Bryan L. Pratt

Summer Internship Portfolio

June-September 2013

Bellevue College – Programming for Web Development
Certificate of Achievement

WellKit Corporation

Michael Kysar – President and Development Manager
Sandra Eisert – CEO and Design Manager

Introduction

In early June, 2013, I accepted an intern position as a junior software developer for WellKit Corporation.

WellKit is a new startup venture headed up by Michael Kysar and Sandra Eisert. Michael and Sandra met as senior key members of Microsoft's MSNBC.com team. (See www.wellkit.org/home/about.)

Michael is a former public school music teacher, was conductor of the Boeing Choir for over 25 years, and worked on other major program teams at Microsoft, including Access and Visual Basic. (See http://pfco.com/About_MKysar.html.)

Sandra is a Pulitzer Prize winning photo editor, journalist, and designer, and was the first White House photo editor where she worked under three different administrations. (See http://sandraeisert.com/EisertBio.pdf.)

WellKit is a web based application designed to help people with chronic illness manage their condition. It provides tools to track, chart, correlate, and report their symptoms, keep a journal, and communicate with others who share their symptoms. (See http://www.wellkit.org/home/features.)

My primary assignments at WellKit were:

1. Implementation of the visual design of the site
2. Complete design and creation of the News pages, including administrative pages
3. Create custom server controls

This portfolio showcases the work I did on the News pages. This included:

- Writing a specification
- Designing and creating database tables in SQL Server
- Creating and testing stored procedures
- Creating classes and methods
- Building the interface

Specification

Manager: Michael Kysar – 425-747-0746 / 425-260-8479
Developer: Bryan Pratt – 425-216-3748

Purpose

Create the News and News archives pages with the tables, stored procedures, methods, and layout of the pages, and develop and testing plan and schedule for development and deployment.

Assumptions / Layout

- **'News' page** – 1st item under 'News' on the top navigation bar
  - 2 Columns
  - Uses main master page
  - Layout (within top and bottom nav bars):
    - Headline of the 'top' story - <h1> (crosses both columns)
    - Left column – Lead news story:
      - Sub-head of the 'top' story
      - Byline – Date posted (/Date published?) / Source (/Author?)
      - A summary of the story; ~60 – 70 words
      - 'Read more' section
        - Link to the original article
        - Links to referenced or related articles (1 or 2; optional?)
    - Right column:
      - 'Other News' section (sidebar)
        - 3 stories
        - Layout item (each):
          - Headline - <h2>
          - Link to story
      - Link to 'News Archive' page
    - Ad(s) below both columns
- **'News Archive' page** – 2nd item under 'News' on the top navigation bar
  - Uses main master page
  - Layout (within top and bottom nav bars):
  - Left column - wide
    - Layout item (each):
      - Headline
      - Sub-head
      - Link to story
      - Repeat until page is 'full'
      - Link to 'next page'
        - Clicking link repopulates page with next set of stories
        - Add a 'previous page' link
  - Right column – narrow

- Ads
- **'News Reader' page (?)** – Page that is part of the WellKit site with an iframe for displaying the selected news story from the original source (www). When a link to a story is selected on either the 'News' or 'News Archive' page, it opens to this page and loads the selected link into the iframe.
  - Uses main master page
  - Layout (within top and bottom nav bars):
    - Iframe – website of referenced news article appears her
    - Ad(s) below the iframe
- **Admin page** – Page for adding, editing, and deleting news items
  - Accessible only to users validated as administrators
  -

Data / table structure

Tables are required for the following data items:

- Headline
- Sub-head
- Body (summary)
- Links
- Source name
- Author
- Source URL

- Source publication date
- Creation date
- Status
- Published to site date
- Keywords
- Item order

wk_NewsArticles		
FieldName	**DataType**	**Key**
newsItemID	int	P
newsHeadline	nvchar(100)	
newsSubhead	nvchar(100)	
newsBody	nvchar(600)	
newsSourceName	varchar(100)	
newsAuthor	varchar(25)	
newsSourceURL	varchar(100)	
newsSourcePubDate	smalldatetime	
newsCreateDate	smalldatetime	
ContentStatus	int	enum
newsPubToSiteDate	smalldatetime	
newsKeywords	varchar(400)	
newsItemOrder	int	
newsViews	int	

wk_NewsLinks		
FieldName	**DataType**	**Key**
newsLinksID	int	P
newsItemID	int	F
newsLinksURL	varchar(100)	
newsLinkText	varchar(100)	
newsLinkOrder	int	
Enum: ContentStatus		
0	New	
1	Pending	
2	Review	
3	Released	
4	Withdrawn	

Stored Procedures

News items:

- wk_NewsAdd – add a new news item
- wk_NewsGetNewest – fetch the 4 most recent news item

- wk_NewsGet – fetch a specific news item
- wk_NewsUpdate – modify an existing news item
- wk_NewsDelete – 'delete' a news item – change status to 'Withdrawn' (4)

Links:

- wk_NewsLinksAdd – add links associated with a news item
- wk_NewsLinksGet – get links associated with a specific news item

Methods in news utility

Need one method to match each of the queries listed above:

Classes required

- NewsItemInfo
- NewsItemList
- NewsUtility

Basic layout

News page:

(See descriptions of other pages' layouts in the Assumptions / Layout section.)

Controls needed

- Headline: <asp:Label>
- Subhead: <asp:Label>
- Byline: <asp:Label>
- Body: <asp:Label>
- Links: <asp:DataList> with repeater. Include red triangle icon in repeater.
- Other news – within a Panel
    - OtherNewsHeadline: <asp:Label>
    - OtherNewsLinks: <asp:DataList> with repeater. Include red triangle icon in repeater.

Questions

- Should the byline of the top story include the author's name, or just the source? Should it include the date the article was published, or just the date it was posted?
- Should links be optional or mandatory?
- Include a control to let users 'vote' on a story's importance? This data could be used for sorting stories. (Later – beyond the scope of the prototype.)

Testing

Once data tables, queries, methods and classes are created, create News and Reader pages, add test data to tables (e.g., 'Test: Breaking news…'), and test populating the page. Create News Archive page, test. Create Admin page, test adding, editing, and deleting stories.
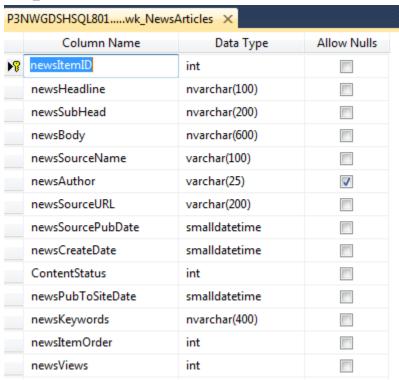
Schedule

- Spec 1st draft, data schema draft: 8/6/13
- Finalize drafts, create tables, write and test stored procedures: 8/7/13
- Write classes and methods; catch up: 8/8/13 - 8/12/13
- Create News and Reader pages: 8/13/13 - 8/16/13
- Debug classes and methods: 8/19/13 - 8/20/13
- Populate tables with test data, start testing: 8/20/13 – 8/23/13
- Create News Archive page, test: 8/23/13 – 8/27/13
- Create Admin page, test: 8/27/13 – 9/4/13
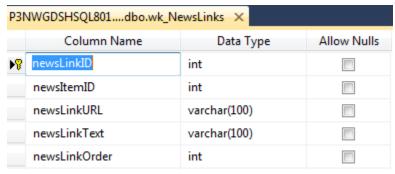- Catch up, test, debug, repeat until complete: 9/5/13 – 9/9/13

Database

Tables

Wk_NewsArticles

	P3NWGDSHSQL801.....wk_NewsArticles ✕	

Column Name	Data Type	Allow Nulls
newsItemID	int	☐
newsHeadline	nvarchar(100)	☐
newsSubHead	nvarchar(200)	☐
newsBody	nvarchar(600)	☐
newsSourceName	varchar(100)	☐
newsAuthor	varchar(25)	☑
newsSourceURL	varchar(200)	☐
newsSourcePubDate	smalldatetime	☐
newsCreateDate	smalldatetime	☐
ContentStatus	int	☐
newsPubToSiteDate	smalldatetime	☐
newsKeywords	nvarchar(400)	☐
newsItemOrder	int	☐
newsViews	int	☐

Wk_NewsLinks

	P3NWGDSHSQL801....dbo.wk_NewsLinks ✕	

Column Name	Data Type	Allow Nulls
newsLinkID	int	☐
newsItemID	int	☐
newsLinkURL	varchar(100)	☐
newsLinkText	varchar(100)	☐
newsLinkOrder	int	☐

wk_NewsAdd

Adds a news item to the wk_NewsArticles table.
Used in the AddItem method of NewsUtility.cs.

```sql
USE [WKSiteDB]
GO
/****** Object:  StoredProcedure [dbo].[wk_NewsAdd]    Script Date: 9/3/2013 1:40:42 PM
******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
-- =============================================
-- Author:         Bryan Pratt
-- Create date: 8/7/2013
-- Description:     Add a news item
-- =============================================
CREATE PROCEDURE [dbo].[wk_NewsAdd]
        @headline           nvarchar(100) = '',
        @subhead            nvarchar(100) = '',
        @body               nvarchar(600) = '',
        @sourceName         varchar(100) = '',
        @author             varchar(25) = '',
        @sourceURL          varchar(100) = '',
        @sourcePubDate      smalldatetime = '1/1/1900',
        @pubToSiteDate      smalldatetime = '1/1/1900',
        @keywords           nvarchar(400) = '',
        @itemOrder          int,
        @views              int,
        @newsID             int out

AS
BEGIN
        SET NOCOUNT ON;
        insert into wk_NewsArticles
                (       newsHeadline,
                        newsSubHead,
                        newsBody,
                        newsSourceName,
                        newsAuthor,
                        newsSourceURL,
                        newsSourcePubDate,
                        newsPubToSiteDate,
                        newsKeywords,
                        newsItemOrder,
                        newsViews)
        values
                (       @headline,
                        @subhead,
                        @body,
                        @sourceName,
                        @author,
                        @sourceURL,
```

```
                    @sourcePubDate,
                    @pubToSiteDate,
                    @keywords,
                    @itemOrder,
                    @views)

        select newsID = @@IDENTITY
        return 0
END
```

wk_NewsLinksAdd

Adds items to the wk_NewsLinks table.
Used in the AddItem method of NewsUtility.cs.

```
USE [WKSiteDB]
GO
/****** Object:  StoredProcedure [dbo].[wk_NewsLinksAdd]    Script Date: 9/3/2013 2:01:25
PM ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
-- =============================================
-- Author:          Bryan Pratt
-- Create date: 8/9/2013
-- Description:      Adds items to wk_NewsLinks table
-- =============================================
CREATE PROCEDURE [dbo].[wk_NewsLinksAdd]
        @newsItemID         int,
        @linkURL            varchar(100),
        @linkText           varchar(100),
        @linkOrder          int
AS
BEGIN
        SET NOCOUNT ON;
        insert into wk_NewsLinks
                (       newsItemID,
                        newsLinkURL,
                        newsLinktext,
                        newsLinkOrder)
        values
                (       @newsItemID,
                        @linkURL,
                        @linkText,
                        @linkOrder)

        return 0
END
```

wk_NewsGet

Gets a single news item from the wk_NewsArticles table using the newsID parameter.
Used in the GetItem method of NewsUtility.cs.

```
USE [WKSiteDB]
GO
/****** Object:  StoredProcedure [dbo].[wk_NewsGet]    Script Date: 9/3/2013 1:58:07 PM
******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
-- =============================================
-- Author:          Bryan Pratt
-- Create date: 8/7/2013
-- Description:     Gets a specific news item
-- =============================================
CREATE PROCEDURE [dbo].[wk_NewsGet]
       @newsid int = 0
AS
BEGIN
       select       top 1 *
       from         wk_NewsArticles
       where        wk_NewsArticles.newsItemID = @newsid

       select       *
       from         wk_NewsLinks
       where        newsItemID = @newsid
       order by     newsLinkOrder

       return 0
END
```

wk_NewsGetArchive

Gets all the 'archived' news items from the wk_NewsArticles table, meaning all but the four newest.
These are used to populate the NewsArchive page.
Used in the GetArchive method of NewsUtility.cs.

```
USE [WKSiteDB]
GO
/****** Object:  StoredProcedure [dbo].[wk_NewsGetArchive]    Script Date: 9/3/2013
1:58:33 PM ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
-- =============================================
-- Author:          Bryan L. Pratt
-- Create date: 8/20/2013
-- Description:     Gets all news items except most recent 4
-- =============================================
CREATE PROCEDURE [dbo].[wk_NewsGetArchive]
AS
```

```
BEGIN
      select        *
      from          wk_NewsArticles a
      where         a.newsItemID not in
                          (select top 4 newsItemID
                           from wk_NewsArticles a2
                           order by a2.newsPubToSiteDate desc, a2.newsItemID)
                          and
                          a.ContentStatus = 3
      order by      a.newsPubToSiteDate desc

END
```

wk_NewsGetList

Gets all news items in the wk_NewsArticles table; used for the NewsAdmin page.
Used in the GetNewsList method of NewsUtility.cs.

```
USE [WKSiteDB]
GO
/****** Object:  StoredProcedure [dbo].[wk_NewsGetList]    Script Date: 9/3/2013 1:58:55
PM ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
-- =============================================
-- Author:          Bryan L. Pratt
-- Create date: 8/27/2013
-- Description:     Get a list of all news articles for Admin page
-- =============================================
CREATE PROCEDURE [dbo].[wk_NewsGetList]
AS
BEGIN
      select        *
      from          wk_NewsArticles a
      order by      a.newsItemID
END
```

wk_NewsGetNewest

Gets the 4 most recent news items in the wk_NewsArticles table (by date published to the web site).
Used for the News page; only these 4 news stories are shown when the page first loads. Items returned
depend on the type of user logged on; Admins and Testers get all stories regardless of ContentStatus,
regular users get only "released" stories (ContentStatus = 3). User type passed in from the method using
the @testMode parameter.
Used in the GetList method of NewsUtility.cs.

```
USE [WKSiteDB]
GO
/****** Object:  StoredProcedure [dbo].[wk_NewsGetNewest]    Script Date: 9/3/2013
1:59:37 PM ******/
SET ANSI_NULLS ON
GO
```

```sql
SET QUOTED_IDENTIFIER ON
GO
-- =============================================
-- Author:         Bryan Pratt
-- Create date: 8/7/2013
-- Modified:  8/21/2013
-- Description:     Gets 4 most recent news item
-- =============================================
CREATE PROCEDURE [dbo].[wk_NewsGetNewest]
        @newsItem     int = 0 output,
        @testMode     bit = 0
AS
        declare           @itemDate datetime
        select     @itemDate =   newsPubToSiteDate
        from       wk_NewsArticles
        where      newsItemID = @newsItem

        if (@itemDate is null)
              set @itemDate = GetUtcDate()

        if (@testMode = 0)
              begin
                    -- Get a valid Top Story ID
                    if (@newsItem = 0)
                          select top 1 @newsItem = newsItemID
                          from   wk_NewsArticles
                          where  (newsSourcePubDate <= @itemDate)

                    select top 1 *
                    from   wk_NewsArticles
                    where  (newsItemID = @newsItem)

                    select top 3 *
                    from   wk_NewsArticles
                    where  (newsItemID < @newsItem)
                    order by newsItemID desc

              end

        else if (@testMode = 1)
              begin

                    if (@newsItem = 0)
                          select top 1 @newsItem = newsItemID
                          from   wk_NewsArticles
                          where  (newsCreateDate <= @itemDate)

                    select top 1 *
                    from   wk_NewsArticles
                    where  (newsItemID = @newsItem)

                    select top 3 *
                    from   wk_NewsArticles
                    where  (newsCreateDate > @itemDate)

              end
```

Code reproduced with permission from WellKit Corp.

```
select          *
from            wk_NewsLinks lks
inner join      wk_NewsArticles a
on                      lks.newsItemID = a.newsItemID
where           (lks.newsItemID = @newsItem)
order by        lks.newsLinkOrder

return 0
```

wk_NewsGetTopStoryID

Gets the ItemID of the most recently posted news item in the wk_NewsArticles table.
Used in the GetTopStoryID method of NewsUtility.cs.

```
USE [WKSiteDB]
GO
/****** Object:  StoredProcedure [dbo].[wk_NewsGetTopStoryID]    Script Date: 9/3/2013
2:00:59 PM ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
-- =============================================
-- Author:          Bryan L. Pratt
-- Create date: 8/23/2013
-- Description:     Gets the newsItemID of the most recent news item
-- =============================================
CREATE PROCEDURE [dbo].[wk_NewsGetTopStoryID]
AS
BEGIN
        select          top 1 newsItemID
        from            wk_NewsArticles
        where           ContentStatus = 3
        order by        wk_NewsArticles.newsPubToSiteDate Desc, newsItemID

        return 0
END
```

wk_NewsUpdate

Updates an existing news item in the wk_NewsArticles table.

```
USE [WKSiteDB]
GO
/****** Object:  StoredProcedure [dbo].[wk_NewsUpdate]    Script Date: 9/3/2013 2:02:04
PM ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
-- =============================================
-- Author:          Bryan Pratt
-- Create date: 8/7/2013
-- Description:     Modify news items
-- =============================================
```

```
CREATE PROCEDURE [dbo].[wk_NewsUpdate]
      @id int = 0,
      @headline           nvarchar(100) = '',
      @subhead            nvarchar(100) = '',
      @body               nvarchar(600) = '',
      @sourceName         varchar(100) = '',
      @author                   varchar(25) = '',
      @sourceURL          varchar(100) = '',
      @sourcePubDate      smalldatetime = '1/1/1900',
      @pubToSiteDate      smalldatetime = '1/1/1900',
      @keywords           nvarchar(400) = '',
      @itemOrder          int,
      @views              int

AS
BEGIN
      update wk_NewsArticles
      set
                  newsHeadline = @headline,
                  newsSubHead = @subhead,
                  newsBody = @body,
                  newsSourceName = @sourceName,
                  newsAuthor = @author,
                  newsSourceURL = @sourceURL,
                  newsSourcePubDate = @sourcePubDate,
                  newsPubToSiteDate = @pubToSiteDate,
                  newsKeywords = @keywords,
                  newsItemOrder = @itemOrder,
                  newsViews = @views

      where  newsItemID = @id

      return 0
END
```

wk_NewsLinksGet

Gets links for a particular news item from the wk_NewsLinks table.

```
USE [WKSiteDB]
GO
/****** Object:  StoredProcedure [dbo].[wk_NewsLinksGet]    Script Date: 9/3/2013 2:01:47
PM ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
-- =============================================
-- Author:          Bryan Pratt
-- Create date: 8/9/2013
-- Description:     Gets links associated with a specified news item
-- =============================================
CREATE PROCEDURE [dbo].[wk_NewsLinksGet]
      @newsID              int = 0
AS
BEGIN
```

```
        SET NOCOUNT ON;

        select *
        from wk_NewsLinks
        where newsItemID = @newsID
        order by newsLinkOrder
END
```

Classes
and
Methods

NewsItemInfo.cs

This class creates an object for holding a news item.

```csharp
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Data.SqlClient;
using System.Data;
using System.Web.UI;
using WellKitEngine.Data;

namespace WellKitEngine.Site.News
{
    public class NewsItemInfo
    {
        #region Private Fields

        private bool _isDirty = false;
        private bool _isLoading = true;

        #endregion

        #region Properties

        public int ID { get; set; }
        public string Headline { get; set; }
        public string SubHead { get; set; }
        public string Body { get; set; }
        public string SourceName { get; set; }
        public string Author { get; set; }
        public string SourceURL { get; set; }
        public DateTime SourcePubDate { get; set; }
        public DateTime CreateDate { get; set; }
        public ContentStatus ContentStatus { get; set; }
        public DateTime PubToSiteDate { get; set; }
        public string Keywords { get; set; }
        public int ItemOrder { get; set; }
        public int Views { get; set; }
        public List<NewsLinkInfo> Links { get; set; }
        public List<NewsItemInfo> Items { get; set; }

        #endregion

        #region Constructors
        public NewsItemInfo()
        {
        }

        public NewsItemInfo(SqlDataReader dr)
        {
            if (dr == null || !dr.HasRows)
            {
                ID = -1;
```

```
            _isLoading = false;
            return;
        }

        ID = (Int32)dr["newsItemID"];
        Headline = SqlUtility.GetStringFromReader(dr, "newsHeadline");
        SubHead = SqlUtility.GetStringFromReader(dr, "newsSubHead");
        Body = SqlUtility.GetStringFromReader(dr, "newsBody");
        SourceName = SqlUtility.GetStringFromReader(dr, "newsSourceName");
        Author = SqlUtility.GetStringFromReader(dr, "newsAuthor");
        SourceURL = SqlUtility.GetStringFromReader(dr, "newsSourceURL");
        SourcePubDate = SqlUtility.GetDateFromReader(dr, "newsSourcePubDate");
        CreateDate = SqlUtility.GetDateFromReader(dr, "newsCreateDate");
        ContentStatus = (ContentStatus)dr["ContentStatus"];
        PubToSiteDate = SqlUtility.GetDateFromReader(dr, "newsPubToSiteDate");
        Keywords = SqlUtility.GetStringFromReader(dr, "newsKeywords");
        ItemOrder = (Int32)dr["newsItemOrder"];
        Views = (Int32)dr["newsViews"];

    }

    #endregion

    }
}
```

NewsItemList.cs

This class simply creates a list object of type NewsItemInfo. Used for holding lists of news items. It consists of nothing more than a constructor.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Data.SqlClient;
using WellKitEngine.Data;


namespace WellKitEngine.Site.News
{
    public class NewsItemList : List<NewsItemInfo>
    {

    }
}
```

NewsLinkInfo.cs

This class creates a news link object.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
```

Code reproduced with permission from WellKit Corp.

```csharp
using System.Data.SqlClient;
using WellKitEngine.Data;

namespace WellKitEngine.Site.News
{
    public class NewsLinkInfo
    {
        #region Private Fields

        private bool _isDirty = false;
        private bool _isLoading = true;

        #endregion

        #region Properties

        public int LinkID { get; set; }
        public int NewsItemID { get; set; }
        public string LinkURL { get; set; }
        public string LinkText { get; set; }
        public int LinkOrder { get; set; }

        #endregion

        #region Constructors

        public NewsLinkInfo()
        {
        }

        public NewsLinkInfo(SqlDataReader dr)
        {
            if (dr == null || !dr.HasRows)
            {
                LinkID = -1;
            }
            else
            {
                LinkID = (Int32)dr["newsLinkID"];
                NewsItemID = (Int32)dr["newsItemID"];
                LinkURL = SqlUtility.GetStringFromReader(dr, "newsLinkURL");
                LinkText = SqlUtility.GetStringFromReader(dr, "newsLinkText");
                LinkOrder = (Int32)dr["newsLinkOrder"];
            }
            _isLoading = false;
        }

        #endregion
    }
}
```

Enumerations.cs

Defines an enumeration for the ContentStatus property. This enumeration is used to populate drop-down lists in the interface.

```csharp
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace WellKitEngine.Site.News
{
    public enum ContentStatus : int
    {
        New = 0,
        Pending = 1,
        Review = 2,
        Released = 3,
        Withdrawn = 4
    }
}
```

Methods

NewsUtility.cs

This utility contains the methods used by the code behind for the News.aspx and NewsArchive.aspx pages.

```csharp
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Data;
using System.Data.SqlClient;
using WellKitEngine.Data;


namespace WellKitEngine.Site.News
{
    public static class NewsUtility
    {
        /// <summary>
        /// Adds a news item to the database
        /// </summary>
        /// <param name="newsItem">NewsItemInfo newsItem</param>
        public static void AddItem(NewsItemInfo newsItem)
        {
            int newsID = 0;

            using (SqlUtility sql = new SqlUtility("wk_NewsAdd"))
            {
                sql.Command.Parameters.AddWithValue("@headline", newsItem.Headline);
                sql.Command.Parameters.AddWithValue("@subhead", newsItem.SubHead);
                sql.Command.Parameters.AddWithValue("@body", newsItem.Body);
                sql.Command.Parameters.AddWithValue("@sourceName", newsItem.SourceName);
                sql.Command.Parameters.AddWithValue("@author", newsItem.Author);
                sql.Command.Parameters.AddWithValue("@sourceURL", newsItem.SourceURL);
                sql.Command.Parameters.AddWithValue("@sourcePubDate",
newsItem.SourcePubDate);
                sql.Command.Parameters.AddWithValue("@pubToSiteDate",
newsItem.PubToSiteDate);
                sql.Command.Parameters.AddWithValue("@keywords", newsItem.Keywords);
                sql.Command.Parameters.AddWithValue("@itemOrder", newsItem.ItemOrder);
                sql.Command.Parameters.AddWithValue("@views", newsItem.Views);
                sql.Command.Parameters.Add("@newsID", SqlDbType.Int);
                sql.Command.Parameters["@newsID"].Direction = ParameterDirection.Output;

                sql.Open();
                sql.ExecuteCommand();

                newsID = (int)sql.Command.Parameters["@newsID"].Value;

                SqlCommand cmd = sql.GetCommand("wk_NewsLinksAdd");

                sql.Close();

            }
```

```csharp
        }

        /// <summary>
        /// Updates a news item in the database
        /// </summary>
        /// <param name="newsItem">NewsItemInfo newsItem</param>
        public static void UpdateItem(NewsItemInfo newsItem)
        {
            using (SqlUtility sql = new SqlUtility("wk_NewsUpdate"))
            {
                sql.Command.Parameters.AddWithValue("@id", newsItem.ID);
                sql.Command.Parameters.AddWithValue("@headline", newsItem.Headline);
                sql.Command.Parameters.AddWithValue("@subhead", newsItem.SubHead);
                sql.Command.Parameters.AddWithValue("@body", newsItem.Body);
                sql.Command.Parameters.AddWithValue("@sourceName", newsItem.SourceName);
                sql.Command.Parameters.AddWithValue("@author", newsItem.Author);
                sql.Command.Parameters.AddWithValue("@sourceURL", newsItem.SourceURL);
                sql.Command.Parameters.AddWithValue("@sourcePubDate",
newsItem.SourcePubDate);
                sql.Command.Parameters.AddWithValue("@pubToSiteDate",
newsItem.PubToSiteDate);
                sql.Command.Parameters.AddWithValue("@keywords", newsItem.Keywords);
                sql.Command.Parameters.AddWithValue("@itemOrder", newsItem.ItemOrder);
                sql.Command.Parameters.AddWithValue("@views", newsItem.Views);

                sql.Open();
                sql.ExecuteCommand();
                sql.Close();

            }
        }

        /// <summary>
        /// Gets a specified news item from the database
        /// </summary>
        /// <param name="newsID">Integer news item ID</param>
        /// <returns>An news item object if it exists, otherwise null</returns>
        public static NewsItemInfo GetItem(int newsID)
        {
            NewsItemInfo ns = null;

            using (SqlUtility sql = new SqlUtility("wk_NewsGet"))
            {
                sql.Command.Parameters.AddWithValue("@newsid", newsID);

                sql.Open();

                SqlDataReader dr = sql.ExecuteReader();

                if (dr.Read())
                    ns = new NewsItemInfo(dr);

                if (dr.NextResult())
                {
                    List<NewsLinkInfo> list = new List<NewsLinkInfo>();
                    while (dr.Read())
```

```csharp
                    list.Add(new NewsLinkInfo(dr));

                ns.Links = list;
            }

            dr.Close();
            sql.Close();
        }

        return ns;
    }

    /// <summary>
    /// Gets a list of the 4 most recent news items from the database
    /// </summary>
    /// <returns>A list of the 4 most recent news item objects</returns>
    public static NewsItemList GetList(int newsID = 0)
    {
        NewsItemList nl = new NewsItemList();
        int idx = 0, itemCount = 0;

        using (SqlUtility sql = new SqlUtility("wk_NewsGetNewest"))
        {
            sql.Command.Parameters.AddWithValue("@newsItem", newsID);

            sql.Open();

            SqlDataReader dr = sql.ExecuteReader();

            while (dr.Read())
            {
                nl.Add(new NewsItemInfo(dr));
                itemCount++;
            }

            for (idx = 1; idx < itemCount; idx++)
                nl[idx].SubHead = SiteUtility.TruncateText(nl[idx].SubHead, 140,
true);

            idx = 0;

            if (dr.NextResult())
            {

                NewsLinkInfo linkItem = null;

                while (dr.Read())
                {
                    linkItem = new NewsLinkInfo(dr);
                    if (linkItem.NewsItemID != nl[idx].ID)
                        idx++;

                    if (nl[idx].Links == null)
                        nl[idx].Links = new List<NewsLinkInfo>();

                    nl[idx].Links.Add(linkItem);
```

Methods

```csharp
                }
            }
            dr.Close();
            sql.Close();
        }
        return nl;
    }

    public static int GetTopStoryID()
    {
        int newsItemID = 0;

        using (SqlUtility sql = new SqlUtility("wk_NewsGetTopStoryID"))
        {
            sql.Open();
            newsItemID = (Int32)sql.Command.ExecuteScalar();

            sql.Close();
        }
        return newsItemID;
    }

    /// <summary>
    /// Gets 'archived' news stories (all but the first 4)
    /// </summary>
    /// <returns></returns>
    public static NewsItemList GetArchive()
    {
        NewsItemList nl = new NewsItemList();
        int idx = 0, itemCount = 0;

        using (SqlUtility sql = new SqlUtility("wk_NewsGetArchive"))
        {
            sql.Open();
            SqlDataReader dr = sql.ExecuteReader();

            while (dr.Read())
            {
                nl.Add(new NewsItemInfo(dr));
                itemCount++;
            }

            for (idx = 0; idx < itemCount; idx++)
                nl[idx].SubHead = SiteUtility.TruncateText(nl[idx].SubHead, 96,
true);

            dr.Close();
            sql.Close();
        }
        return nl;
    }

    public static NewsItemList GetNewsList()
    {
        NewsItemList nl = new NewsItemList();
        int idx = 0, itemCount = 0;
```

Methods

```csharp
        using (SqlUtility sql = new SqlUtility("wk_NewsGetList"))
        {
            sql.Open();
            SqlDataReader dr = sql.ExecuteReader();

            while (dr.Read())
            {
                nl.Add(new NewsItemInfo(dr));
                itemCount++;
            }

            for (idx = 0; idx < itemCount; idx++)
                nl[idx].Headline = SiteUtility.TruncateText(nl[idx].Headline, 25,
true);

            dr.Close();
            sql.Close();
        }
        return nl;
    }

    /// <summary>
    /// 'Deletes' a specified news item by changing its ContentStatus to 4
('withdrawn')
    /// </summary>
    /// <param name="newsID">Integer news item ID</param>
    public static void DeleteItem(int newsID)
    {
        using (SqlUtility sql = new SqlUtility("wk_NewsDelete"))
        {
            sql.Command.Parameters.AddWithValue("@newsID", newsID);

            sql.Open();
            sql.ExecuteCommand();

            sql.Close();
        }
    }
    }
}
```

Pages
and
Code-behind

News.aspx

The News page.

```
<%@ Page Title="" Language="C#" MasterPageFile="~/Site/MasterPages/WellKit2.Master"
AutoEventWireup="true" CodeBehind="News.aspx.cs"
Inherits="WellKitOrg.Site.Content.News.News" %>

<asp:Content ID="headContent" ContentPlaceHolderID="HeadContent" runat="server">
</asp:Content>
<asp:Content ID="MainContent" ContentPlaceHolderID="MainContent" runat="server">
    <%--Data control for all news items--%>
    <asp:FormView ID="NewsView" runat="server" AllowPaging="false"
EnableViewState="false" CssClass="news_content" OnDataBound="NewsView_DataBound">
        <ItemTemplate>
            <h1 class="headline"><%# Eval("Headline") %></h1>
            <%--Left panel contains the selected news item--%>
            <asp:Panel ID="mainNewsContent" runat="server" CssClass="news_content_left">
                <p class="news_subhead">
                    <%# Eval("Subhead") %>
                </p>
                <p class="byline"><span><%# Eval("SourcePubDate", "{0:MMMM d, yyyy}")
%></span> | <span><%# Eval("SourceName") %></span></p>
                <p class="news_body"><%# Eval("Body") %></p>
                <p class="more_label">Read more:</p>
                <asp:Table ID="LinksTable" runat="server" >
                    <asp:TableRow runat="server">
                        <asp:TableCell runat="server">
                            <img class="arrow_icon" src="/Site/Images/right-arrow.png"
alt="right arrow"></img>
                        </asp:TableCell>
                        <asp:TableCell runat="server" >
                            <p class="link" style="font-size:13px; font-size:0.813rem;">
                            <asp:HyperLink ID="LinkToStory" runat="server"
NavigateUrl=<%# Eval("SourceURL") %>
                                ToolTip=<%# Eval("Headline") %> Target="_blank" >
                                Original article
                            </asp:HyperLink>
                            </p>
                        </asp:TableCell>
                    </asp:TableRow>
                </asp:Table>
                <%--Links for the selected news item go in this list--%>
                <asp:ListView ID="ReadMoreList" runat="server" DataMember="Links" >
                    <ItemTemplate>
                        <asp:Table ID="ReadMoreTable" runat="server">
                            <asp:TableRow runat="server">
                                <asp:TableCell ID="TableCell1" runat="server">
                                    <img class="arrow_icon" src="/Site/Images/right-
arrow.png" alt="right arrow"></img>
                                </asp:TableCell>
                                <asp:TableCell ID="TableCell2" runat="server" >
                                    <p class="link">
                                    <asp:HyperLink ID="LinkToMore" runat="server"
NavigateUrl=<%# Eval("LinkURL") %>
```

```aspx
                                    ToolTip=<%# Eval("LinkText") %> Target="_blank" >
                                    <%# Eval("LinkText") %>
                                </asp:HyperLink>
                                </p>
                            </asp:TableCell>
                        </asp:TableRow>
                    </asp:Table>
                </ItemTemplate>
            </asp:ListView>
        </asp:Panel>

        <%--The right column contains links to other news items--%>
        <asp:Panel ID="otherNewsColumn" runat="server" CssClass="news_content_right">

            <%--Temporary link to admin page--%>
            <%--<asp:HyperLink ID="AdminTest" runat="server"
NavigateUrl="~/Site/Admin/News/NewsAdmin.aspx" Text="Admin Page" />--%>

                <nav class="other_news_bar">
                    OTHER NEWS
                </nav>

                <asp:ListView ID="OtherNewsList" runat="server">
                    <ItemTemplate>
                        <asp:Table ID="OtherNewsTable" runat="server">
                            <asp:TableRow runat="server" >
                                <asp:TableHeaderCell runat="server"
class="other_news_headline" ColumnSpan="2">
                                    <asp:HyperLink ID="ArticleLink" runat="server"
Text='<%# Eval("Headline") %>'
                                        NavigateURL='<%# Eval("ID","/news/{0}") %>' />

                                </asp:TableHeaderCell>
                            </asp:TableRow>
                            <asp:TableRow>
                                <asp:TableCell ID="TableCell2" runat="server" >
                                    <p class="other_news_subhead">
                                        <%# Eval("Subhead") %>
                                    </p>
                                </asp:TableCell>

                            </asp:TableRow>
                        </asp:Table>
                    </ItemTemplate>
                </asp:ListView>
                <nav class="news_archive_bar">
                    <asp:HyperLink ID="ArchiveBarLink" runat="server"
NavigateUrl="/newsarchive" Text="NEWS ARCHIVE" />
                </nav>
            </asp:Panel>
        </ItemTemplate>
    </asp:FormView>
    <div style="text-align: center">
        <iframe src="http://rcm-na.amazon-adsystem.com/e/cm?t=wel09a-
20&o=1&p=48&l=ur1&category=wireless&banner=0TJY74H4VE9W4MDG6J82&f=ifr" width="728"
        height="90" scrolling="no" border="0" marginwidth="0" style="border:none;"
frameborder="0"></iframe>
    </div>
```

Code reproduced with permission from WellKit Corp.

```
</asp:Content>
```

News.aspx.cs

Code-behind for News.aspx.

```csharp
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using WellKitEngine.Site.News;
using WellKitEngine.Site;
using System.Web.Routing;

namespace WellKitOrg.Site.Content.News
{
    public partial class News : System.Web.UI.Page
    {
        NewsItemList newsItems = null;

        protected void Page_Load(object sender, EventArgs e)
        {
            if (!IsPostBack)
            {
                int newsID = 0;
                string obj = (string)RouteData.Values["id"];

                if (!String.IsNullOrEmpty(obj))
                    newsID = Int32.Parse(obj);

                newsItems = NewsUtility.GetList(newsID);

                newsID = newsItems[0].ID;
                //Context.RewritePath(String.Format("/news/{0}", newsID));

                NewsView.DataSource = newsItems;
                DataBind();
                SiteUtility.RegisterRoutes(RouteTable.Routes);

                HyperLink ArchiveBarLink = NewsView.FindControl("ArchiveBarLink") as
HyperLink;
                if (ArchiveBarLink != null)
                    ArchiveBarLink.NavigateUrl = GetRouteUrl("NewsArchive", null);
            }
        }

        protected void NewsView_DataBound(object sender, EventArgs e)
        {
            ListView ReadMoreList = NewsView.FindControl("ReadMoreList") as ListView;

            if (ReadMoreList != null)
            {
                ReadMoreList.DataSource = newsItems[0].Links;
                ReadMoreList.DataBind();
```

Code reproduced with permission from WellKit Corp.

```csharp
            }

            ListView OtherNewsList = NewsView.FindControl("OtherNewsList") as ListView;

            if (OtherNewsList != null)
            {
                newsItems.Remove(newsItems[0]);
                OtherNewsList.DataSource = newsItems;
                OtherNewsList.DataBind();
            }
        }

    }
}
```

NewsArchive.aspx
The News Archive page.

```aspx
<%@ Page Title="" Language="C#" MasterPageFile="/Site/MasterPages/WellKit2.Master"
AutoEventWireup="true" CodeBehind="NewsArchive.aspx.cs"
Inherits="WellKitOrg.Site.Content.News.NewsArchive" %>

<asp:Content ID="headContent" ContentPlaceHolderID="HeadContent" runat="server">
</asp:Content>
<asp:Content ID="mainContent" ContentPlaceHolderID="MainContent" runat="server">
    <h1 class="headline">News Archive</h1>
    <asp:Panel ID="ArchiveContent" runat="server" CssClass="content_wide"
OnPreRender="ArchiveContent_PreRender"  >
        <asp:ListView ID="ArchiveListView" runat="server"  >
            <ItemTemplate>
                <asp:Table ID="ArchiveTable" runat="server" CellPadding="0">
                    <asp:TableRow ID="HeadlineRow" runat="server">
                        <asp:TableCell ID="ArchiveHeadline" runat="server"
CssClass="other_news_headline">
                            <asp:HyperLink ID="ArticleLink" runat="server" Text='<%#
Eval("Headline") %>'
                                    NavigateURL='<%# Eval("ID","/news/{0}") %>' />
                            <%--<%# Eval("Headline") %>--%>
                        </asp:TableCell></asp:TableRow><asp:TableRow ID="SubheadRow"
runat="server">
                        <asp:TableCell ID="ArchiveSubhead" runat="server"
CssClass="archive_subhead">
                            <%# Eval("Subhead") %>

</asp:TableCell></asp:TableRow></asp:Table></ItemTemplate></asp:ListView><asp:DataPager
ID="ArchivePager" runat="server" PagedControlID="ArchiveListView" PageSize="7">
            <Fields>
                <asp:NextPreviousPagerField ButtonCssClass="archive_button"
                                            ShowFirstPageButton="false"
                                            ShowNextPageButton="true"
                                            ShowLastPageButton="false"
                                            ShowPreviousPageButton="true" />
            </Fields>
        </asp:DataPager>
        <span>
            <asp:HyperLink ID="ArchiveBarLink" runat="server"
                NavigateUrl="/news" CssClass="archive_button"
                ForeColor="#5c4592" Font-Bold="false" Text="Latest News" />
```

```
        </span>
    </asp:Panel>
</asp:Content>
```

NewsArchive.aspx.cx

Code-behind for NewsArchive.aspx.

```csharp
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using WellKitEngine.Site.News;

namespace WellKitOrg.Site.Content.News
{
    public partial class NewsArchive : System.Web.UI.Page
    {
        NewsItemList archiveList = null;

        protected void Page_Load(object sender, EventArgs e)
        {
            archiveList = NewsUtility.GetArchive();

            ArchiveListView.DataSource = archiveList;
            DataBind();
        }

        protected void ArchiveContent_PreRender(object sender, EventArgs e)
        {
            ArchiveContent.DataBind();
        }
    }
}
```

NewsAdmin.aspx

The admin page for the News and News Archive pages.

```aspx
<%@ Page Title="Admin: Edit News Articles" Language="C#"
MasterPageFile="~/Site/MasterPages/WellKitAdmin.master" AutoEventWireup="true"
    CodeBehind="NewsAdmin.aspx.cs" Inherits="WellKitOrg.Site.Admin.News.NewsAdmin" %>

<asp:Content ID="headContent" ContentPlaceHolderID="HeadContent" runat="server">
    <title>Admin: Edit News Articles</title>
    <style type="text/css">
        .admin_pad {
            padding:6px;
            padding:0.375rem;
        }
    </style>
</asp:Content>
<asp:Content ID="mainContent" ContentPlaceHolderID="MainContent" runat="server">
    <asp:ScriptManager ID="ScriptManager" runat="server"></asp:ScriptManager>
    <table style="width: 1200px;">
        <tr>
            <td style="vertical-align: top">
                <asp:GridView ID="NewsListGridView" runat="server" DataKeyNames="ID"
Font-Size="0.938em"
                    AutoGenerateColumns="False" CellPadding="4"
AutoGenerateSelectButton="True" ForeColor="#333333" GridLines="None">
                    <AlternatingRowStyle BackColor="White" ForeColor="#284775" />
                    <Columns>
                        <asp:BoundField DataField="Headline" HeaderText="Headline"
SortExpression="Headline" />
                        <asp:BoundField DataField="PubToSiteDate"
DataFormatString="{0:d}" HeaderText="PubToSite" SortExpression="PubToSite" />
                    </Columns>
                    <EditRowStyle BackColor="#999999" />
                    <FooterStyle BackColor="#5D7B9D" ForeColor="White" Font-Bold="True"
/>
                    <HeaderStyle BackColor="#5D7B9D" Font-Bold="True" ForeColor="White"
/>
                    <PagerStyle BackColor="#284775" ForeColor="White"
HorizontalAlign="Center" />
                    <RowStyle BackColor="#F7F6F3" ForeColor="#333333" />
                    <SelectedRowStyle BackColor="#E2DED6" Font-Bold="True"
ForeColor="#333333" />
                    <SortedAscendingCellStyle BackColor="#E9E7E2" />
                    <SortedAscendingHeaderStyle BackColor="#506C8C" />
                    <SortedDescendingCellStyle BackColor="#FFFDF8" />
                    <SortedDescendingHeaderStyle BackColor="#6F8DAE" />
                </asp:GridView>
            </td>
            <td style="width: 800px; vertical-align: top">
                <asp:ObjectDataSource ID="NewsItemDetailsDS" runat="server"
DataObjectTypeName="WellKitEngine.Site.News.NewsItemInfo"
                    DeleteMethod="DeleteItem" InsertMethod="AddItem"
SelectMethod="GetItem" TypeName="WellKitEngine.Site.News.NewsUtility"
                    UpdateMethod="UpdateItem">
```

```xml
                    <DeleteParameters>
                        <asp:Parameter Name="newsID" Type="Int32" />
                    </DeleteParameters>
                    <SelectParameters>
                        <asp:ControlParameter ControlID="NewsListGridView" Name="newsID"
PropertyName="SelectedValue" Type="Int32" DefaultValue="1" />
                    </SelectParameters>
                    <UpdateParameters>
                        <asp:Parameter Name="Headline" Type="String" />
                        <asp:Parameter Name="SubHead" Type="String" />
                        <asp:Parameter Name="Body" Type="String" />
                        <asp:Parameter Name="SourceName" Type="String" />
                        <asp:Parameter Name="Author" Type="String" />
                        <asp:Parameter Name="SourceURL" Type="String" />
                        <asp:Parameter Name="SourcePubDate" Type="DateTime" />
                        <asp:Parameter Name="ContentStatus" Type="Int32" />
                        <asp:Parameter Name="PubToSiteDate" Type="DateTime" />
                        <asp:Parameter Name="Keywords" Type="String" />
                    </UpdateParameters>
                    <InsertParameters>
                        <asp:Parameter Name="Headline" Type="String" />
                        <asp:Parameter Name="SubHead" Type="String" />
                        <asp:Parameter Name="Body" Type="String" />
                        <asp:Parameter Name="SourceName" Type="String" />
                        <asp:Parameter Name="Author" Type="String" />
                        <asp:Parameter Name="SourceURL" Type="String" />
                        <asp:Parameter Name="SourcePubDate" Type="DateTime" />
                        <asp:Parameter Name="Keywords" Type="String" />
                    </InsertParameters>
                    <DeleteParameters>
                        <asp:ControlParameter ControlID="NewsListGridView"
DefaultValue="0" Name="newsID" PropertyName="SelectedValue"
                            Type="Int32" />
                    </DeleteParameters>
                </asp:ObjectDataSource>
                <asp:DetailsView ID="NewsItemDetailsView" runat="server" Height="50px"
Width="800px"
                    AutoGenerateRows="False" Font-Size="0.875em" CellPadding="4"
DataSourceID="NewsItemDetailsDS"
                    ForeColor="#333333" GridLines="None"
OnItemUpdating="NewsItemDetailsView_ItemUpdating">
                    <AlternatingRowStyle BackColor="White" ForeColor="#284775" />
                    <CommandRowStyle BackColor="#E2DED6" Font-Bold="True" />
                    <EditRowStyle BackColor="#e8e8e8" />
                    <FieldHeaderStyle BackColor="#E9ECF1" Font-Bold="True" />
                    <Fields>
                        <asp:CommandField ShowDeleteButton="True" ShowEditButton="True"
                            ShowInsertButton="True" ControlStyle-
CssClass="admin_controls" />
                        <asp:BoundField DataField="ID" HeaderText="ID"
SortExpression="ID" ReadOnly="true"
                            InsertVisible="false" ControlStyle-CssClass="admin_pad" />
                        <asp:TemplateField HeaderText="Headline">
                            <ItemTemplate>
                                <asp:Label ID="HeadlineLabel" runat="server" Text='<%#
Eval("Headline") %>' CssClass="admin_pad" />
                            </ItemTemplate>
                            <EditItemTemplate>
```

```
                                            <asp:TextBox ID="HeadlineEditTextbox" runat="server"
Text='<%# Bind("Headline") %>'
                                                Columns="98" MaxLength="100" TextMode="SingleLine"
Font-Names="Arial, Helvetica" />
                                    </EditItemTemplate>
                                    <InsertItemTemplate>
                                            <asp:TextBox ID="HeadlineNewTextbox" runat="server"
Text='<%# Bind("Headline") %>'
                                                Columns="98" MaxLength="100" TextMode="SingleLine"
Font-Names="Arial, Helvetica" />
                                    </InsertItemTemplate>
                                </asp:TemplateField>
                                <asp:TemplateField HeaderText="Subhead" ItemStyle-
VerticalAlign="Top">
                                    <ItemTemplate>
                                        <asp:Label ID="SubheadLabel" runat="server" Text='<%#
Eval("SubHead") %>' />
                                    </ItemTemplate>
                                    <EditItemTemplate>
                                        <asp:TextBox ID="SubheadEditTextbox" runat="server"
Text='<%# Bind("SubHead") %>'
                                            Columns="100" TextMode="MultiLine" onkeypress="return
this.value.length<=199"
                                            Rows="2" Font-Names="Arial, Helvetica" />
                                    </EditItemTemplate>
                                    <InsertItemTemplate>
                                        <asp:TextBox ID="SubHeadEditTextbox" runat="server"
Text='<%# Bind("SubHead") %>'
                                            TextMode="MultiLine" onkeypress="return
this.value.length<=199" Columns="100"
                                            Rows="2" Font-Names="Arial, Helvetica" />
                                    </InsertItemTemplate>
                                </asp:TemplateField>
                                <asp:TemplateField HeaderText="Body" ItemStyle-
VerticalAlign="Top">
                                    <ItemTemplate>
                                        <asp:Label ID="BodyLabel" runat="server" Text='<%#
Eval("Body") %>' />
                                    </ItemTemplate>
                                    <EditItemTemplate>
                                        <asp:TextBox ID="BodyEditTextbox" runat="server"
Text='<%# Bind("Body") %>'
                                            Columns="110" TextMode="MultiLine" onkeypress="return
this.value.length<=599"
                                            Rows="10" Font-Names="Arial, Helvetica" />
                                        <asp:HtmlEditorExtender ID="BodyTextEditorExtender"
runat="server"
                                            TargetControlID="BodyEditTextbox"
DisplaySourceTab="true"
                                            EnableSanitization="false">
                                            <Toolbar>
                                                <asp:Bold />
                                                <asp:Italic />
                                                <asp:InsertOrderedList />
                                                <asp:InsertUnorderedList />
                                                <asp:Indent />
                                                <asp:Undo />
                                                <asp:Redo />
```

```
                                        </Toolbar>
                                </asp:HtmlEditorExtender>
                        </EditItemTemplate>
                        <InsertItemTemplate>
                                <asp:TextBox ID="BodyNewTextbox" runat="server" Text='<%#
Bind("Body") %>'
                                        Columns="110" TextMode="MultiLine" onkeypress="return
this.value.length<=599"
                                        Rows="10" Font-Names="Arial, Helvetica" />
                                <asp:HtmlEditorExtender ID="BodyNewEditorExtender"
runat="server"
                                        TargetControlID="BodyNewTextbox"
DisplaySourceTab="true"
                                        EnableSanitization="false" >
                                        <Toolbar>
                                                <asp:Bold />
                                                <asp:Italic />
                                                <asp:InsertOrderedList />
                                                <asp:InsertUnorderedList />
                                                <asp:Indent />
                                                <asp:Undo />
                                                <asp:Redo />
                                        </Toolbar>
                                </asp:HtmlEditorExtender>
                        </InsertItemTemplate>
                </asp:TemplateField>
                <asp:TemplateField HeaderText="SourceName">
                        <ItemTemplate>
                                <asp:Label ID="SourceNameLabel" runat="server" Text='<%#
Eval("SourceName") %>' />
                        </ItemTemplate>
                        <EditItemTemplate>
                                <asp:TextBox ID="SourceNameEditTextbox" runat="server"
Text='<%# Bind("SourceName") %>'
                                        Columns="98" MaxLength="100" TextMode="SingleLine" Font-
Names="Arial, Helvetica" />
                        </EditItemTemplate>
                        <InsertItemTemplate>
                                <asp:TextBox ID="SourceNameNewTextbox" runat="server"
Text='<%# Bind("SourceName") %>'
                                        Columns="98" MaxLength="100" TextMode="SingleLine" Font-
Names="Arial, Helvetica" />
                        </InsertItemTemplate>
                </asp:TemplateField>
                <asp:BoundField DataField="Author" HeaderText="Author"
SortExpression="Author" />
                <asp:TemplateField HeaderText="SourceURL" ItemStyle-
VerticalAlign="Top">
                        <ItemTemplate>
                                <asp:Label ID="SourceURLLabel" runat="server" Text='<%#
Eval("SourceURL") %>' />
                        </ItemTemplate>
                        <EditItemTemplate>
                                <asp:TextBox ID="SourceURLEditTextbox" runat="server"
Text='<%# Bind("SourceURL") %>'
                                        Columns="100" TextMode="MultiLine" onkeypress="return
this.value.length<=199"
                                        Rows="2" Font-Names="Arial, Helvetica" />
```

```
                                        </EditItemTemplate>
                                        <InsertItemTemplate>
                                            <asp:TextBox ID="SourceURLEditTextbox" runat="server"
Text='<%# Bind("SourceURL") %>'
                                                TextMode="MultiLine" onkeypress="return
this.value.length<=199" Columns="100"
                                                Rows="2" Font-Names="Arial, Helvetica" />
                                        </InsertItemTemplate>
                                    </asp:TemplateField>
                                    <asp:TemplateField HeaderText="SourcePubDate">
                                        <ItemTemplate>
                                            <asp:Label ID="SourcePubDateLabel" runat="server"
Text='<%# Eval("SourcePubDate", "{0:d}") %>' />
                                        </ItemTemplate>
                                        <EditItemTemplate>
                                            <asp:TextBox ID="SourcePubDateEditTextbox" runat="server"
Text='<%# Bind("SourcePubDate", "{0:d}") %>'
                                                Columns="8" MaxLength="10" TextMode="SingleLine"
Font-Names="Arial, Helvetica" />
                                            <asp:CalendarExtender
ID="SourcePubDateEditCalendarExtender" runat="server"

TargetControlID="SourcePubDateEditTextbox"></asp:CalendarExtender>
                                        </EditItemTemplate>
                                        <InsertItemTemplate>
                                            <asp:TextBox ID="SourcePubDateNewTextbox" runat="server"
Text='<%# Bind("SourcePubDate", "{0:d}") %>'
                                                Columns="8" MaxLength="10" TextMode="SingleLine"
Font-Names="Arial, Helvetica" />
                                            <asp:CalendarExtender
ID="SourcePubDateNewCalendarExtender" runat="server"

TargetControlID="SourcePubDateNewTextbox"></asp:CalendarExtender>
                                        </InsertItemTemplate>
                                    </asp:TemplateField>
                                    <asp:TemplateField HeaderText="ContentStatus">
                                        <ItemTemplate>
                                            <asp:Label ID="ContentStatusLabel" runat="server"
Text='<%# Eval("ContentStatus") %>' />
                                        </ItemTemplate>
                                        <EditItemTemplate>
                                            <asp:DropDownList ID="ContentStatusEditDropDown"
runat="server"
                                                DataSourceID="NewsItemDetailsDS"
DataValueField="ContentStatus" DataTextField="ContentStatus"
                                                OnDataBinding="ContentStatusEditDropDown_DataBinding"
/>
                                        </EditItemTemplate>
                                        <InsertItemTemplate>
                                            <asp:DropDownList ID="ContentStatusNewDropDown"
runat="server"
                                                DataSourceID="NewsItemDetailsDS"
DataValueField="ContentStatus"
                                                OnDataBinding="ContentStatusEditDropDown_DataBinding"
/>
                                        </InsertItemTemplate>
                                    </asp:TemplateField>
```

```
                        <asp:TemplateField HeaderText="Keywords" ItemStyle-
VerticalAlign="Top">
                            <ItemTemplate>
                                <asp:Label ID="KeywordsLabel" runat="server" Text='<%#
Eval("Keywords") %>' />
                            </ItemTemplate>
                            <EditItemTemplate>
                                <asp:TextBox ID="KeywordsEditTextbox" runat="server"
Text='<%# Bind("Keywords") %>'
                                    Columns="100" TextMode="MultiLine" onkeypress="return
this.value.length<=399"
                                    Rows="3" Font-Names="Arial, Helvetica" />
                            </EditItemTemplate>
                            <InsertItemTemplate>
                                <asp:TextBox ID="KeywordsEditTextbox" runat="server"
Text='<%# Bind("Keywords") %>'
                                    TextMode="MultiLine" onkeypress="return
this.value.length<=399" Columns="100"
                                    Rows="3" Font-Names="Arial, Helvetica" />
                            </InsertItemTemplate>
                        </asp:TemplateField>
                        <asp:TemplateField HeaderText="ItemOrder" ItemStyle-
VerticalAlign="Top">
                            <ItemTemplate>
                                <asp:Label ID="ItemOrderLabel" runat="server" Text='<%#
Eval("ItemOrder") %>' />
                            </ItemTemplate>
                            <EditItemTemplate>
                                <asp:TextBox ID="ItemOrderEditTextbox" runat="server"
Text='<%# Bind("ItemOrder") %>'
                                    TextMode="SingleLine" MaxLength="4" Columns="3"
Font-Names="Arial, Helvetica" />
                                <asp:MaskedEditExtender
ID="ItemOrderEditMaskedEditExtender" runat="server"
                                    TargetControlID="ItemOrderEditTextbox" Mask="9999"
MaskType="Number" InputDirection="RightToLeft" />
                            </EditItemTemplate>
                            <InsertItemTemplate>
                                <asp:TextBox ID="ItemOrderNewTextbox" runat="server"
Text='<%# Bind("ItemOrder") %>'
                                    TextMode="SingleLine" MaxLength="4" Columns="3" Font-
Names="Arial, Helvetica" />
                                <asp:MaskedEditExtender
ID="ItemOrderNewMaskedEditExtender" runat="server"
                                    TargetControlID="ItemOrderNewTextbox"
MaskType="Number" Mask="9999" InputDirection="RightToLeft" />
                            </InsertItemTemplate>
                        </asp:TemplateField>
                        <asp:BoundField DataField="CreateDate" DataFormatString="{0:d}"
HeaderText="CreateDate" SortExpression="CreateDate" ReadOnly="true" InsertVisible="false"
/>
                        <asp:BoundField DataField="PubToSiteDate"
DataFormatString="{0:d}" HeaderText="PubToSiteDate" SortExpression="PubToSiteDate"
ReadOnly="true" InsertVisible="false" />
                        <asp:BoundField DataField="Views" HeaderText="Views"
SortExpression="Views" ReadOnly="true" InsertVisible="false" />
                    </Fields>
```

```
                    <FooterStyle BackColor="#5D7B9D" ForeColor="White" Font-Bold="True"
/>
                    <HeaderStyle BackColor="#5D7B9D" Font-Bold="True" ForeColor="White"
/>
                    <PagerStyle BackColor="#284775" ForeColor="White"
HorizontalAlign="Center" />
                    <RowStyle BackColor="#F7F6F3" ForeColor="#333333" />

                </asp:DetailsView>
            </td>
        </tr>
    </table>
</asp:Content>
```

NewsAdmin.aspx.cs

Code-behind for NewsAdmin.aspx

```csharp
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.Web.Routing;   // Might not be needed; only if linking to other pages
using WellKitEngine.Site;
using WellKitEngine.Site.News;

namespace WellKitOrg.Site.Admin.News
{
    public partial class NewsAdmin : System.Web.UI.Page
    {
        protected void Page_Init(object sender, EventArgs e)
        {
            if (!Request.IsSecureConnection)
                SiteUtility.ForceSSL();

            if (!User.Identity.IsAuthenticated)
            {
                SiteUtility.RegisterRoutes(RouteTable.Routes);

                //ErrorUtility.AddLogItem("Admin.Page_Init: User is not authenticated!",
"");
                Response.Redirect(GetRouteUrl("AccountLoginRoute", null));
            }
            if (!SiteUtility.Account.IsInRole("Administrators"))
            {
                //ErrorUtility.AddLogItem("Admin.Page_Init: User is not an
Administrator!", "");
                Response.Redirect("/home");
            }
        }

        protected void Page_Load(object sender, EventArgs e)
        {
            if (!IsPostBack)
```

```csharp
        {
            Page.Title = "Admin: Edit News Articles";
            Label lbl = ((WellKitAdminMaster)Master).FindControl("lblTitle") as
Label;

            if (lbl != null)
                lbl.Text = Page.Title;
        }

        NewsItemList nl = NewsUtility.GetNewsList();
        NewsListGridView.DataSource = nl;
        NewsListGridView.DataBind();

    }

    protected void ContentStatusEditDropDown_DataBinding(object sender, EventArgs e)
    {
        DropDownList dl =
NewsItemDetailsView.FindControl("ContentStatusEditDropDown") as DropDownList;

        foreach (ContentStatus dlItem in Enum.GetValues(typeof(ContentStatus)))
        {
            dl.Items.Add(new ListItem(dlItem.ToString(),
Convert.ToInt32(dlItem).ToString()));
        }
    }
}
}
```

Screenshots
User Interface

Interface

News page

WellKit

THE PATIENT'S ALLY

Sign In | Sign Up

HOME   BLOGS ▸   RESOURCES ▸   ACCOUNTS ▸

Concerns About Dementia Screening

Tthe push for ever-earlier dementia screening raises troubling questions for patients and their families

July 29, 2013 | The New York Times

When the diagnosis is early Alzheimer's disease, the medical profession has little treatment to offer. This month, researchers at an Alzheimer's Association conference in Boston urged policy makers to think hard before recommending wider dementia screening, saying studies have found no evidence that early detection improves outcomes.

And the experts said that little, if anything, is known about the potential risks of early detection. The diagnosis may cause stress, anxiety, depression and even suicide in patients .

Read more:

▸ Original article

OTHER NEWS

Facebook is a Bummer Study Finds

Facebook users wound up feeling worse about themselves after two weeks.

Could Dietary Tweaks Ease Type 1 Diabetes?

Eating foods that contain certain nutrients may help people with newly diagnosed type 1 diabetes continue producing some insulin for as long...

Smart Technology May Help Kids With Autism Learn, Communicate

Smart technologies such as the iPad and iPod may serve as therapeutic tools for a very specific group: children with autism

NEWS ARCHIVE

TERMS OF USE   ABOUT US   CONTACT US   FAQ   PRIVACY POLICY

News archive page

WellKit
THE PATIENT'S ALLY

HOME   BLOGS ▸   RESOURCES ▸   ACCOUNTS ▸

News Archive

CDC: Lyme disease sickens 10 times more people than are annually reported
As many as 300,000 Americans are actually diagnosed with Lyme disease each year, the Centers for...

Google Trends data highlights medical and healthcare app growth
Trends in internet keyword searches over time monitor the growth (and waning) of products and to...

Self-monitoring tied to lower blood pressure
People with hypertension who regularly monitor their own blood pressure tend to have lower numbe...

The quantified patient
Boston hospitals show the future of data-driven remote care

Global to Local Diabetes Management Program Launched
Program provides smartphones and support services to help low-income Puget Sound residents suffe...

Alabama state health department releases mobile app
The Alabama Department of Public Health has become possibly the first state health department to...

Ten biggest digital health investments in 2013 so far
Halfway through the year, there have already been some major funding stories for mobile health c...

[Previous]   [**Next**]   [Latest News]

TERMS OF USE   ABOUT US   CONTACT US   FAQ   PRIVACY POLICY

News Admin page - select mode

WellKit
THE PATIENT'S ALLY

Admin: Edit News Articles

Home
Admin

	Headline	PubToSite
Select	Global to Local Diabetes ...	8/16/2013
Select	Google Trends data highli...	8/19/2013
Select	Smart Technology May Help...	8/20/2013
Select	Could Dietary Tweaks Ease...	8/20/2013
Select	CDC: Lyme disease sickens...	8/20/2013
Select	Alabama state health depa...	8/7/2013
Select	Facebook is a Bummer Stud...	8/22/2013
Select	Ten biggest digital healt...	8/4/2013
Select	10 trends to shape mHealt...	7/29/2013
Select	Concerns About Dementia S...	8/23/2013
Select	The quantified patient	8/17/2013
Select	Self-monitoring tied to I...	8/18/2013
Select	Filler Headline 1 - Updat...	8/1/2013
Select	Filler Headline 2	8/1/2013
Select	Filler Headline 3	8/1/2013
Select	Filler Headline 4	8/1/2013
Select	No Headline	1/1/2013

Edit   Delete   New

ID	6
Headline	CDC: Lyme disease sickens 10 times more people than are annually reported
Subhead	As many as 300,000 Americans are actually diagnosed with Lyme disease each year, the Centers for Disease Control and Prevention announced.
Body	Usually, only 20,000 to 30,000 illnesses are reported each year. For many years, CDC officials have known that many doctors don't report every case and that the true count was probably much higher. The new figure is the CDC's most comprehensive attempt at a better estimate. The number comes from a survey of seven national laboratories, a national patient survey and a review of insurance information.
SourceName	Washington Post
Author	Associated Press
SourceURL	http://www.washingtonpost.com/national/health-science/cdc-lyme-disease-sickens-300000-a-year-10-times-more-people-that-are-annually-reported/2013/08/19/917e5b4c-08e2-11e3-89fe-abb4a5067014_story.html
SourcePubDate	8/19/2013
ContentStatus	Released
Keywords	Lyme disease
ItemOrder	3
CreateDate	8/19/2013
PubToSiteDate	8/20/2013
Views	0

News Admin page - edit mode

WellKit
THE PATIENT'S ALLY

Admin: Edit News Articles

Home
Admin

	Headline	PubToSite
Select	Global to Local Diabetes ...	8/16/2013
Select	Google Trends data highli...	8/19/2013
Select	Smart Technology May Help...	8/20/2013
Select	Could Dietary Tweaks Ease...	8/20/2013
Select	CDC: Lyme disease sickens...	8/20/2013
Select	Alabama state health depa...	8/7/2013
Select	Facebook is a Bummer Stud...	8/22/2013
Select	Ten biggest digital healt...	8/4/2013
Select	10 trends to shape mHealt...	7/29/2013
Select	Concerns About Dementia S...	8/23/2013
Select	The quantified patient	8/17/2013
Select	Self-monitoring tied to I...	8/18/2013
Select	Filler Headline 1 - Updat...	8/1/2013
Select	Filler Headline 2	8/1/2013
Select	Filler Headline 3	8/1/2013
Select	Filler Headline 4	8/1/2013
Select	No Headline	1/1/2013

Update   Cancel

ID	6
Headline	CDC: Lyme disease sickens 10 times more people than are annually reported
Subhead	As many as 300,000 Americans are actually diagnosed with Lyme disease each year, the Centers for Disease Control and Prevention announced.
Body	**B** *I* ≣ ≣ ≣ ≣ ↺ ↻ Usually, only 20,000 to 30,000 illnesses are reported each year. For many years, CDC officials have known that many doctors don't report every case and that the true count was probably much higher. The new figure is the CDC's most comprehensive attempt at a better estimate. The number comes from a survey of seven national laboratories, a national patient survey and a review of insurance information.
SourceName	Washington Post
Author	Associated Press
SourceURL	http://www.washingtonpost.com/national/health-science/cdc-lyme-disease-sickens-300000-a-year-10-times-more-people-that-are-annually-reported/2013/08/19/917e5b4c-08e2-11e3-89fe-
SourcePubDate	8/19/2013
ContentStatus	Released
Keywords	Lyme disease
ItemOrder	3
CreateDate	8/19/2013
PubToSiteDate	8/20/2013
Views	0