b>xix</b>
<b>Chapter 1: Introducing the Project: TheBeerHouse</b>	<b>1</b>
<b>Problem</b>	<b>2</b>
<b>Design</b>	<b>2</b>
<b>Solution</b>	<b>4</b>
<b>Summary</b>	<b>6</b>
<b>Chapter 2: Developing the Site Design</b>	<b>9</b>
<b>Problem</b>	<b>9</b>
<b>Design</b>	<b>12</b>
Designing the Site Layout	13
Sharing the Common Design among Multiple Pages	19
Creating a Set of User-selectable Themes	26
Creating a Navigation System	28
Creating XHTML-Compliant and -Accessible Sites	29
Sharing a Common Behavior among All Pages	30
<b>Solution</b>	<b>32</b>
Creating the Site Design	33
Creating the Master Page	34
Binding a BreadCrumb to the SiteMap	45
Creating the First Theme	46
Creating a Sample Default.aspx Page	46
Creating the Second Theme	48
Creating the ThemeSelector User Control	49
Creating Consistent Sidebar Content	55
Another Small Touch of Style	56
<b>Summary</b>	<b>57</b>
<b>Chapter 3: Planning an Architecture</b>	<b>59</b>
<b>Problem</b>	<b>59</b>
<b>Design</b>	<b>60</b>
Creating a Common Class Library	60

# Contents

---

Adding a Class Library to a Visual Studio Solution	61
Designing a Layered Infrastructure	63
Choosing a Data Store	64
Designing the Data Access Layer	65
Designing the Business Logic Layer	87
Caching Data for Better Performance	100
Choosing a Caching Strategy That Fits Your Needs	101
Transaction Management with ADO.NET Entity Framework	102
Storing Connection Strings and Other Settings	102
User Interface	105
Error Logging	118
Search Engine Optimization Techniques	119
Navigation	123
<b>Solution</b>	<b>130</b>
TheBeerHouse Configuration Section	130
Implementing Good Search Engine Optimization Techniques	134
Configuring ELMAH	139
<b>Summary</b>	<b>142</b>
<b>Chapter 4: Membership and User Profiling</b>	<b>143</b>
<b>Problem</b>	<b>144</b>
<b>Design</b>	<b>145</b>
Password Storage Mechanisms	146
Authentication Modes: Windows Security or Custom Login Form?	146
The Let's Do Everything on Our Own Approach	147
The Membership and MembershipUser Classes	148
Setting Up and Using Roles	167
Setting Up and Using User Profiles	172
OpenId Identity Services	178
Designing Our Solution	179
<b>Solution</b>	<b>180</b>
The Configuration File	181
Creating the Login Box	184
The AccessDenied.aspx Page	186
Implementing OpenId Authentication	190
The UserProfile Control	193
The Register Page	203
The PasswordRecovery Page	208
The EditProfile Page	210
Creating an AJAX Login Dialog	214



---

Persisting the Favorite Theme between Sessions	220
The Administration Section	221
<b>Summary</b>	<b>246</b>
<b>Chapter 5: News and Article Management</b>	<b>247</b>
<hr/>	
<b>Problem</b>	<b>247</b>
<b>Design</b>	<b>249</b>
Features to Implement	249
Designing the Database Tables	251
Creating the Entity Model	256
Building Repository Classes	258
Designing the Configuration Module	266
Designing the User Interface	267
The Need for Security	272
<b>Solution</b>	<b>273</b>
The Database Solution	273
Implementing the Configuration Module	274
Implementing the Business Logic Layer	279
Implementing Gravatars	293
Implementing the User Interface	293
<b>Summary</b>	<b>326</b>
<b>Chapter 6: Opinion Polls</b>	<b>327</b>
<hr/>	
<b>Problem</b>	<b>327</b>
<b>Design</b>	<b>329</b>
Handling Multiple Votes	330
Designing the Database Tables	331
Designing the Configuration Module	332
Creating the Entity Data Model	332
Designing the Business Layer	333
Designing the User Interface Services	335
<b>Solution</b>	<b>336</b>
Working on the Database	336
Implementing the Configuration Module	337
Implementing the Repositories	340
Extending the Entity Model Entities	340
Implementing the User Interface	342
<b>Summary</b>	<b>363</b>

# Contents

---

## **Chapter 7: Newsletters** **365**

---

<b>Problem</b>	<b>365</b>
<b>Design</b>	<b>366</b>
Creating and Sending E-mails	366
Managing Long Operations on the Server	367
Designing the Database Tables	371
Designing the Configuration Module	372
Designing the User Interface Services	373
<b>Solution</b>	<b>374</b>
Implementing the Configuration Module	374
Implementing the Data Access Layer	377
Implementing the Business Logic Layer	377
Implementing the User Interface	384
<b>Summary</b>	<b>394</b>

## **Chapter 8: Forums** **397**

---

<b>Problem</b>	<b>397</b>
<b>Design</b>	<b>398</b>
Designing the Database Tables	399
Designing the Configuration Module	401
Designing the Business Layer	402
Designing the User Interface Services	402
<b>Solution</b>	<b>403</b>
Implementing the Database	403
Implementing the Data Access Layer	405
Implementing the Business Logic Layer	407
Implementing the User Interface	407
<b>Summary</b>	<b>429</b>

## **Chapter 9: E-Commerce Store** **431**

---

<b>Problem</b>	<b>431</b>
<b>Design</b>	<b>432</b>
Choosing an Online Payment Solution	434
Designing the Database Tables	441
Designing the Configuration Module	442
Designing the Entity Model	443
Designing the Business Layer	444
Designing the User Interface Services	446

---

<b>Solution</b>	<b>448</b>
Implementing the Business Logic Layer	448
Implementing the User Interface	455
Creating a Policies and Procedures Page	498
<b>Summary</b>	<b>499</b>
<b>Chapter 10: Calendar of Events</b>	<b>501</b>
<hr/>	
<b>Problem</b>	<b>501</b>
<b>Design</b>	<b>502</b>
Designing the Database Tables	502
Creating the Entity Data Model	502
Designing the Business Layer	503
Designing the User Interface Services	505
<b>Solution</b>	<b>510</b>
Implementing the Repositories	510
Extending the Entity Model Entities	511
Implementing the User Interface	512
<b>Summary</b>	<b>518</b>
<b>Chapter 11: Photo Gallery</b>	<b>519</b>
<hr/>	
<b>Problem</b>	<b>519</b>
<b>Design</b>	<b>519</b>
Designing the Database Tables	520
Creating the Entity Data Model	520
Designing the Business Layer	521
Designing the Configuration Module	523
The Photo Gallery Storage	524
Designing the User Interface Services	525
<b>Solution</b>	<b>526</b>
Implementing the Repositories	526
Extending the Entity Model Entities	526
Implementing the User Interface	533
<b>Summary</b>	<b>543</b>
<b>Chapter 12: Localizing the Site</b>	<b>545</b>
<hr/>	
<b>Problem</b>	<b>545</b>
<b>Design</b>	<b>546</b>
Localization Features of ASP.NET	547

# Contents

---

<b>Solution</b>	<b>556</b>
Localizing ASP.NET AJAX	560
<b>Summary</b>	<b>562</b>
<b>Chapter 13: Deploying the Site</b>	<b>563</b>
<b>Problem</b>	<b>563</b>
<b>Design</b>	<b>564</b>
Deploying the Database to SQL Server 2008	565
Deploying the Site	576
<b>Solution</b>	<b>590</b>
<b>Summary</b>	<b>590</b>
<b>Index</b>	<b>591</b>

# Introduction

Dear reader, thanks for picking up this book, and welcome to the *ASP.NET 3.5 Website Programming: Problem–Design–Solution*, fully updated to ASP.NET version 3.5! The idea for this book was born in 2001, with ASP.NET 1.0, from the desire to have a book that teaches how to create real-world web sites. The first edition was published in 2002, and fortunately it was a success. I believe that this was due to the fact that most ASP.NET books on the market were (and still are) reference-type books, which describe every single control of the framework, and all their methods and properties, but the examples they provide are single-page demos showing how to use a control of a feature. Typically these references don't show how to integrate all ASP.NET features and controls into a single site with rich functionality, which is what readers have to do at work. Designing and implementing a real-world site is very different from creating simple examples, and that's why I think a book like this is helpful for developers facing real problems in their everyday work.

Much of this new edition was rewritten completely from scratch, while trying to preserve as much of the existing application integrity as possible. As a new author for this book I tried to find a balance between the existing Beer House application framework, add some new modules, and integrate some of the great new technologies in ASP.NET 3.5 SP1.

This book is aimed at describing, designing, and implementing a site much like the ones you're probably working on or will be soon, while taking the opportunity to introduce and explain many of the new features that the new great ASP.NET 3.5 Framework offers. I don't hide difficult problems so that the solution can be simpler and shorter to develop; rather, I try to explain most of the problems you'll typically face when writing a modern web site, and provide one or more solutions for them. The result is a web site that features a layout with user-selectable themes, a membership system, a content management system for publishing and syndicating articles, photos, polls, mailing lists, forums, photo gallery, calendar of events, an e-commerce store with support for real-time credit card processing, home page personalization, search engine optimization and localization (refer to Chapter 1 for a more detailed list of features to be implemented). I also decided to write the book in VB.NET, which differs from the previous editions, but source code is available in both C# and VB.NET. I hope you enjoy reading this book, and that it offers guidance that speeds up the development of your next project and makes it more solid, extensible, and well organized.

You can browse the web site online at [www.thebeerhousebook.com](http://www.thebeerhousebook.com).

The author's blog is available at <http://professionalaspnet.com>. Please keep an eye on it to read about further development and expansion of the sample project.

## Who This Book Is For

Let me state up front that this isn't a book for completely novice programmers, or for experienced developers who have never touched ASP.NET and the .NET Framework in general. This book teaches how to write a real-world web site from scratch to deployment, and as such it can't explain every single detail

of the technology, but must concentrate on designing and writing actual solutions. To get the most from this book, you should already have had some experience with ASP.NET 2.0, even if not advanced solutions. You're not required to know ASP.NET 3.5, as each chapter will introduce the new controls and features that you'll use in that chapter, providing enough background information to implement the solution. If you then want to learn more about a control, you can check the MSDN official documentation or another reference-type book such as Wrox's *Professional ASP.NET 3.5*. A good reference on search engine optimization principles is Wiley's *Search Engine Optimization: Your Visual Blueprint for Effective Internet Marketing* by Kristopher Jones. I also found Wrox's *Professional LINQ* by Scott Klein very helpful.

## What This Book Covers

This book is basically a large case study that starts from the foundation and works its way through to completion with a series of designs and solutions for each incremental step along the way. What sets the Problem–Design–Solution series apart from other Wrox series is the structure of the book and the start-to-finish approach to one completed project. Specifically, this book leads the reader through the development of a complete ASP.NET 3.5 web site that has most of the features users expect to find in a modern content-related and e-commerce site:

- Account registration, personalization, and themes
- Site Navigation
- News and events, organized into categories
- Opinion polls
- Newsletter
- Forums
- Photo Gallery
- Calendar of Events
- E-commerce store with shopping cart and order management
- Localization

From an administrative point of view, the following features and problems are also covered:

- Full online back-end administrative section, to manage practically all data from an intuitive user interface
- Site deployment

The implementation of each of these features provides the opportunity to teach various new features introduced by ASP.NET 3.5, such as the following:

- ASP.NET AJAX
- The new `ListView`
- Entity Framework and LINQ to Entities

Because the book is meant to be a case study of a real-world site, it does not stop at just implementing ASP.NET features; it applies them to typical scenarios. Included are:

- ❑ Search engine optimization techniques
- ❑ Error logging and handling
- ❑ CSS layouts

Not only does this book cover the new features of ASP.NET 3.5, it also demonstrates how to integrate all of them together, for the development of a single full-featured site. All the design options are explained and discussed (including the database design, the data access and business logic components design, and the overall site architecture); at the end of the book you will have learned many of the best practices for web development, based on a solid, scalable, and extensible architecture.

## How This Book Is Structured

The book builds a complete project from start to finish. All the chapters (other than the first one) are self-contained modules within the larger project, and are structured in three sections:

- ❑ **Problem:** This defines the problem or problems to be addressed in the chapter: What do you want to do in this chapter? What features do you want to add to the site and why are they important? What restrictions or other factors need to be taken into account?
- ❑ **Design:** After the problem is defined adequately, this section describes what features are needed to solve the problem. This will give you a broad idea of how the solution will work or what will be entailed in solving the problem.
- ❑ **Solution:** After setting up what you are going to accomplish and why (and how that solves the problem defined earlier), we will produce and discuss the code and any other material that will realize the design and solve the problem laid out at the beginning of the chapter. Just as the coverage of the book as a whole is weighted toward solution, so is each chapter. This is where you will get hands-on practice and create the code.

The book is intended to be read from cover to cover, so that you start with nothing and finish with a complete and deployed web site ready to be launched. However, the book follows a modular structure, so every chapter is quite self-contained and implements a module that, if necessary, can be taken out of the proposed sample project and re-used in some other web site.

## What You Need to Use This Book

To follow the book by building the project on your own computer, or to run the downloadable and ready-to-use project, you'll need the following:

- ❑ Windows XP Professional, Windows Vista, Windows 7, Windows Server 2008, Windows Server 2003, or Windows 2000 Professional or Server.
- ❑ Any edition of Visual Studio 2008, including the freely available Visual Web Developer 2008 Expression Edition. However, Visual Studio 2008 Standard is suggested. You'll be able to follow the book, and run the sample project, even if you don't use a Microsoft editor at all (if, for

## Introduction

---

example, you prefer using Macromedia Dreamweaver MX or some other text editor), because Visual Studio's designers are described and demonstrated in the "Design" section of some chapters, but are not used to write the code in the "Solution" section.

- ❑ The freely available SQL Server 2008 Express Edition, and possibly SQL Server 2008 Standard Edition.

## Conventions

To help you get the most from the text and keep track of what's happening, we've used a number of conventions throughout the book.

**Boxes like this one hold important, not-to-be forgotten information that is directly relevant to the surrounding text.**

*Notes, tips, hints, tricks, and asides to the current discussion are offset and placed in italics like this.*

As for styles in the text:

- ❑ New terms and important words are *highlighted* when introduced.
- ❑ Keyboard combination strokes look like this: Ctrl+A.
- ❑ File names, URLs, and code within the text look like so: `persistence.properties`.
- ❑ Code is presented in two different ways:

We use a monofont type with no highlighting for most code examples.

We use gray highlighting to emphasize code that is of particular importance in the present context, or to indicate where new code is added.

## Source Code

As you work through the examples in this book, you may choose either to type in all the code manually or to use the source code files that accompany the book. All of the source code used in this book is available for download at <http://thebeerhouse.codeplex.com/>. Once at the site, you can either download the current source code by selecting the files under 2.0 Production – WebForms. This site is shared with the MVC edition as well, so be aware of that. You can also download the source code by selecting the Source Code tab and following the instructions on the page.

If for any reason the source is not available at CodePlex, we will keep a copy available for download at [www.wrox.com](http://www.wrox.com). Once at the site, simply locate the book's title (either by using the Search box or by using one of the title lists) and click the Download Code link on the book's detail page to obtain all the source code for the book.

*Because many books have similar titles, you may find it easiest to search by ISBN; this book's ISBN is 978-0-470-18758-6.*



Once you download the code, just decompress it with your favorite compression tool. Alternately, you can go to the main Wrox code download page at <http://www.wrox.com/dynamic/books/download.aspx> to see the code available for this book and all other Wrox books.

## Errata

We make every effort to ensure that there are no errors in the text or in the code. However, no one is perfect, and mistakes do occur. If you find an error in one of our books, like a spelling mistake or faulty piece of code, we would be very grateful for your feedback. By sending in errata you may save another reader hours of frustration and at the same time you will be helping us provide even higher quality information.

To find the errata page for this book, go to [www.wrox.com](http://www.wrox.com) and locate the title using the Search box or one of the title lists. Then, on the book details page, click the Book Errata link. On this page you can view all errata that has been submitted for this book and posted by Wrox editors. A complete book list including links to each book's errata is also available at [www.wrox.com/misc-pages/booklist.shtml](http://www.wrox.com/misc-pages/booklist.shtml).

If you don't spot "your" error on the Book Errata page, go to [www.wrox.com/contact/techsupport.shtml](http://www.wrox.com/contact/techsupport.shtml) and complete the form there to send us the error you have found. We'll check the information and, if appropriate, post a message to the book's errata page and fix the problem in subsequent editions of the book.

## p2p.wrox.com

For author and peer discussion, join the P2P forums at [p2p.wrox.com](http://p2p.wrox.com). The forums are a Web-based system for you to post messages relating to Wrox books and related technologies and interact with other readers and technology users. The forums offer a subscription feature to e-mail you topics of interest of your choosing when new posts are made to the forums. Wrox authors, editors, other industry experts, and your fellow readers are present on these forums.

At <http://p2p.wrox.com> you will find a number of different forums that will help you not only as you read this book, but also as you develop your own applications. To join the forums, just follow these steps:

1. Go to [p2p.wrox.com](http://p2p.wrox.com) and click the Register link.
2. Read the terms of use and click Agree.
3. Complete the required information to join as well as any optional information you wish to provide and click Submit.
4. You will receive an e-mail with information describing how to verify your account and complete the joining process.

*You can read messages in the forums without joining P2P but in order to post your own messages, you must join.*

## Introduction

---

Once you join, you can post new messages and respond to messages other users post. You can read messages at any time on the Web. If you would like to have new messages from a particular forum e-mailed to you, click the Subscribe to this Forum icon by the forum name in the forum listing.

For more information about how to use the Wrox P2P, be sure to read the P2P FAQs for answers to questions about how the forum software works as well as many common questions specific to P2P and Wrox books. To read the FAQs, click the FAQ link on any P2P page.

# 1

## Introducing the Project: TheBeerHouse

This chapter introduces the project that we're going to develop in this book. I'll explain the concept behind the sample website that is the subject of this book, but as you read along you should keep in mind that this is a general-purpose, data-driven, content-based style of website that can easily be modified to meet the needs of a myriad of real-world website requirements. Although we'll use many of the older features of ASP.NET, the clear focus of this book is on showing you how to leverage the powerful new features of ASP.NET 3.5 SP1 such as ASP.NET AJAX, and the ADO.NET Entity Framework. I'll also integrate some basic search engine optimization techniques and third-party services for social networking and SPAM filtering to make a much richer site.

This book follows a "Problem-Design-Solution" approach in each chapter: the Problem section explains the business requirements for the module designed in that chapter; the Design section is used to develop the roadmap for meeting those requirements, and the Solution section is where we write the code to implement our design. This is unlike traditional computer books because the focus is not on teaching basic concepts but rather on showing you how to apply your knowledge to solve real-world business problems.

If you are new to ASP.NET, this is perhaps not the best book to start with, but if you're generally familiar with the basic concepts of web development and ASP.NET (any version of ASP.NET), you're ready to put that knowledge to use, and perhaps you want to learn about the new features in ASP.NET 3.5 SP1. Then, fasten your seat belt!

TheBeerHouse was originally created by Marco to serve as an online presence for a local pub in his hometown of Bologna, Italy. It was meant to serve as a way to reach out to the pub's mostly young clientele. This is still the primary goal of the site, but as technology matures, so does the average user's expectations of a site. The success of the site has driven website traffic and demand for TheBeerHouse merchandise much higher. This demand means a new site is needed to help extend the brand and the sharing of information with patrons. It also provides an opportunity to

# Chapter 1: Introducing the Project: TheBeerHouse

---

upgrade the underlying plumbing with some new technologies for both the user interface and the business layers.

## Problem

Although the owner has always used traditional, printed marketing ads for her pub, and has a popular website, she wants to expand into social networking because many of her patrons are already active on Facebook, Twitter, and other popular social networking sites. Retaining features with which users are familiar is important because users are used to reading about new specials and events, and receiving newsletters with promotions and coupons. Now you want to extend their capability to browse photos of past events and to see a true calendar of upcoming events. Since the typical patron of TheBeerHouse is a young student who is active online, these upgrades should go a long way in extending TheBeerHouse brand, and, hopefully, profits!

The owner also realizes that, just as online marketing and frontend experiences have evolved, so has the technology that drives the Internet. Building a flexible and scalable business tier is also important to the owner. Object Relational Mapping (ORM) frameworks have come of age since the last version of the site. She is also interested in building some client software for back-office management with Windows Presentation Foundation (WPF) and maybe even a mobile client in the next phase, so creating a common business library is very important.

## Design

At the beginning of a project, you think about your client's needs and how you might meet those needs, and possibly even expand on them to give your client more functionality than the minimum needed, while still staying within your time limits and budgetary guidelines. In this scenario your client is a pub owner who wants to have a website to promote her pub, providing online information about upcoming events, reports about past events, and more. The current site can be expanded in many ways, to create a new site that has a lot more interesting things, good for its users (who are also potential customers for the physical pub) and for the store owner. We can begin by writing down a list of features that a modern content-based site should have, and a few reasons why they are useful:

- ❑ The site will need to provide a rich interactive user experience (now known as UX). The experience users have on the website directly affects the impression they have of the pub. This means that users must find information and desired activities easy to browse and interact with.

Attention should also be given to cross-browser compatibility, that is, ensuring that the site looks good and behaves well on different platforms and browsers. While Internet Explorer is still the dominant browser, FireFox, Opera, and Safari are growing in popularity. You can't know in advance which browser your customers will use, as you might in the case of an intranet site for a corporation, for example.

- ❑ A successful content-based site owes its popularity to its users. Loyal users who regularly visit the site, help write content, and participate in polls and special events are those who guarantee that the site will keep growing. To build a vibrant community of active members, users must have some sort of *identity*, something that describes and distinguishes them from other members. Because of this, the site needs a registration feature, as part of a larger authentication/authorization infrastructure. This will also be used to grant and restrict access to some areas of the site.

- ❑ Extending the concept of identity and interaction is leveraging popular social networking sites like Facebook, MySpace, Twitter, and YouTube. These sites will give TheBeerHouse another avenue of extending its brand, interacting with patrons, and allowing them to interact with TheBeerHouse site. Most social networking sites, such as Twitter, have available APIs that can be used by the site to publish content. Twitter allows you to post and monitor your Twitter feed through any client you want. All of these external sites have been designed to add value to the overall experience for a patron of TheBeerHouse.
- ❑ The site needs a constant supply of fresh content to stay alive and vibrant. If the content becomes stale, visitors will lose interest in the site and won't visit it anymore. A pub's site can't be very good unless it has regular updates about upcoming events, parties, and concerts. What's the point in visiting the site if it doesn't display photos that were shot at the last party? To facilitate a constant stream of new content, the site needs some mechanism that enables the editor to easily update it with dynamic content. The content should also be thoughtfully organized so that visitors can find, and even subscribe to, the type(s) of information in which they are most interested. Furthermore, the editor who will be in charge of the content updates will probably not be a technical person, so you must build some simple administration pages that make updates easy, even for nontechnical people.
- ❑ The Beer House owes much of its success to its nightly entertainment, themed parties and other events. The new site includes a Calendar of Events for visitors to keep track of what is happening at the Beer House.
- ❑ Once the site has new content ready to be read, the site's manager must have some way to inform its users about this. Not all users visit the site every day, so the site manager must be proactive and notify the customers about recent updates. If customers have registered on the site, providing their e-mail address, they might also have requested to receive a newsletter notifying them about recent changes and additions to the site. Of course, there are also other ways to syndicate news, such as exposing RSS (Really Simple Syndication) feeds to which a user can register and then control from their favorite RSS reader, and get automatic notifications about news without having to visit the site daily to get the information.
- ❑ A site like this can also be a good opportunity to get feedback from customers about a variety of issues: What do they like most in a pub? What brand of beer do they prefer? Do they want to listen to live music while drinking with friends, or perhaps they don't like all that noise? Establishing some kind of user-to-site communication is important, and if you get a good number of responses, it can even lead to strategic decisions and changes that may improve the business.
- ❑ If the presence of some sort of user-to-site communication is important, user-to-user communication may be even more so, because that's the central point of creating a community of loyal users, who come to the site frequently to chat, discuss the news posted on the site, ask for suggestions from the others about upcoming events, and more. This translates into more traffic on the site and a feeling of membership that will pay off in both the short and long run.
- ❑ Once the store has a user base, the store's owner may decide to expand it so that it supports an online store. In fact, the pub already offers a catalog of products for beer enthusiasts, such as glasses, T-shirts, key chains, and more. If the site has a lot of traffic, it may be a good way to promote these products so that people can place orders without even visiting the pub in person. And once users see a product and like it, they can rate that product to tell other people how much they like it. The online store must be easy to manage by nontechnical people, because it might possibly be the pub's owner who adds and edits products, and manages the orders. Thus, there must be a module with a simple and intuitive UI that automates as many operations as possible, and guides the user through the tasks.

## Chapter 1: Introducing the Project: TheBeerHouse

---

- ❑ Demonstrating how lively and fun TheBeerHouse is very important in fostering dedicated patrons. Photos and videos are a great way to share this atmosphere. Adding a photo gallery to the site and taking advantage of YouTube.com and other video-sharing sites are great ways to share what life is like in TheBeerHouse.
- ❑ With the site offering news and articles, lists of products, user-to-user discussions, and other dynamic content, it's easy to imagine that the home page could easily become crowded, and possibly more difficult to read and understand because of too much information. It would be good if the user herself could build her own home page, according to what she is interested in. Maybe she wants to read about upcoming events but doesn't care about shopping online for gadgets? Great, you want to give her the capability to do that, by adding and deleting content to and from the home page, or maybe just moving around the existing content so that it's placed and organized in a way that she finds more comfortable and useful for her. This type of customization is done on some large sites such as Windows Live and My MSN, for example, and is a great example of personalization, which helps encourage users to decide to register on the site.
- ❑ As mentioned previously, the pub is typically visited by a lot of customers coming from many different countries, and the pub's owner expects the same to happen on the website. Because of this, the site must be partially or fully translated into multiple languages, making it easy for most users to understand it. Not only must text be translated, but information such as dates and numbers should also be displayed according to the user's preferred locale settings, so that nobody will misunderstand an announcement about an upcoming party or event.
- ❑ Optimizing the site for search engine exposure is also a high priority. Adding social networking features to the online strategy should help gain valuable inbound links to assist with search engine ranking. But you still need to make sure that common search engine optimization techniques are applied to the site to help with overall ranking and placement of the site for targeted keyword phrases.

To recap everything in a few words, the TheBeerHouse site will have everything a modern content-based site will have, including dynamic articles and news, polls for user-to-site communication, forums for user-to-user communication, newsletters and RSS feeds to notify members about new content on the site, an e-commerce store for selling products online, home page personalization, and content localization. It will also have a photo gallery, calendar of events, social networking, and OpenId integration. Although the sample project is built around a fictitious pub, you'll recognize in this list of requirements the common features of the majority of content- and commerce-based sites you find online now, and sites that you're likely to develop in the near future, or maybe even sites you're developing right now.

## Solution

The Solution section of each chapter will contain the instructions and actual code for implementing all the features and requirements outlined and designed in the previous sections. However, this first chapter gives you a more detailed description of exactly what the following chapters will cover, so that you can get a good idea of what the final result will be like.

In Chapter 2, you'll build the site's design, the graphics, and the layout that are shared among all pages of the site, through the use of master pages and nested master pages. You will also use themes and Cascading Style Sheets (CSS) to create a couple of different visual appearances for the same master