

Санкт-Петербургский Государственный Университет

Математико-механический университет

Кафедра системного программирования

Публикация документов в SharePoint 2010

Курсовая работа студента 361 группы

Рябиченко Павла Николаевича

Научный руководитель Г.Ф. Дейкало

Санкт-Петербург

2010

Оглавление

Введение.....	3
Постановка задачи.....	4
Существующие подходы к публикации документов.....	5
Реализация публикации документов	6
Работа с SharePoint 2010	6
Описание проекта и работы веб-частей	7
Примеры кода	11
Результаты работы	12
Дальнейшие варианты развития.....	12
Список литературы.....	13

Введение

В последнее время в компаниях и организациях появляются все более и более функциональные корпоративные порталы. Раньше это были внутренние сайты, на которых публиковались новости и материалы для сотрудников, имелась база файлов и документов для быстрого доступа, а также какой-нибудь форум для общения. Со временем появились развитые системы для совместной работы и платформы для интеграции с корпоративными приложениями.

В настоящее время наиболее популярными решениями для корпоративных порталов являются (IntranetNo.Ru):

- Microsoft SharePoint
- 1С-Битрикс: Корпоративный портал
- Affari
- IBM WebSphere Portal

На математико-механическом факультете по адресу <http://info.math.spbu.ru/> (или <http://info> для внутреннего доступа) имеется внутренний портал, работающий на базе SharePoint 2010. В данный момент портал используется для нужд ИТ службы факультета и находится в тестовом режиме.

SharePoint 2010 – это платформа для совместной работы, которая обеспечивает увеличение производительности труда и предоставляет удобное управление контентом в знакомой среде Office.

SharePoint может быть использован для создания сайтов, предоставляющих пользователям возможность для совместной работы. Создаваемые на платформе SharePoint сайты могут быть использованы в качестве хранилища информации, знаний и документов, а также использоваться для исполнения облегчающих взаимодействие веб-приложений, таких как wiki и блоги. Пользователи могут управлять и взаимодействовать с информацией в списках и библиотеках документов, используя контролы, называемые веб-части (SharePoint WebParts)

Постановка задачи

В рамках данной курсовой реализуется публикация документов на базе SharePoint 2010 с учетом интересов пользователей и разграничения прав доступа на просмотр документов.

В данной работе были поставлены следующие задачи:

1. Ознакомиться с методами разработки Web-частей, используя технологию ASP.Net
2. Подготовить и настроить тестовое окружение для удаленной разработки, чтобы не мешать работе самого портала.
3. Разработать веб-части для публикации документов.
4. Используя информацию из Active Directory, разграничить права доступа к документам.

Существующие подходы к публикации документов

- SharePoint Server 2010 с корпоративной лицензией клиентского доступа предоставляет интеграцию для документов Microsoft Office, ранее был известен как Microsoft Office SharePoint Server (MOSS). Он является надстройкой над бесплатной версией Microsoft SharePoint Foundation и расширяет его возможности. В таком случае появляется возможность просмотра и редактирования документов прямо в браузере.
- Веб-приложения Office Web Apps — это интернет-дополнения к приложениям Word, Excel, PowerPoint и OneNote, позволяющие пользователям получать доступ к документам из любого места. Пользователи могут просматривать документы и предоставлять к ним доступ другим, а также работать над ними совместно с другими пользователями, используя при этом персональные компьютеры, мобильные телефоны и Интернет. Office Web Apps доступны пользователям через Windows Live, а бизнес-клиентам — через программы корпоративного лицензирования Microsoft Office 2010 и решения по управлению документами, основанные на продуктах Microsoft SharePoint 2010. (Technet)
- В SharePoint 2010 имеется публикация и версионность в списках и библиотеках документов. Также имеется возможность настроить права доступа к списку.

Реализация публикации документов

Для разработки под SharePoint 2010 мне требовались следующие инструменты:

- Visual Studio 2010 (VS 2008 тоже поддерживает SharePoint 2010)
- SharePoint Designer 2010
- Средство разработчика в IE9 для отладки скриптов и веб-частей

Работа с SharePoint 2010

В Microsoft Visual Studio 2010 имеется альтернативный способ создания приложений SharePoint с помощью конструктора SharePoint. Visual Studio 2010 предоставляет возможность быстрой разработки решений в SharePoint с помощью таких компонентов, как дополнительные средства отладки, IntelliSense, завершение операторов и шаблоны проекта. Visual Studio также использует преимущества дополнительных возможностей средств и языков платформы .NET Framework. Проекты SharePoint можно разрабатывать на языке Visual Basic, C# и F#.

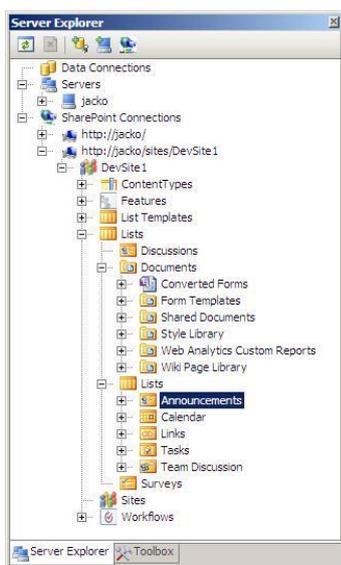


Рис. 1 SharePoint Explorer

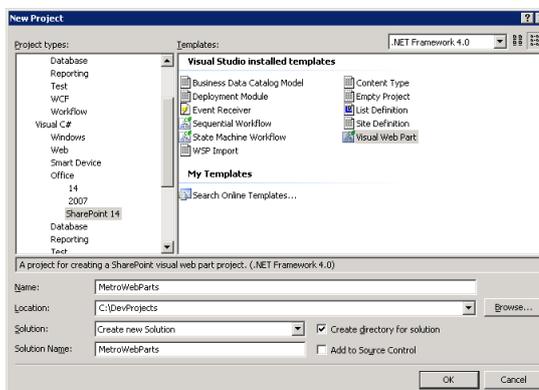


Рис. 2 New Project and Item Templates

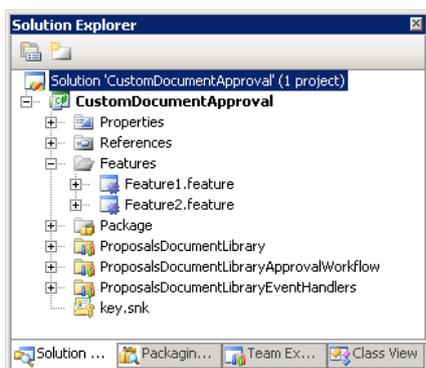


Рис. 3 New Project Structure

Веб-части в SharePoint представляют собой серверные элементы управления, которые выполняются внутри страницы специального типа, называемые страницей веб-частей. Исторически сложилось, что SharePoint поддерживает два вида веб-частей:

- Веб-части ASP.Net – наследуются от класса веб-частей из System.Web. Импортируются в *.webpart, потому их webpart'ами чаще всего и называют.
- Веб-части Windows SharePoint Services – наследуются от класса веб-частей из Microsoft.SharePoint. Импортируются в *.dwp. Используется для поддержания обратной связи с SharePoint v2.

Описание проекта и работы веб-частей

В рамках курсовой работы для выполнения поставленной задачи было выбрано следующее решение:

1. Для публикации документов создается библиотека документов по заранее созданному шаблону с полями «LinkFilename», «Published», «Category», «ID», «ListId», «Status», «PublishDate», которые используются для информации о документе. Для демонстрации на узле <http://info/sharedocs/> такая библиотека была создана вручную, но для внедрения достаточно сделать шаблон этой библиотеки документов и подключить его, тем самым автоматизируя создание.
2. Для разграничения прав доступа было решено использовать группы разрешений SharePoint, которые представляют собой набор отдельных пользователей, а также могут включать в себя группы Active Directory. На внутреннем портале факультета, на котором велась разработка, сведения о профиле пользователя берутся из доменных служб Active Directory. Потому для поставленной задачи вполне достаточно ограничиться созданием групп разрешений, с учетом всей доступной информации о пользователе. Для настройки синхронизации профилей в SharePoint 2010 можно воспользоваться статьей на сайте TechNet. (Microsoft, Настройка синхронизации профилей)
3. Информация о категориях групп разрешений хранится в списке «Category». В целях автоматизации создания категории была создана веб-часть «CreateCategory», с помощью которой заполняется информация в списке «Category» и создаются группы разрешений для пользователей на портале. Документы публикуются для какой-либо

категории, и пользователи, не вошедшие в нее, не увидят этот документ в отображении веб-части «DocViewByCategory».

4. В качестве списка тем, иначе можно сказать тегов, для документов, используется список «Sections». То есть помимо разграничения прав документы будут иметь еще и тему.

Подводя итог, получилось следующее: для начала создается список тем документов, по мере необходимости его можно редактировать, потом создаются категории, чтобы ограничить доступ на просмотр, далее можно стандартными средствами добавлять документы в библиотеку.

Настройку тем и категорий можно произвести вручную сразу. Для автоматизации этого процесса можно создавать списки с помощью заранее подготовленных шаблонов.

Проект курсовой работы состоит из нескольких веб-частей (webpart'ов), двух обработчиков событий для списков «Category» и «Sections» и отдельно вынесенных стилей для веб-частей. Списки «Category» и «Sections» используются для хранения информации о списке категорий и разделов документов, соответственно.

Для публикации документов было реализовано несколько веб-частей:

- CreateCategory
 - Создание категории группы пользователей для публикации документов, которым они будут видны на странице.
 - Категории используются для разграничения прав доступа на просмотр. При создании категории с помощью данной веб-части в список «Category» добавляется информация о созданной категории. Далее в группах разрешений SharePoint создается группа пользователей с названием «Группа_<названия созданной категории>», в которую вносится создатель категории. В данную группу можно добавить других пользователей или групп пользователей.
 - Категории с «кратким представлением» будут отображаться в веб-части DocViewByCategory

CreateCategoryWebPart

Название категории *	<input type="text"/>
Описание категории	<input type="text"/>
Раздел *	test <input type="button" value="v"/>
Порядковый номер	<input type="text"/>
Вариант отображения	<input checked="" type="radio"/> Табличное представление <input type="radio"/> Краткое представление
Отображать на главной странице	<input checked="" type="checkbox"/>
Отображаемые поля	
<input type="button" value="Добавить новую"/>	<input type="button" value="Отмена"/>

Рис. 4 CreateCategory

- AddLinkedDocs
 - Создает ссылку на документ.
- ViewLinkedDocs
 - Отображает ссылки на документы
- DocLibNavigator
 - Показывает глобальные библиотеки документов на портале

DocLibNavigation



Рис 5. DocLibNavigator

- SetCategory
 - Добавляет категории для опубликованных документов.



Рис. 6 SetCategory

- DocViewByCategory
 - Отображение опубликованных документов по темам с учетом категорий

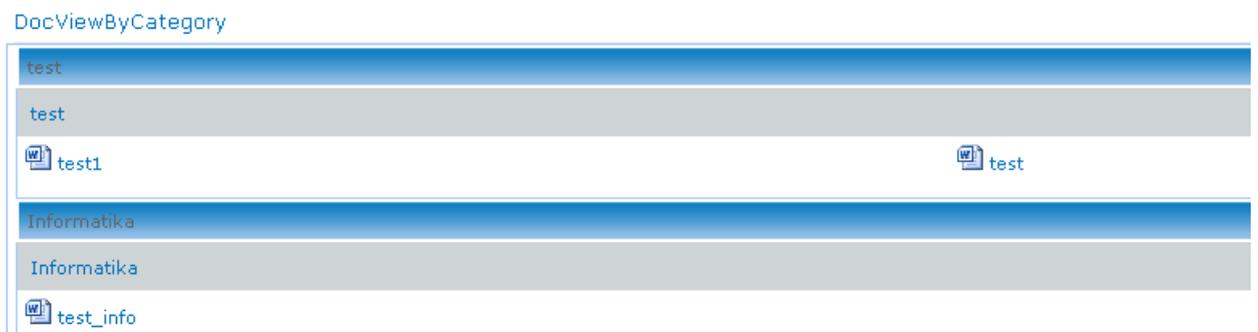


Рис. 7 DocViewByCategory

Примеры кода

Для обработки события «обновление элемента» вызывается Event Handler “CategoryHandler”. Сначала проверяется название списка (библиотеки документов), далее делается CAML-запрос к базе данных SharePoint.

В проекте имеются отдельные обработчики для библиотеки документов, списка категорий и списка разделов.

```
public override void ItemUpdating(SPItemEventProperties properties)
{
    base.ItemUpdating(properties);

    if (properties.ListTitle == "Category") {
        try {
            string buffer="";
            string newTitle = properties.AfterProperties["Title"].ToString();
            string oldTitle = properties.ListItem["Title"].ToString();
            if (oldTitle!= newTitle ) {

                SPWeb web = properties.OpenWeb();
                SPSiteDataQuery docQuery = new SPSiteDataQuery();
                docQuery.Webs = "<Webs Scope='SiteCollection'>";
                docQuery.Lists = "<Lists BaseType='1' />";
                docQuery.ViewFields = "<FieldRef Name='ID' /><FieldRef Name='LinkFilename' /> +
                    "<FieldRef Name='Published' /><FieldRef Name='Category' />";
                docQuery.Query = "<Where><And>"+
                    "<Eq><FieldRef Name='Published' /><Value Type='Boolean'>1</Value></Eq>"+
                    "<Contains><FieldRef Name='Category' /><Value Type='Text'>"+ oldTitle + "</Value></Contains>"+
                    "</And></Where>";

                DataTable dt = web.GetSiteData(docQuery);
                foreach (DataRow drow in dt.Rows)
                {
```

Рис. 8 Код обработчика событий CategoryHandler

Для вызова модального окна в веб-частях на странице используется JavaScript:

```
<script type="text/javascript">
    function OpenPublishDialog(itemId,title) {
        var options = SP.UI.$create_DialogOptions();
        options.url = '/SiteAssets/createCategory.aspx?Category=' + itemId;
        options.dialogReturnValueCallback = Function.createDelegate(null, CloseCallback);
        options.title = title;
        SP.UI.ModalDialog.showModalDialog(options);
    }
    var messageId;

    // Dialog callback
    function CloseCallback(result, target) {

    }

</script>
```

Рис. 9 Код вызова модального окна

Результаты работы

В результате работы была создана система для публикации документов на внутреннем портале математико-механического факультета с учетом информации о пользователях факультета.

Данное решение было разработано для подсайта <http://info/sharedocs/> как тестовое решение и для внедрения на персональные страницы пользователей требует доработки.

Дальнейшие варианты развития

Подводя итоги, можно выделить некоторые задачи для дальнейшего развития проекта:

- Необходимо автоматическое создание групп категории и добавление туда пользователей, основываясь на информации о кафедре и номера группы пользователя.
- Создать средства быстрого редактирования тем при добавлении файлов в библиотеку.
- Для более удобной публикации документов стоит добавить использование InfoPath форм.
- В целях автоматизации работы и повышения прозрачности развертывания системы требуется создать и подключить шаблоны списков «Category», «Sections» и библиотеки документов.

Список литературы

IntranetNo.Ru. (б.д.). *Корпоративные Порталы*. Получено из intranetno.ru:
<http://www.intranetno.ru/portal/>

Microsoft. (б.д.). *Настройка синхронизации профилей*. Получено из TechNet:
<http://technet.microsoft.com/ru-ru/library/ee721049.aspx>

Microsoft. (б.д.). *Схема: публикация документов Visio на сайте SharePoint*. Получено из Office Microsoft: <http://office.microsoft.com/ru-ru/visio-help/HA101835144.aspx>

TechNet, М. (б.д.). *Развертывание Office Web Apps (устанавливаемых с продуктами SharePoint 2010)*. Получено из TechNet: <http://technet.microsoft.com/ru-ru/library/ff431687.aspx>