

Сайт семинаров ректора МГУ

В. В. Осокин, Р. А. Ахунов

Рассматривается задача разработки сайта для ознакомления с материалами семинаров, проводимых ректором МГУ В. А. Садовничим. Создан сайт <http://sa.msu.ru>, позволяющий выкладывать для общественности результаты семинаров: статьи, доклады и фотографии. Архитектура сайта подразумевает использование уже готовой системы онлайн-сопровождения научно-образовательных процессов ДвинемНауку <http://dvinemnauku.ru> для хранения и ведения связанных с семинарами событий.

Ключевые слова: сайт, ДвинемНауку, база данных, программный интерфейс, материалы семинара.

Введение

В данной статье описывается создание сайта семинаров ректора МГУ Виктора Антоновича Садовничего под названиями «Время, хаос и математические проблемы» и «Спектральная теория дифференциальных операторов».

К сайту предъявлен ряд требований, часть из которых реализована при помощи интеграции с онлайн системой управления научно-образовательными процессами «ДвинемНауку». Другая часть требований реализована непосредственным использованием фреймворка Yii [1]. В статье описываются упомянутые требования к системе, использованные при реализации технологии, описывается сама система, ее база данных и наиболее значимые части исходного кода.

Постановка задачи и полученные результаты

Необходимо создать сайт, предоставляющий возможность добавлять и редактировать события, касающиеся семинаров ректора. События должны выводиться списком с кратким текстом и заголовком. Каждое событие можно просмотреть более подробно. Должна

быть возможность прикреплять к событию вложения, которые представляют собой рассмотренный на семинаре материал. Все вложения должны быть скачиваемыми, а если вложение — фотография, то эта фотография должна отображаться вместе с текстом события при подробном просмотре события. События, которые произошли до создания сайта, должны храниться в отдельной базе истории семинаров. Часть трудов (книги, статьи), касающихся тематики семинаров, является общей для всех событий семинаров, поэтому эти труды также должны храниться отдельно. Для этого необходима база книг семинаров. На сайте должна быть база научных школ МГУ, связанных с семинарами. Каждый элемент базы нужно отображать как в кратком, так и в подробном виде.

В результате работы создан сайт <http://sa.msu.ru>, который для реализации функционала ведения событий по семинарам использует уже готовое решение системы ДвинемНауку. Через API данной системы сайт получает список событий семинара и их вложения. Сайт делится на три части: верхняя часть сайта, содержащая меню и ленту фотографий, левая нижняя часть, содержащая список событий, и правая нижняя часть, содержащая контент по выбранной фотографии из ленты фотографий.

Левая часть сайта выводит список событий семинара. Каждый элемент списка содержит заголовок события, его дату и первый абзац текста события. В конце списка выводится пагинация.

Заголовок каждого события ссылается на страницу подробного просмотра события. На этой странице выводится заголовок события, его полный текст и его вложения. Если у события в качестве вложений имеются фотографии, то они отображаются не только в списке прикрепленных файлов, но и в полосе с фотографиями в верхней части сайта.

Верхняя часть содержит два меню: меню внешних ссылок и меню разделов сайта. Внешние ссылки указывают на головной сайт МГУ и на сайт кафедры математического анализа механико-математического факультета МГУ. Разделы сайта — это страницы, отображающие содержание баз истории семинаров, научных школ МГУ и книг семинаров. Описание этих баз представлено выше в требованиях к сайту.

Краткое содержание этих разделов выводится под меню разделов сайта. Эта часть сайта представляет собой полосу с фотографиями

и подписями под ними и возможностью прокрутки контента влево и вправо.

При нажатии на фотографию в правой части сайта появляется колаж из фотографий и краткое описание выбранного элемента полосы. Под кратким описанием имеется ссылка на подробное описание. Страница подробного описания подобна внутренней странице события, за исключением списка вложений, которого нет.

Интеграция с системой ДвинемНауку и ее использование на сайте

Как указано выше, часть задач по сайту возложена на сервис ДвинемНауку. Для этой цели написана программная часть сайта ректора, отвечающая за обращение к сервису. Создан класс, использующий в себе библиотеку cURL [2] для соединения с ДвинемНауку.

Ниже представлены переменные класса: `url` — адрес API ДвинемНауку для авторизации, `news_pid` и `ads_pid` — идентификаторы процессов в ДвинемНауку, из которых берутся события семинара.

```
class DvinemNauku extends ApplicationComponent
{
```

```
    public $url = 'http://dvinemnauku.ru/api/auth';
    public $news_pid=310;
    public $ads_pid=142;
```

Сам сервис ДвинемНауку предполагает только авторизованный доступ к контенту, поэтому для ведения событий семинара и получения этих событий через сайт семинара создан служебный аккаунт в ДвинемНауку. Для подключения к сервису ДвинемНауку и авторизации в ней выделен отдельный метод `connect` упомянутого класса. Разберем его подробнее. Для начала формируем данные запроса: имя пользователя и пароль для доступа к сервису ДвинемНауку.

```
public function connect()
{
    $fields_string='email='.urlencode('someemail@mail.com').
        '&password='.urlencode('somepass');
```

Инициализируем соединение через библиотеку cURL.

```
$ch = curl_init();
curl_setopt($ch,CURLOPT_URL, $this->url);
```

```
curl_setopt($ch,CURLOPT_POSTFIELDS, $fields_string);
curl_setopt($ch,CURLOPT_RETURNTRANSFER,1);
```

Выполняем запрос, получаем ответ с сервиса ДвинемНауку и закрываем соединение.

```
$result = curl_exec($ch);
curl_close($ch);
```

В качестве ответа сервис присылает идентификатор сессии, который используется в последующих запросах к сервису. Он сохраняется в сессии пользователя сайта.

```
$mas=json_decode($result,1);
Yii::app()->session['dn_sid']=$mas['sid'];
return $mas['sid'];
}
```

Следующий метод `get_info` нужен для получения списка событий через API ДвинемНауку. Параметры данного метода: идентификатор процесса, с которого берется список событий, идентификатор сессии, полученный при авторизации на сервисе ДвинемНауку, номер страницы и количество элементов на странице. Последние два параметра введены для удобства пагинации. Для начала мы формируем адрес, по которому будем получать события.

```
protected function get_info($pid,$sid,$page,$perpage)
{
    $get_url="http://dvinemnauku.ru/api/processes/" . $pid .
        "/events?PHPSESSID=" . $sid . "&page=" . $page . "&perpage=" .
        $perpage;
```

Авторизуемся на сервисе ДвинемНауку и инициализируем соединение через библиотеку cURL.

```
$sid=$this->connect();
$ch = curl_init();
curl_setopt($ch,CURLOPT_URL, $get_url);
curl_setopt($ch,CURLOPT_RETURNTRANSFER,1);
```

Выполняем запрос, получаем ответ сервиса и код ответа.

```
$result = curl_exec($ch);
$http_code = curl_getinfo($ch, CURLINFO_HTTP_CODE);
```

Если код ответа 200, то декодируем ответ от сервиса ДвинемНауку и возвращаем его в качестве результата выполнения метода.

```
if($http_code==200)
{
    $events=json_decode($result,1);
    return $events;
}
}
```

Информация по каждому событию семинара, получаемая от сервиса, включает в себя список вложений по событию (если таковые есть). Доступ к вложениям, как и к любому иному контенту на сервисе, осуществляется при наличии авторизованной сессии. В связи с этим нельзя использовать ссылки на вложения, которые непосредственно указывают на ДвинемНауку. Для того чтобы пользователь сайта мог скачать вложения написан метод `get_file` описываемого класса. Параметры метода: идентификатор сессии, ссылка на вложение.

```
protected function get_file($sid,$ref)
{
```

Дописываем идентификатор сессии в конец ссылки на вложение для авторизованного доступа.

```
    $get_url=$ref."?PHPSESSID=".$sid;
```

Инициализируем соединение через библиотеку cURL.

```
    $ch = curl_init();
    curl_setopt($ch,CURLOPT_URL, $get_url);
    curl_setopt($ch,CURLOPT_RETURNTRANSFER,1);
```

Выполняем запрос, получаем ответ и если код ответа 200, то полученный ответ от сервиса ДвинемНауку возвращаем как результат работы метода.

```
    $result = curl_exec($ch);
    $http_code = curl_getinfo($ch, CURLINFO_HTTP_CODE);
    if($http_code==200)
    {
        return $result;
    }
}
```

Все описанные выше методы являются базовыми для взаимодействия сайта семинара с сервисом ДвижемНауку. Остальные методы описанного класса используют эти базовые методы для получения конкретных данных с сервиса и не представляют особого интереса для описания. Описанный выше класс для работы с сервисом подключается в приложении как расширение фреймворка с названием `dvinemnauku`. Использование этого расширения мы увидим ниже.

Локальный функционал сайта ректора

Не все требования к сайту можно выполнить через систему ДвижемНауку. Часть поставленных задач сделана на стороне самого сайта при помощи фреймворка Yii.

В частности, для хранения статичной информации создана база данных MySQL [3] с двумя таблицами `article` и `info`. В первой таблице хранятся материалы (труды), касающиеся семинаров ректора, а вторая таблица нужна для статичного текста на сайте, который отображается в правой части сайта.

Таблица `article` содержит информацию по файлам, в которых хранятся труды семинаров. Каждый материал, который должен храниться в первой таблице, разбивается на части (главы, параграфы). Связано это с тем, что материалы имеют большие печатные объемы и неудобны для чтения, если будут представлены одним файлом. Поэтому в первой таблице хранятся части материалов. Каждая из частей связывается с материалом, из которого она взята, через идентификатор материала. Структура таблицы `article` следующая: идентификатор части, идентификатор материала, ссылка на файл части, настоящее название части (название главы, параграфа).

Вторая таблица имеет несколько основных полей: фото текста, колаж для текста, название и сам текст, даты создания и изменения данных, и последнее поле — раздел в котором находится контент.

Для работы с этими таблицами написаны классы контроллера и моделей к таблицам в БД. Классы моделей представляют собой стандартные классы моделей фреймворка Yii, и поэтому не стоит их отдельно рассматривать. Интерес представляет контроллер `SiteController`, а в частности те его методы, которые отличны от стандартных методов контроллеров фреймворка.

Первый метод, который с одной стороны является методом использующим сервис ДН, но с другой стороны является методом ло-

кального функционала — это метод `actionEvent`. Для этого метода устанавливается шаблон `internal_one_column`. Далее получаем данные о событии по его идентификатору. Как видно, получение данных идет через расширение `dvinemnauku`.

```
public function actionEvent($eventid, $id)
{
    $this->layout = 'internal_one_column';
    $event=Yii::app()->dvinemnauku->getEvent($eventid);
```

Чтобы отображать картинки из вложений события в ленте фотографий в верхней части сайта, их необходимо поместить в отдельный массив данных, который уже будет использован в шаблоне. Перед помещением вложений-фотографий в отдельный массив, форматируется ссылка на каждую фотографию.

```
    $this->imgs = array();
    foreach($event['uploads'] as $upload)
    {
        $part = explode('.', $upload['real_name']);
        $ext = strtolower(end($part));
        if ($ext == 'jpg' || $ext == 'jpeg')
            $this->imgs[] = array('photo' =>
                '/site/file?ref=http://dvinemnauku.ru
                /api/processes/'.$event['process_id'].
                '/events/'.$event['id'].'/files/'.
                $upload['id'].'&name=' .
                $upload['real_name'], 'title' =>
                $upload['real_name'], 'id' => false);
    }
    $this->render('event',array('event'=>$event));
}
```

Следующий метод используется для возможности скачивания вложений событий. Получаем содержимое вложения по его ссылке. По названию файла определяем его расширение.

```
public function actionFile()
{
    $file = Yii::app()->dvinemnauku->
        getFile(Yii::app()->getRequest()->getQuery('ref'));
    $part = explode('.', Yii::app()->getRequest()->
```

```
        getQuery('name'));
    $ext = strtolower(end($part));
```

Если расширение файла jpg, то в выходной буфер отдается заголовков для этого типа, и картинка вместо скачивания отображается в браузере. Если же вложение иного расширения, то отдаются заголовки для скачивания файла вложения.

```
    if ($ext == 'jpg' || $ext == 'jpeg')
        header('Content-Type: image/jpeg');
    else
    {
        header('Content-Description: File Transfer');
        header('Content-Type: application/octet-stream');
        header('Content-Disposition: attachment; filename="'.
            Yii::app()->getRequest()->getQuery('name').'");
        header('Content-Transfer-Encoding: binary');
        header('Connection: Keep-Alive');
        header('Expires: 0');
        header('Cache-Control: must-revalidate, post-check=0,
            pre-check=0');
        header('Pragma: public');
    }
    exit($file);
}
```

Заключение

Данный сайт разработан с учетом всех поставленных к нему требований. Реализована связь между системой ДвижемНауку и сайтом ректора. Создание и редактирование событий, связанных с семинарами, происходит на стороне системы ДвижемНауку.

Список литературы

- [1] <http://www.yiiframework.com>
- [2] <https://php.net/curl>
- [3] <http://www.mysql.com>