Karamasoft™

Web: www.karamasoft.com
Email: info@karamasoft.com

19 Cambria
Mission Viejo, CA 92692
Phone: 1-800-401-7543
Fax: 1-509-278-5588

Let UISuite™ Do Your Dirty Work

Scott Mitchell, Software Consultant

Summary: This article provides guidance on how UISuite can make life easier for ASP.NET UI developers. (12 printed pages)

Contents

1. Introduction

When creating software, most developers focus on business logic at the expense of the user interface. This preference toward the business logic stems primarily from the fact that creating a professional-looking user interface is a time consuming and challenging task, especially for those that are not artistically inclined. Whatever the reasons may be, neglecting the user interface puts the project at peril. The user interface is the end user's window into the application and therefore directly impacts its usefulness and utility.

When it comes to user interface design, developers are offered little help from Microsoft. While ASP.NET ships with a number of common user interface elements – a TextBox, a CheckBox, a grid, and a DropDownList, among others – these controls are rudimentary in both their function and form. While ASP.NET's built-in Web controls can be used to create user interface features like search, spell checking, tabs, AJAX, and rich date pickers, doing so requires significant time and resources. Fortunately, with Karamasoft's UISuite it is possible to quickly and easily build professional, aesthetic user interfaces.

UISuite is a collection of ten components that are designed to solve common UI-related challenges. With UISuite it is easy to:
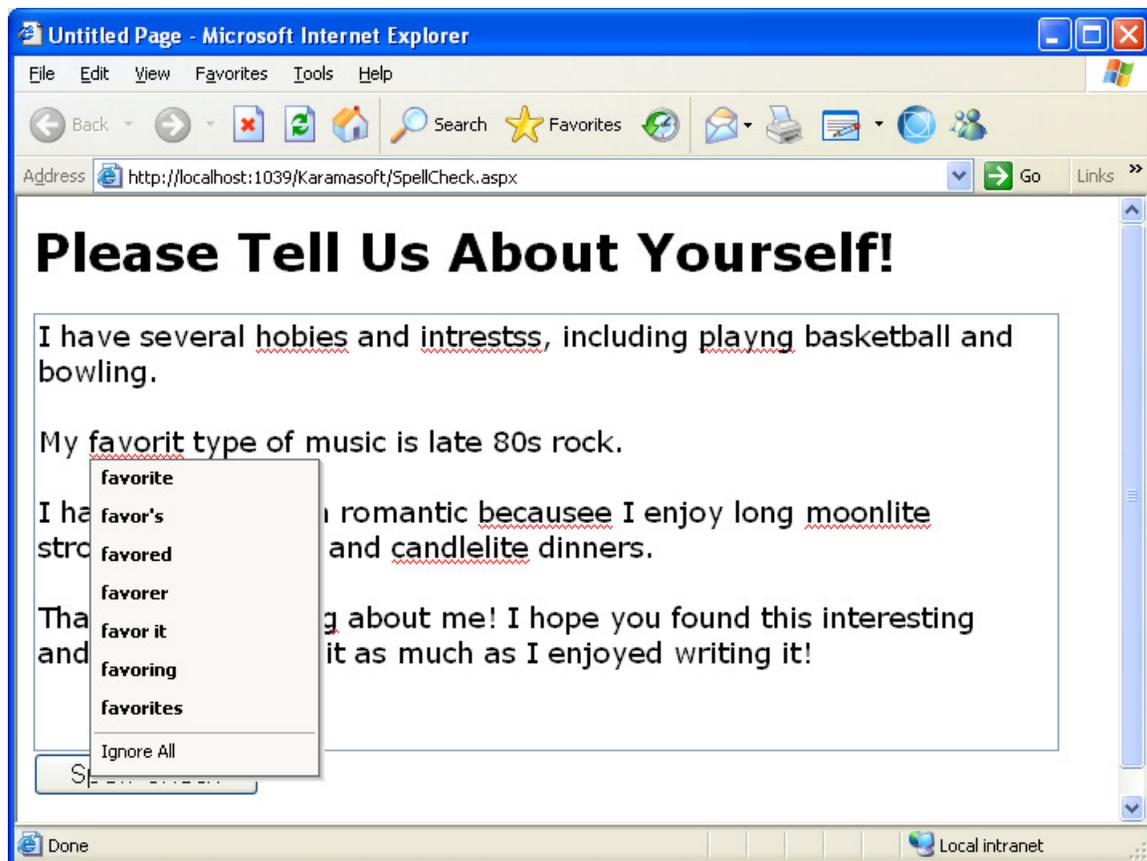
- Spell check user-supplied input
- Offer a rich text editor that allows users to enter formatted and stylized input
- Add search capabilities to your website
- Use popup calendars to collect date inputs

Karamasoft™

Web: www.karamasoft.com
Email: info@karamasoft.com

19 Cambria
Mission Viejo, CA 92692
Phone: 1-800-401-7543
Fax: 1-509-278-5588

- Thoroughly validate user-entered email addresses

- Quickly send personalized emails to users

- Provide a rich navigation user interface

- Improve the responsiveness of your user interface by utilizing AJAX

The remainder of this white paper shows just how easy it is to use the components in UISuite to implement common user interface scenarios.

2. Spell Check User-Supplied Input

Most desktop programs that collect large amounts of text input include an interactive spell checker that highlights misspelled words and offers suggestions for correction. Yet few web applications offer this functionality due to the implementation complexity. With UISuite, however, implementing spell checking is as simple as adding the UltimateSpell control to a page.

Karamasoft™

Web: www.karamasoft.com
Email: info@karamasoft.com

19 Cambria
Mission Viejo, CA 92692
Phone: 1-800-401-7543
Fax: 1-509-278-5588

Once on an ASP.NET page, UltimateSpell renders as a "Spell Check" button that, when clicked, displays a dialog box that lists misspelled words that are found in any textbox, textarea, or other content editable region on the page along with suggestions for replacement. Alternatively, a subset of the user editable regions can be spell checked by using the **ControlIdsToCheck** property.

UltimateSpell can also interactively check spelling as the user types. By setting the **SpellAsYouType** property to True, UltimateSpell will check for spelling mistakes by using AJAX techniques to send text to the server. Misspelled words are then underlined with a red squiggly line; when a user right-clicks on a misspelled word a context menu appears with suggestions for correction.
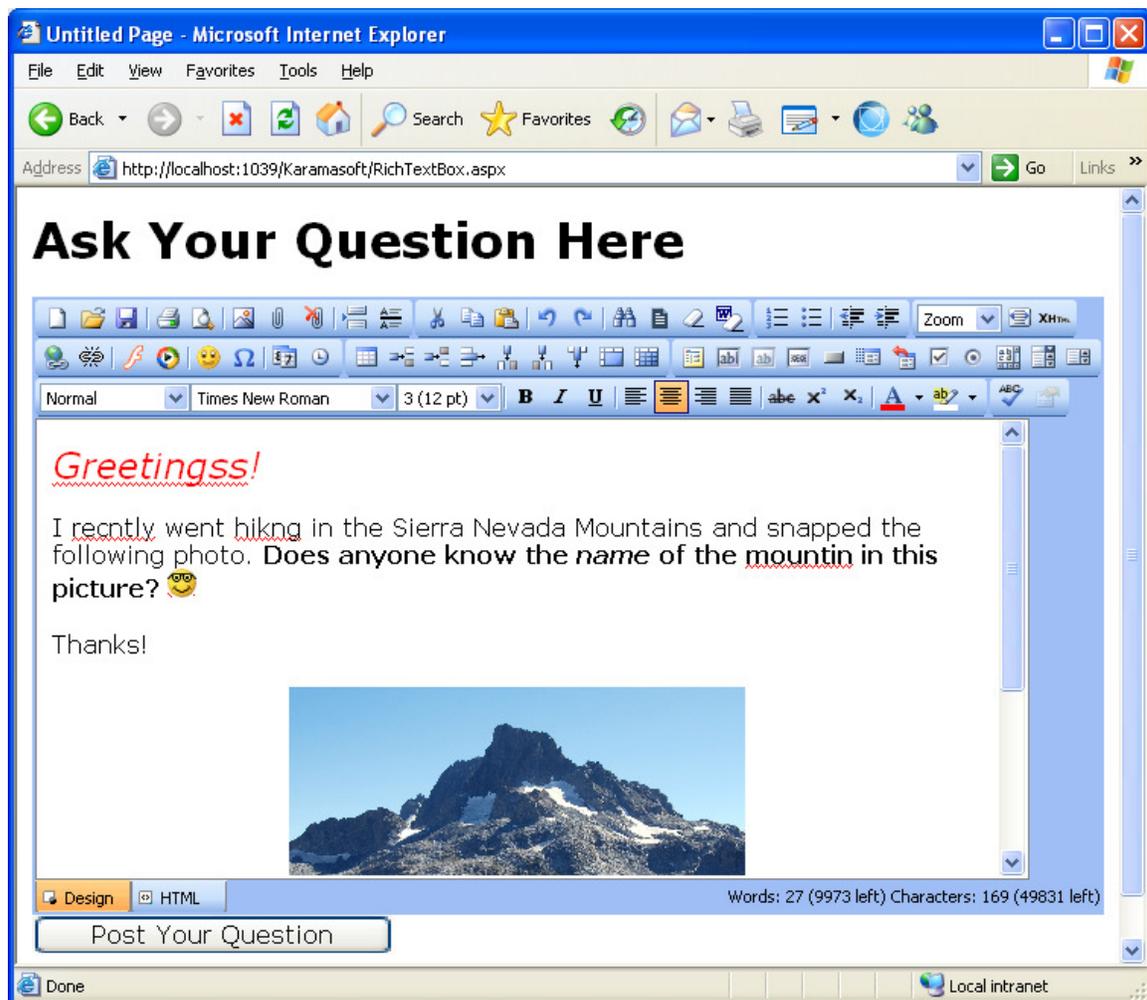
UltimateSpell can be configured to ignore certain types of words, such as HTML tags, email addresses, compound words, and so forth, and these options can be set by the page developer or configured by the end user. The **ShowAddButton** property indicates whether users can add misspelled words to the master dictionary. And UltimateSpell's client- and server-side APIs allow the spell checker to be accessed programmatically.

3. Collect Rich Text Input

Web applications like Content Management Systems (CMS), blogs, and message boards enable visitors to contribute to the website's content. Typically, these sites provide a web page from which the visitor can enter her content, after which the input is saved to a database and is viewable by other visitors to the site. Since this user input is directly displayed on a web page for others to see, the user making the post should be able to indicate the format and layout of the text. The built-in ASP.NET TextBox control, however, can only collect plain text. Therefore, sites that use vanilla textboxes for capturing formatted text input force their users to memorize special characters or tags for formatting their content.

A much more user-friendly approach is to collect text using UISuite's UltimateEditor component. UltimateEditor is a rich textbox control that provides users with a What You See Is What You Get (WYSIWYG) text input experience. With UltimateEditor, users type in their content and format or lay it out by using the component's Toolbar buttons. With the click of the mouse, text can be made bold, italicized, colored, aligned, or turned into a hyperlink. Likewise, numbered and bulleted lists are just a mouse click away; images and emoticons can also be added. And content can be laid out using tables.

In addition to the numerous formatting and layout options, UltimateEditor offers a number of other, more advanced features. For example, users can add attachments and embed Flash and Windows Media Player content. The HTML tab allows advanced users to directly edit the rich textbox's HTML markup. The length of a user's input can be limited by setting the **MaxCharCount** or **MaxWordCount** properties.

Karamasoft™

Web: www.karamasoft.com
Email: info@karamasoft.com

19 Cambria
Mission Viejo, CA 92692
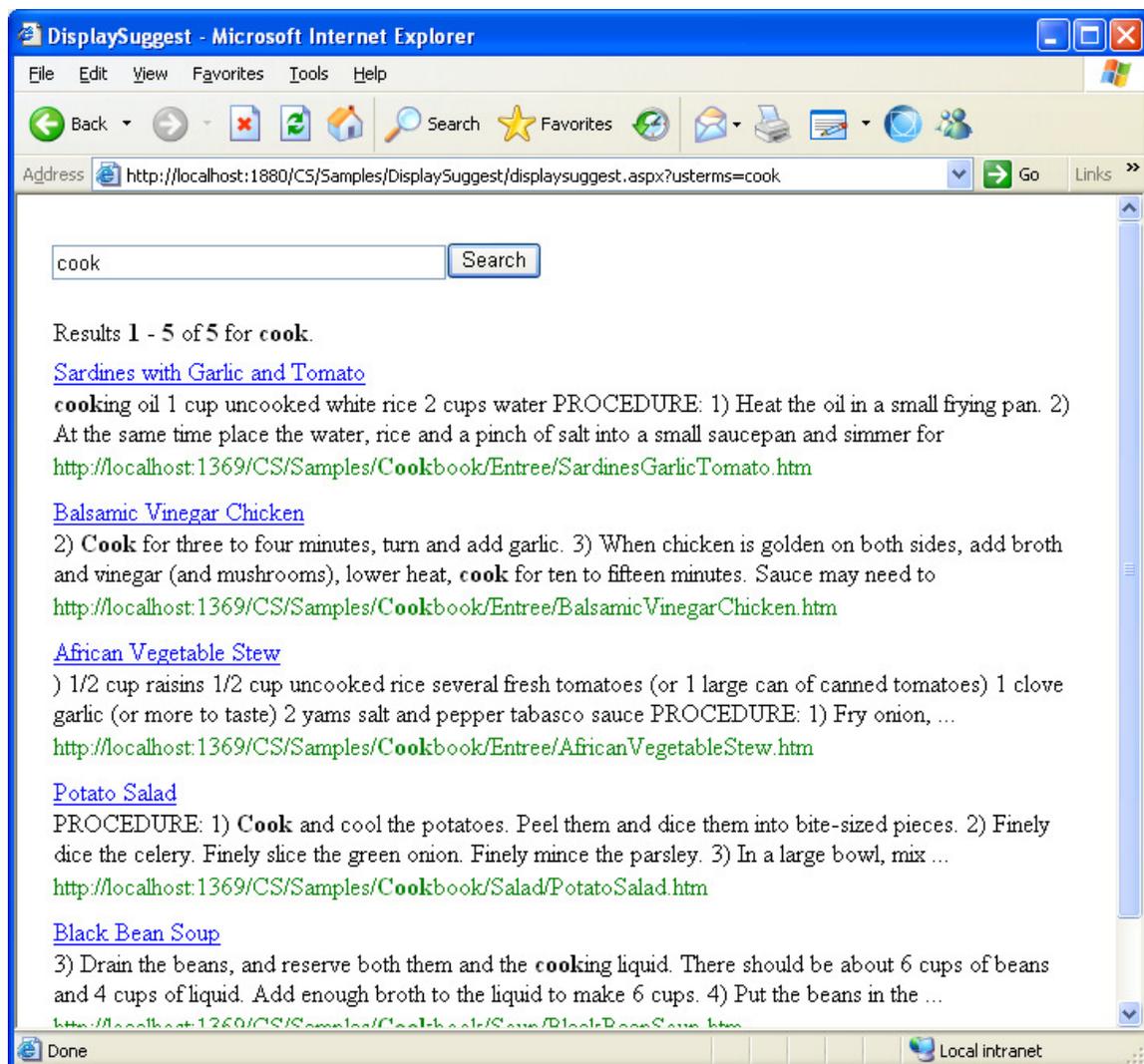Phone: 1-800-401-7543
Fax: 1-509-278-5588

The UltimateEditor user experience can be enhanced with UltimateSpell. The UltimateEditor Toolbar can be configured to include a spell check icon that, when clicked, displays UltimateSpell's spell check dialog box. Additionally, UltimateEditor can be configured to support built-in spell checking, where misspelled words are automatically underlined with a red squiggly and correcting misspellings is as easy as right-clicking on the misspelled word and choosing the correct suggestion.

Best of all, using UltimateEditor is as easy as dragging and dropping. Simply add UltimateEditor to an ASP.NET page and, without having to write a line of code or set a single property, you have a feature-complete rich text editor. And you can fine tune the component's configuration through its properties or from its client- or server-side APIs.

Karamasoft™
Web: www.karamasoft.com
Email: info@karamasoft.com

19 Cambria
Mission Viejo, CA 92692
Phone: 1-800-401-7543
Fax: 1-509-278-5588

4. Help Users Find Content

One challenge every website faces is ensuring that visitors can quickly and accurately find the information they are after. For small websites, this challenge may be met by logically organizing the site's content and providing a navigation user interface. As a site's content grows, however, additional tools and techniques are required to assist users in locating content. One of the most popular and helpful tools is search. Even the most novice users are familiar with the standard search user experience: enter terms or keywords into a textbox, click the Search button, and a list of matching documents appear.

Karamasoft™

Web: www.karamasoft.com
Email: info@karamasoft.com

19 Cambria
Mission Viejo, CA 92692
Phone: 1-800-401-7543
Fax: 1-509-278-5588

While searching a website is easy and straightforward for end users, implementing such functionality from the ground up is a challenging task for web developers. For starters, code must be written to periodically crawl and index the searchable content. More code must be implemented to search this index against a user's query and to display the matching results. Writing and testing this code can take countless hours.

Implementing search is quick and painless with UISuite's UltimateSearch component, which handles both the crawling and indexing of content and rendering of the search user interface. UltimateSearch consults the **UltimateSearch.config** file to determine how to crawl your site.
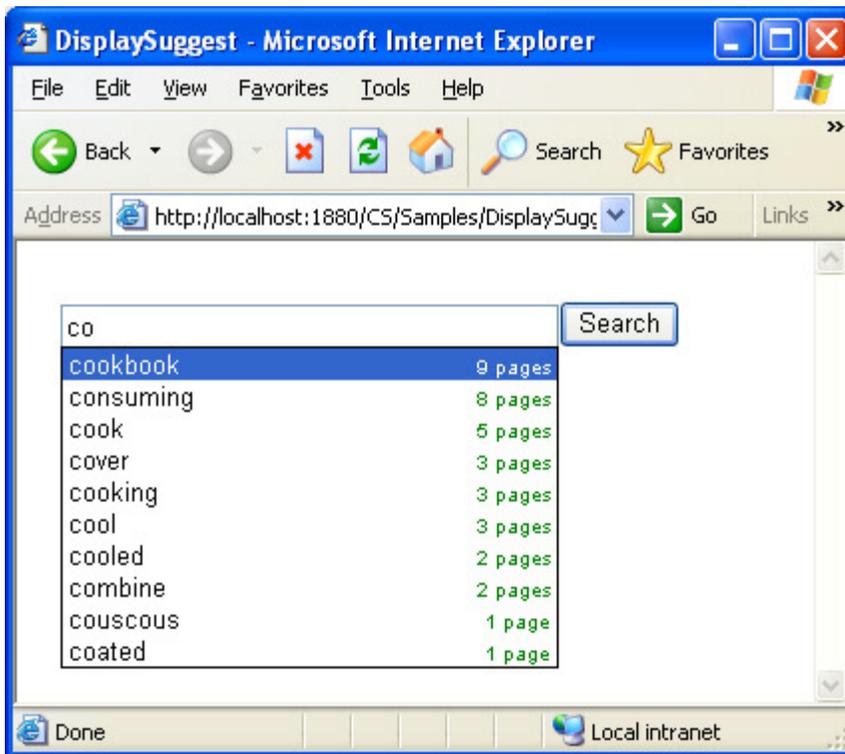
You can configure this XML-formatted file to specify what directories should be crawled, what directories should be ignored, and what file types should be included in the search results (**.aspx**, **.htm**, and so on). In addition to plain text and HTML files, UltimateSearch can also index Microsoft Office and Adobe PDF documents. Moreover, the configuration file also includes a list of common words to exclude from the index, words like "a", "as", and "the".

UltimateSeach's crawler and indexer are launched the first time a user searches the site, and then automatically runs as a background process from then on. You do not need to schedule any sort of task from Windows Scheduler or install any programs on the web server.

In short, you can deploy UltimateSearch by simply copying a few files from your development machine to the production server. This is great news for those using a web hosting company to host their website. UltimateSearch also includes an administration web page that displays the current list of words and pages in the index along with an option to flush and rebuild the index.

UltimateSearch also includes two Web controls: UltimateSearchInput and UltimateSearchOutput. As their names imply, UltimateSearchInput renders a user interface for collecting a user's search terms while UltimateSearchOutput displays the search results.

By default, UltimateSearchInput uses a standard textbox to collect the user's search terms. However, setting the control's **DisplaySuggest** property to True uses AJAX techniques to interactively display search term suggestions based upon the query the user has currently typed into the textbox. The screenshot below illustrates this feature.

Karamasoft™

Web: www.karamasoft.com
Email: info@karamasoft.com

19 Cambria
Mission Viejo, CA 92692
Phone: 1-800-401-7543
Fax: 1-509-278-5588

Adding search to your website does not have to be a difficult or time-consuming process. With UltimateSearch, simply specify crawling and indexing instructions in **UltimateSearch.config** and then add the UltimateSearchInput and UltimateSearchOutput Web controls to an ASP.NET page. It couldn't be any easier!

5. Collect Date Information with Calendars and Date Pickers

Developers relying on ASP.NET's built-in library of Web controls have two options when it comes to prompting users for date values: they can use a TextBox and require that the user type in their date, or they can ask the user to select a date from the Calendar control. Both options have disadvantages. TextBoxes require that the user manually type in the date by hand, making it easy to enter an incorrect or invalid date. The ASP.NET Calendar control is a more straightforward interface, but requires a significant amount of screen real estate, and navigating from one month to another invokes a full postback to the web server.

Many websites have overcome these disadvantages by combining these two interfaces into one. When prompting a user for a date such sites present a textbox into which the user can type the date, if they so choose. Next to the TextBox is a

Karamasoft™

Web: www.karamasoft.com
Email: info@karamasoft.com

19 Cambria
Mission Viejo, CA 92692
Phone: 1-800-401-7543
Fax: 1-509-278-5588

calendar icon that, when clicked, displays a popup calendar from which the user can select a date. Choosing a date from the calendar hides it and displays the selected date in the corresponding textbox. This type of date picking interface is just a drag and a drop away with UISuite's UltimateCalendar and UltimateDatePicker controls.

The UltimateCalendar control is like the built-in ASP.NET Calendar Web control on steroids. In addition to performing the same calendaring tasks that the built-in ASP.NET Calendar does, UltimateCalendar includes a variety of additional properties for enhancing the display and functionality. For example, you can show multiple months at a time via the **MonthCount** property. If you expect users might enter dates far in the past or future, set the **QuickPick** property to True to render the currently displayed month and year as drop-down lists. A user can then quickly jump to a particular month and year by selecting the appropriate values from these lists.

The UltimateDatePicker control displays a TextBox with a calendar icon that, when clicked, pops up an UltimateCalendar. The popped up UltimateCalendar is rendered using client-side script, meaning that no postback is required when navigating from one month to another in the popup calendar. You can easily indicate that certain dates or date spans in the popup calendar should be disabled and therefore not selectable. Best of all, UltimateCalendar includes a rich client-side API that makes it easy to set these properties in response to client-side actions without requiring a roundtrip to the web server.

The following screenshot shows the UltimateCalendar and UltimateDatePicker controls in action in a fictional airline reservation website. On the right is an UltimateCalendar control that shows the special fares for the current month. These specials are set in the ASP.NET page's code section via the UltimateCalendar control's **SpecialDates** collection. Each special date includes a start and end date, the HTML to display, how the table cell(s) should be styled, and whether the special date is a recurring date or not. On the left are two UltimateDatePicker controls; each UltimateDatePicker is assigned to an additional UltimateCalendar control on the page (which is not displayed until the corresponding textbox's calendar icon is clicked).
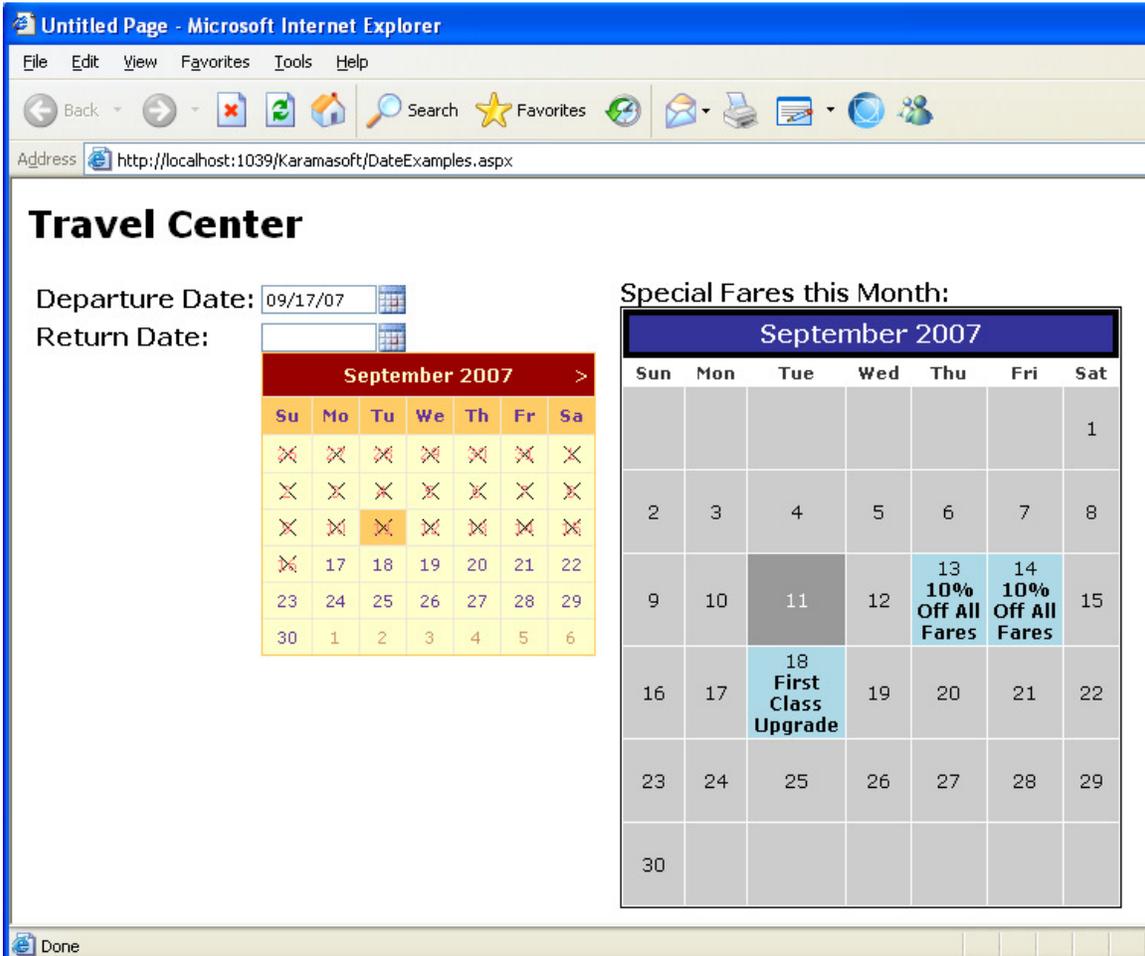
The UltimateCalendar for the Departure Date UltimateDatePicker has its **MinDate** property set to the current date, thereby prohibiting a user from selecting a date in the past. Once a date is selected from the Departure Date UltimateCalendar, the client-side API is used to set the **MinDate** property of the Return Date UltimateCalendar to the selected Departure Date. As the screenshot below shows, all Return Dates preceding the Departure Date (September 17th) are crossed out and are not selectable.

6. Validate Email Addresses and Personalize Email Communications

Despite alternative communication modes like instant messaging, chat rooms, and text messages, email remains the communication mode of choice for most websites for a variety of reasons. Two prime reasons are that anyone with access to the Internet has or can freely obtain an email address and that the cost of sending an

Karamasoft™

Web: www.karamasoft.com
Email: info@karamasoft.com

19 Cambria
Mission Viejo, CA 92692
Phone: 1-800-401-7543
Fax: 1-509-278-5588

email is virtually non-existent. Furthermore, many websites already have a rich email database, and sending an email from an ASP.NET web application can be accomplished in a few short lines of code.

Sending out an email blast is simple enough, but effectively marketing via email is much harder. For starters, there's a high likelihood that many of the email addresses you have in your database are dead ones. Perhaps a user entered a bogus email address when creating an account, or maybe the email address itself no longer exists. Couple that with the fact that due to the abundance of email and the deluge of spam, many people are quick to delete email messages that either don't catch their eye or resonate with them.

UISuite offers two components to assist with improving the effectiveness of email campaigns, UltimateEmailValidator and UltimateEmailMerge. UltimateEmailValidator provides four different techniques for verifying email addresses:

- **A syntax check** – ensures that the email address conforms to the expected pattern, namely some characters followed by "@", followed by more characters that include at least one period, followed by one or more characters.

- **A domain check** – ensures that the domain portion of the supplied email address points to a registered mail server. That is, when validating *accountName@domain.com*, this check ensures that *domain.com* is configured to support email accounts.

- **An SMTP check** – ensures that the SMTP server used by *domain.com* is up and running.

- **An account check** – ensures that the mail server includes an account with the specified account name. For example, when validating *accountName@domain.com*, this check ensures that the account *accountName* exists on the *domain.com* mail server.

You can specify what checks should be applied using the **MaxValidationLevel** property. By selecting a particular validation level, all of the less stringent checks are performed as well. That is, if you set **MaxValidationLevel** to perform an SMTP check, the syntax and domain checks occur as well.

UltimateEmailMerge simplifies the process of sending personalized emails to a list of recipients. Start by adding UltimateEmailMerge to an ASP.NET page. Next, set its email-related properties: **FromAddressTemplate**, **SubjectTemplate**, **BodyTemplate**, and so on. These properties can be set through the Properties window or assigned programmatically based on user input or other criteria. Once these properties have been set, bind the list of recipients to the control and call the **SendMailMerge** method to send an email to each recipient.

The list of recipients can be provided in a number of ways - as an array, an ArrayList, a DataSet, a collection, or as an XML document, to name a few. The email messages sent by UltimateEmailMerge can be personalized on a recipient-by-

Karamasoft™

Web: www.karamasoft.com
Email: info@karamasoft.com

19 Cambria
Mission Viejo, CA 92692
Phone: 1-800-401-7543
Fax: 1-509-278-5588

recipient basis through the use of placeholders. A placeholder is text that appears in one of the template properties using syntax like **[%FirstName%]**. If the recipient list bound to the control includes a property or column named **FirstName**, the value of each recipient's **FirstName** property will be substituted into the placeholder. These placeholders make it easy to send customized emails, which are more likely to resonate with recipients.

Both UltimateEmailValidator and UltimateMailMerge can be used declaratively on a Web page (like the built-in ASP.NET validation controls) or programmatically from an ASP.NET or Windows application.

7. Improve Your Website's Navigation

Most developers prize function over form. It's why we prefer working on the business logic rather than implementing the user interface. Consequently, too many developers working on web applications focus on ensuring that the site contains great content without concerning themselves over how the content is accessed, viewed, or experienced by the user. But a website's navigation is as important as the quality of its content.

In ASP.NET version 1.x, developers were on their own in designing and implementing their site's navigation. ASP.NET 2.0 offers some basic navigation user interface elements – specifically the Menu, TreeView, and SiteMapPath controls – but these offerings from Microsoft lack the functionality and features found in UISuite. UISuite includes four navigation-related controls:

- UltimateMenu
- UltimateTabstrip
- UltimatePanel
- UltimateSitemap

UltimateMenu simplifies the process of providing a menu-based navigation system. The menu's items and structure can be hard-coded or bound to an XML file or DataSet. UltimateMenu's **AllowedUsers** property makes it easy to define menu item access rights on a user- or role-basis. Menus and their submenus can be laid out horizontally or vertically, and scrolling is available for especially long menus. Furthermore, UltimateMenus can be rendered as context-menus (menus that appear when the user right-clicks their mouse within the browser) by simply setting the **ContextMenu** property to True. Best of all, UltimateMenu's variety of pre-defined menu styles makes it easy for non-graphically inclined developers to create attractive menus.

Tabs are a common user interface element that enables a series of related choices to be grouped in a space-efficient manner. Tabs are especially common in configuration dialog boxes in Windows applications; they are becoming more prevalent in web applications, as well. UltimateTabstrip renders a tabbed user interface and, like the

Karamasoft™

Web: www.karamasoft.com
Email: info@karamasoft.com

19 Cambria
Mission Viejo, CA 92692
Phone: 1-800-401-7543
Fax: 1-509-278-5588

other user interface controls in the UISuite library, is a cinch to use. Check out the live demos to see the full range of styles, layout, and features offered by this control.

Many Windows applications use panels that can be resized and collapsed. For example, the myriad of windows in Visual Studio is all resizable and collapsible. UltimatePanel enables web developers to add such resizable and collapsible panels to their web pages in order to maximize valuable screen real estate.

The UltimateSitemap component displays a website's structure in a single page, which helps users quickly find what they are looking for. The sitemap data can come from an XML file or a DataSet and can be displayed in a tabular or hierarchical format. There are a number of built-in sitemap styles that make it easy to create slick, professional-looking sitemaps with little effort or time.

8. Build More Responsive User Interfaces

Anytime a user interacts with a website that requires data from the web server or the execution of server-side logic, the browser must send an HTTP request to the server. By default, such requests require that the *entire* state of the page be transmitted to the server and that the server return the entirety of the markup to display. Such large exchanges of data lead to sluggish user interfaces, requiring the user to wait for several seconds between each client/server interaction. Over the last several years, more and more web applications have turned to AJAX in order to improve the end user's experience. AJAX improves the responsiveness of a web application's user interface by replacing large postbacks with lightweight, asynchronous communications.

Configuring your website to take advantage of AJAX techniques is easy with UltimateAjax, another one of the components that makeup UISuite. Start by adding the UltimateAjax control to your web page. Next, add those controls that need to update asynchronously within the UltimateAjax control. Finally, use the UltimateAjax client-side API to initiate an asynchronous callback to update the controls within UltimateAjax. UltimateAjax works with both the built-in ASP.NET server controls as well as with all of the UISuite controls. The UISuite controls can also be seamlessly integrated with Microsoft ASP.NET AJAX. However, using Microsoft ASP.NET AJAX locks you into using ASP.NET 2.0; UltimateAjax works with both versions 1.x and 2.0.

Happy Programming!

About the Author

Scott Mitchell, author of seven books and founder of 4GuysFromRolla.com, has been working with Microsoft Web technologies since 1998. Scott works as an independent consultant, trainer, and writer. He can be reached at mitchell@4guysfromrolla.com or via his blog, which can be found at http://ScottOnWriting.NET.