



EPiC Series in Computing

Volume 63, 2019, Pages 221–230

Proceedings of 32nd International Conference on  
Computer Applications in Industry and Engineering



# The Design and Implementation of a Cloud Based Presentation System

Abhijeet V Ubale and Jiang Guo \*

California State University, Los Angeles, USA

abhijeet.ubale@gmail.com, jguo@calstatela.edu

## Abstract

Internet's influence has been growing at a rapid speed, and so has been the demand for web based application. The migration from stand-alone to platform independent application offers benefits in terms of maintainability, scalability and ease of deployment.

Web Based Presentation System also referred to be as WBPS is an attempt to minimize the dependence on stand-alone applications, and additionally provide a comprehensive browser based solution which administers users, schedules presentations and provide a platform independent tool for presentations.

In the past, faculties have been required to manually schedule presentations for the students using the calendar, and publishing the schedule calendar. Students were to give their presentation using presentation tools as Microsoft PowerPoint, Dyknow and brinkpad.com.

With WBPS, faculties have the ability to allow the system to auto schedule presentation, and publish the presentation schedule. Students can upload their presentations, and use an online presentation tool for giving their presentation. The faculties have a choice to choose between manual scheduling verses automatic scheduling. Besides being an online tool for giving presentations, the WBPS have features like Member Management, Past and Presentation Templates Repository, Automated Reminders and taking Notes or making annotations while attending a presentation.

## 1 Introduction

The faculties at the department of computer science often setup presentations for students. These presentations assist in advancing the knowledge of the assigned topic; and public speaking abilities of the students.

---

\* Corresponding author: Jiang Guo, jguo@calstatela.edu

Faculties are required to form groups, coordinate amongst these groups, assign topics and communicate the schedule back to the students. As the number of students, group size and public holidays increase, the scheduling and organizing gets more complicated. The entire activity of scheduling presentation is tedious and repetitive every quarter. Each faculty follows their own approach to scheduling and announcing the presentation schedules.

For giving the presentations, students rely upon presentation software tools as PowerPoint and DyKnow. These presentation tools are standalone applications that need to be installed on the client computers. PowerPoint and DyKnow are Windows specific. Non-Windows users are required to install virtual machine software as VM Player or Parallels to install Windows, and give a presentation. The problem further aggravates when non-windows users have developed a program that is on their Mac or Linux environment, while their presentation is on the virtual machine environment.

Occasionally, DyKnow installation on the client computer causes driver compatibility issues with installed display drivers. These drivers are required for the inking and annotations. Also, DyKnow requires that a full version of the .NET Framework be installed on the client computer. DyKnow and PowerPoint provide limited annotating abilities to highlighting and underlining, do not provide common features as taking notes or placing sticky notes. While, brinkpad.com is an online tool for presentations, it is not an end-to-end solution for giving presentations. Also, brinkpad does not save annotations on the slides, but rather a separate annotations file.

Overall, a comprehensive and consistent approach that will allow faculties to schedule, announce schedule, administer the users, a platform independent presentation tool and central repository is missing.

Web Based Presentation System is a comprehensive web based tool that can address the problems of scheduling complexity and dependence on standalone tools. The solution is an attempt to be the one stop shop for the presentation management needs. The solution uses a browser based presentation tool that has a rich set of annotation tools like Rectangle, Ellipse, Line, Arrow, FreeHand and Sticky Notes. The presentation tool also works with pointing devices as the pen, which makes it very easy for the presenter to make freehand annotations. The tool is also equipped to capture the notes of other fellow students and faculties. As the presentation tool is browser based, the dependency on tools as PowerPoint and DyKnow is minimized.

Here is a synopsis to the features –

- **Automatic Scheduling** – gives the ability to the faculty to auto schedule the presentations for the students. The automatic scheduling feature is well suited for scheduling large number of presentations. It works by taking several input parameters as Start and End Date/Time of the course, Public Holidays, Days of the Week, Group Size, and Presentation Time etc. Based upon these input parameters, the system allocates course topics to groups.
- **Presentation Tool** – is the browser based replacement of PowerPoint and DyKnow. The tool is equipped with a rich set of annotation toolset that assists students to make the presentation. As the tool is browser based and platform independent, the students can easily switch back and forth to their technical solutions for presentation references. Thereby, eliminating the need to install a virtual machine environment especially for giving the presentation. Annotations can be made using various geometric shapes like rectangle, ellipse, circle and line. Special annotations like arrow, sticky notes and free hand, and taking notes is possible too. As the presentation tool supports pointing devices as the Pen, it becomes easier for the presenter to use the Free Hand annotation. Annotations and Notes are saved as a personal copy of the user and can be accessed by the user for reference in future.
- **Presentation Repository** – acts as the central repository for all the past presentations. The presenter does not need to distribute the presentation slides to fellow students and faculty. The original presentation and annotated slides are available for download from an application page. As the original or annotated slides are in XPS format, these can be viewed on the Linux

and Mac environments as well. The feature is particularly helpful to refer to the notes that were taken during the presentation by the student or faculty. For consistency, often certain courses require the use of a specific template for the presentation. These templates are available via the presentation templates repository.

- **Reminders** – is the feature of the solution that sends a reminder to the presenters who have upcoming presentations. For this, the solution has a back-end scheduled job that runs from a scheduler such as the Windows Tasks. The feature sends reminder emails to the presenters five and two days prior to the scheduled presentation date.

## 2 WBPS Architecture

No matter the type of application being developed, whether it is an ASP.NET Website, WCF Services or Silverlight rich internet application the logical grouping of the software components involved in the design is called Layer. Each layer is tasked with a specific responsibility in the design of the solution. For e.g. the Data Access Layer is responsible for the storage and retrieval of the data. This segregation of the design into layers (software components) helps with reusability and testability of the components.

Web Based Presentation System has been developed with the Layered architecture. As the Silverlight framework is a subset of the .NET Base Class Libraries, Silverlight applications cannot directly utilize regular .NET code. To address these incompatibility issues between Silverlight and regular .NET Code, and the presence of two independent presentation layers, an additional layer namely the Services Layer has been introduced. The Services Layer acts as a conduit between the Business Logic Layer and the Presentation Layer.

Usage of the layered application architecture provides the following benefits:

- As the application layers are loosely coupled and highly cohesive the maintenance of the application is easier.
- The ability to switch between varying implementations of the software component.
- The development effort can be distributed amongst team developers. The work can be distributed at layer level.
- It is easier to test independent layers of the architecture.

In a layered architecture, each layer is dependent upon the lower layer, and provides certain services to above layer. The layer is unaware about the internal implementation of the functionality. The layered architecture can thought of as a set of co-operating software components grouped into layers [6]. The division of an application into separate layers maximizes the maintainability of the code, and optimizes the application's functioning in different deployment scenarios.

In certain cases, the data access layers and business logic layer may be located on the same physical tier as the database server, while the services layer resides on a separate physical tier. In such scenarios, the design must accommodate this. However, with Web Based Presentation System, all the layers are deployed on the same physical tier.

### 2.1 Layers

The system is divided into several layers.

- **Presentation Layer** – The layer implements the components that implement and display the user interface and manage user interaction with the system. The layer contains web pages, custom user controls, controls and Silverlight canvases for user input and displaying data.

Each visual component contains a number of fields that display output from business logic layer and collects user input. The presentation layer includes the following:

- Visual Elements – This includes all the visual elements in the Web Based Presentation System i.e. Master Pages, Web Pages, Custom User Control and Silverlight Canvases.
- Presentation Logic – is the logic that defines the logical behavior and the structure of the application that is independent of any specific user interface implementation. This includes MVVM, MVC or the MVP patterns. In Web Based Presentation System, the Simple Web Forms with Master Pages and code-behind has been implemented.
- Services Layer – In Web Based Presentation the Services Layer provides the business functionality through the services layer to the ASP.NET website and the Silverlight 4 application. The Services Layer is the conduit between the Presentation Layer and Business Logic Layer. The Services Layer exposes an Interface to which all the messages are sent. In Web Based Presentation System, the Services Layer is hosted in Internet Information Services for maximum interoperability between ASP.NET and Silverlight.
- Business Logic Layer – separates the Presentation layer and the Data Access Layer. It enforces the business rules applicable to the application, and serves as an intermediary for data exchange between the services layer and the data access layer. Typically, the Business Logic Layer contains:
  - Business Logic Components – is defined as any application logic whose task is to:
    - o Retrieve, process, transform and manage application data.
    - o Apply Business Rules and Policies
    - o Ensuring Data Consistency and Validity
  - Application Façade – This is a simplified interface to the to the Business Logic components. This is frequently done by combing multiple business operations into a single operation. For the Web Based Presentation System, the Business Logic Components are tasked with retrieval, storage and application of business rules and policies. The Business components are accessed from the Services Layer. The Services layer initializes the Business Component with the Database information.
- Data Access Layer – is tasked with the storage and retrieval of data from the database. Data Access Components are responsible for storing and exposing the data to the Business Logic Layer. As the data access layer is isolated from the business logic layer, the impact of change in the DBMS is minimized. This also simplifies the maintenance as all the code that manipulates a particular data item is in one central place. The Data Access Layer includes:
  - Data Access Components – abstract the logic required for accessing the underlying data stores. With Entity Framework such logic is already implemented, thereby reducing the amount of data access code.
  - Service Agents – Business Components can retrieve data provided by an external service, by implementing the code to communicate with the external service.

The Web Based Presentation system employs the Entity Framework and Self Tracking Entities to achieve this. The Data Access Components of the Web Based Presentation System follow the repository design pattern.

## 2.2 Modules

Modules represent certain pieces of functionality of the application. These pieces could be interconnected. Modular programming is a concept of programming where programs are self contained. Such programs are easier to maintain, read and write. The Web Based Presentation System is divided into modules to ease the maintenance effort in future. Although the modules are interdependent, the interfaces are clear and concise which makes it easier for replacement if needed.

- Membership and Role Manager – Membership and Role Providers give a built-in way of validating and storing user credentials. [11] The membership and role providers exist to provide authentication and authorization services to Web Based Presentation System.

When a user provides the credentials on the login page, the MembershipHelper works with the Membership Provider to authenticate the user. The Membership Provider authenticates the user by using the ValidateUser method. For each authenticated user, the AuthenticationHelper creates a Session Cookie. SetAuthCookie in AuthenticationHelper creates the session cookie for the supplied username. The user's information like Name, Email is retrieved from the database, and maintained in the Session.

- Scheduling Manager – The scheduling manager module contains classes responsible for scheduling automatic or manual presentations. The classes take input from the faculty and create presentation schedules for the students.

The module works with the ScheduleHelper and AutomaticSchedulerOptions classes. The ScheduleHelper class contains methods for Automatic and Manual scheduling and selecting or unselecting presenters from a schedule. The ScheduleHelper class generates a List of Schedules which is bound on the page. The Scheduling Manager checks the role of the member by communicating with the Role Manager. If the logged on user has the role of a faculty, then the User is allowed to setup the presentations. Further, the Scheduling Manager communicates with the Membership Module to retrieve all the users that are available.

- Presentation Manager – The presentation manager module is responsible for providing access to the repository and templates. It also consists of the presentation tool, which is used for giving a presentation.

As there two independent presentation layers, the website communicates with the presentation tool through a cookie. When a student starts the presentation, the website sends a cookie to the Presentation Tool, which loads the appropriate presentation. The presentation tool contains annotation shapes that can be used for highlighting during a presentation. This is achieved by the drawing the geometric shapes on the presentation canvas. The presentation tool communicates with the database through the WCF Services Layer for saving the notes or annotated slides.

## 3 WBPS Design

### 3.1 Data Access Layer

This layer is tasked with the storage and retrieval of data for the Web Based Presentation System. The Data Access Layer has been implemented using Entity Framework 4, Self Tracking Entities and T4 Templates for the Data Access Components. Entity Framework is Microsoft's Object Relational Mapping solution to solve the impedance mismatch problem. Entity Framework allows to work with data in the form of objects and properties without having the concerns of:

- Retrieval of data from the underlying database tables, views and stored procedures.
- Storage of data to the tables

The Data Access Classes inherit from the Ef4DaoBase class, which contains functionality that is common across the Data Access Classes. The Ef4DaoBase takes an Entity Data Model and Entity Type as a Type parameter. The Data Access Classes are generated using T4 templates:

- **WBPS.tt** – Generates Self Tracking Entities. [7] A file for each entity type and complex type defined on the conceptual model with the `DataContract` and `DataMember` attributes are generated by the T4 Templates.
- **WBPS.Context.tt** – Generates `ObjectSet` Properties, Function Import methods for all the Entities that are defined on the conceptual model.
- **WBPSDao.tt** – Generates Data Access Components. A custom T4 Template has been written that generates a Data Access Component based on the `Ef4DaoBase`. The T4 Template generates `<entity or complex type name>Dao.cs` and `<entity or complex type name>Dao.Extension.cs`. The `<entity or complex type name>Dao.Extension.cs` is useful for writing custom methods, as this file is not overwritten on every modification of the Entity Model.

The Data Access Classes are based on the Repository Pattern, with maximum code written in the `Ef4DaoBase` class.

## 3.2 Business Logic Layer

This layer contains Business Object classes which enforce Business rules of the application. The Business Object classes are accessed from the Services Layer.

All the Business Object classes inherit from `BoBase` class, that takes a Collection of Key Value Pair. The collection contains information as Connection String, Command Timeout for the Dao Classes. As there is no requirement today, which requires that the Business Object apply any business rules, the Business object simply calls the Dao method, and passes the results back to the Services Layer.

Considering the number of operations that the Business Logic Layer is performing, the Business Logic Layer has independent Business Logic Components instead of the Application Façade.

## 3.3 Services Layer

This layer contains interfaces between the Business Logic layer and the Presentation layer. In most cases, the services instantiate the Business Object, and perform the appropriate action.

Web Based Presentation System contains one service with multiple service methods that are invoked by the presentation layer. The services have been configured with `WsHttpBinding` and `basicHttpBinding` to allow maximum interoperability with other systems. The end-points can be configured through the `web.config` at any given point without changing any code on the services layer.

The Services Layer contains all the configuration like Connection Strings, Service End-Point configuration in the `web.config`. The configuration information is read by the `GetConfiguration` method.

## 3.4 Presentation Layer

The Web Based Presentation System has two presentation layers namely the ASP.NET Website and the Silverlight Presentation Tool. The solution contains two independent presentation projects to cater to this requirement.

## 3.5 ASP.NET Presentation Layer

This layer contains the user interfaces that are displayed to the user. This presentation projects contains the ASP.NET master pages, web pages, custom user controls, style sheets and scripts that the user can access data through.

### 3.6 Silverlight Presentation Layer

The Silverlight presentation layer contains the presenter tool that is used for giving the presentation. This presentation layer is deployed in the ClientBin folder of the ASP.NET web site and is accessed by the Presenter.aspx page.

The tool has abilities to allow the user to make annotations and take notes. All the annotations inherit from the AnnotationBase, that contains all the common functionality. The Annotation classes such as LineAnnotation are responsible for drawing their respective shapes. The annotation class does this by handling MouseDown, MouseMove and MouseUp events.

The tool also has the ability to take notes. The notes can be taken by the Faculty or the student. The notes can be saved by invoking the SaveNotesAndAnnotations service method of the Web Service.

### 3.7 Backend Reminder Job

This layer contains the implementation for sending reminders to the presenters five and two days ahead of scheduled presentation date. The job is invoked by a Windows scheduler every midnight. The job communicates with the services layer and fetches all the presentations that are due in the next two or five days, and sends an email reminder with the schedule and topic. The Smtplib details and service details can be configured through the app.config file.

## 4 Implementation

Web Based Presentation System is a protected website, and hence requires the user to login into the system. After the user is authenticated, the user is redirected to the Home Page that displays the applicable Menu options for the user. The Menu Items are displayed as per the role(s) of the User. An authenticated user can maintain his/her profile by accessing the My Profile link. As a non-administrative user, the user can only update their demographic details. A user with administrative privileges can also update the role, approval or locked status.

A user with faculty privileges has the ability to schedule presentations for the students (See Figure 1). For this, the faculty can click on Schedule Presentations from the Navigation Menu. Setting up presentations is a two step process. Step 1 includes selecting the course, and students. Step 2 includes assigning and scheduling course topics. After a faculty has scheduled presentations for students, the student can start their presentation by clicking on Start Your Presentation from the Navigation Menu. As the student may have multiple presentations scheduled, the screen prompts for the course topic.

The screenshot shows the 'Faculty Page' for the 'Web Based Presentation System' at California State University, Los Angeles. The page is titled 'SCHEDULE PRESENTATIONS' and is in the 'ASSIGN TOPICS TO PRESENTERS (STEP 2 OF 2)' phase. The user is logged in as 'Jiang Guo'.

**COMMON TASKS:** My Profile, Sign Out

**FACULTY TASKS:** Schedule Presentations

**ADMINISTRATIVE TASKS:** Manage Users

**PRESENTATIONS:** Start Your Presentation, Past Presentations, Active Presentations, Templates

**DEPARTMENT LINKS:** Computer Science Department, CSNS, ABET

**UNIVERSITY:** Academic Calendar, Add/Drop Dates, Catalog, Class schedule, Final Exam Schedule, GET

**ASSIGN TOPICS TO PRESENTERS (STEP 2 OF 2)**

Choose the scheduling option:  
 Manual  Automatic

**SELECTED PRESENTERS**

Name	Email
<input type="checkbox"/> Christopher Moore	christopher.moore@gmail.com
<input type="checkbox"/> Debra Swanson	debra.swanson@gmail.com
<input type="checkbox"/> Ameya Ubale	ameyaubale@gmail.com

**Topic Schedule**

Topic	Schedule
<b>MATLAB</b> Presenters: Abhijeet Ubale, Allan Russell, Christie Huff on Monday, May 30, 2011 9:00 AM for 45minute(s).	Date & Duration (Mins) [ ] [ ] <a href="#">Cancel Course Topic Assignment</a>
<b>Temporal Logical</b> Presenters:	Date & Duration (Mins) [ ] [ ] <a href="#">Assign Course Topic</a>
<b>Message Driven Software Architecture</b> Presenters:	Date & Duration (Mins) [ ] [ ] <a href="#">Assign Course Topic</a>
<b>Model Driven Development</b> Presenters:	Date & Duration (Mins) [ ] [ ] <a href="#">Assign Course Topic</a>
<b>Temporal Logical in Software Development</b> Presenters:	Date & Duration (Mins) [ ] [ ] <a href="#">Assign Course Topic</a>
<b>Wireless Sensor and Wireless Sensor Network</b>	Date & Duration (Mins)

Figure 1: Faculty Page

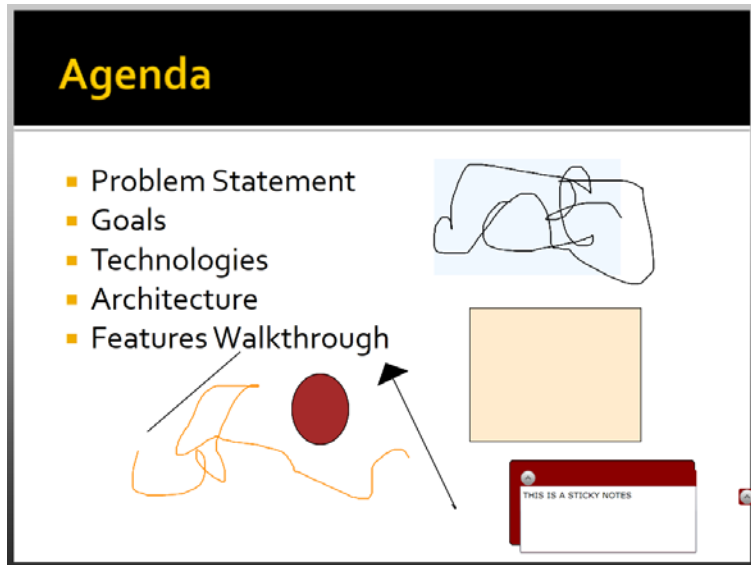
The student can select the PowerPoint presentation, and click Start Presentation to give a presentation. The presenter tool receives a Cookie with the presentation id. The Silverlight control retrieves the presentation information based upon the presentation id. Further the slides are converted into XAML canvas which is understood by Silverlight. The presenter tool starts the presentation. The tool has features to make annotations and take notes.

The navigation bar also contains the Save icon that can be used to save the annotations and notes made by the user.

The user's annotations and notes are saved to the database through the WCF Services. For this, Silverlight asynchronously invokes the SaveNotesAndAnnotations method. The method internally creates an annotated Xps file that contains the original slides with super imposed annotations. This newly created Xps file is saved into the users' past presentation session. Similarly, the notes that the user has taken during the presentation session are saved to the users' past presentation session.

The user can draw Rectangle, Ellipse, Line, Arrow, FreeHand and StickyNotes on the slides. [12] Silverlight's ink and geometric shapes drawing support has been helpful for the rectangle, line and ellipse annotations. Sticky Note annotation is a combination of the Rounded Rectangle, Silverlight Expander and a TextBox. Arrow Annotation is a combination of the Polygon and a Line, and freehand annotation is a series of small lines that are formed by the dragging mouse. Figure 2 shows a sample slide with the annotations.





**Figure 2:** Sample Annotations Screen

The user can also take notes while the presentation is in progress. The notes and annotated slides are available through the Past Presentations link for future reference.

- Rectangle Annotation – is used to draw rectangles on the slides. This is achieved by using the geometric Rectangle shape of Silverlight. Stroke Thickness, Stroke Color and Fill Color can be used for drawing a rectangle.
- Ellipse Annotation – is used to draw ellipses on the slides. This is achieved by using the geometric Ellipse shape of Silverlight. Stroke Thickness, Stroke Color and Fill Color can be used for drawing an ellipse.
- Line Annotation – is used to draw lines on the slides. This is achieved by using the geometric Line shape of Silverlight. Stroke Thickness and Stroke Color can be used for drawing a line. Fill Color is not applicable for drawing the line.
- Arrow Annotation – is used to draw arrows on the slides. This is achieved by using the geometric Line and Polygon shape of Silverlight. Stroke Thickness, Stroke Color and Fill Color can be used for drawing an arrow.
- FreeHand Annotation – is used to make freehand drawings on the slides. This is achieved by using the geometric Line shape of Silverlight. Stroke Thickness and Stroke Color can be used for making freehand annotations. The FreeHand annotation deviates in the implementation of the MouseMove event. In the implementation, small lines are drawn on the canvas.
- Sticky Note Annotation – is used to make sticky notes on the slides. This is achieved by using the geometric Rectangle, TextBox and Expander of Silverlight. Stroke Thickness, Stroke Color and Fill Color can be used for making sticky notes.
- Zoom Control – is used to zoom in and out of the slides. By using the Reset Zoom button, it is possible to revert to the original zoom size.

## 5 Conclusions

The Web Based Presentation System is a comprehensive presentation tool that is simple to use. Further, the entire application is browser based thereby minimizing the dependency on software tools like DyKnow, PowerPoint for presenting.

Scheduling, Online Calendar and reminders of presentations make it easier for the students and faculties to organize and avoid confusion. Features to view past presentations and templates repository are helpful. Membership and Role providers abstract out the authentication and authorization from the application, making it easier and more maintainable.

An important benefit of the application is that it tries to automate process of scheduling and giving presentations.

Most important is its ability of online notation.

## Acknowledgements

The work was supported in part by the NASA under grant NNX15AQ06A.

## References

- [1] Microsoft .NET Framework Overview - <http://msdn.microsoft.com/en-us/library/zw4w595w.aspx>
- [2] Programming Microsoft ASP.NET 4 - Dino Esposito
- [3] Professional Silverlight 4 - Jason Beres, Bill Evjen and Devin Rader
- [4] Microsoft SQL Server 2008 Express Edition - [http://en.wikipedia.org/wiki/Microsoft\\_SQL\\_Server](http://en.wikipedia.org/wiki/Microsoft_SQL_Server)
- [5] Silverlight implementation on Linux - <http://www.mono-project.com/Moonlight>
- [6] Microsoft Application Architecture Guide, 2nd Edition - <http://msdn.microsoft.com/en-us/library/ff650706.aspx>
- [7] ADO.NET Self-Tracking Entity Generator Template - <http://msdn.microsoft.com/en-us/library/ff477604.aspx>
- [8] T4 Templates - <http://www.olegrych.com/2007/12/text-template-transformation-toolkit/>
- [9] Professional .NET Framework 2.0 - Joe Duffy
- [10] Entity Framework 4.0 Recipes: A Problem-Solution Approach - Larry Tenny and Zeeshan Hirani
- [11] Membership and Role Provider - <http://msdn.microsoft.com/en-us/library/yh26yfzy.aspx>
- [12] Silverlight Drawing Tools - [http://msdn.microsoft.com/en-us/library/cc189073\(VS.95\).aspx](http://msdn.microsoft.com/en-us/library/cc189073(VS.95).aspx)