

Г. Л. МУРАВЬЕВ, В. П. ФОКИН
БрГТУ (г. Брест, Беларусь)

СПЕЦИАЛИЗИРОВАННЫЕ СРЕДСТВА ДОСТУПА К СОДЕРЖИМОМУ ВИКИ-САЙТОВ

Внедрение интернет-технологий привело к появлению новых форм коллективной работы пользователей, удаленных друг от друга физически. Перспективное направление – возможность накапливания содержимого баз данных путем коллективного управления содержимым соответствующими средствами оперативного доступа и редактирования (добавления, исправления, модификации). Известно значительное число сайтов указанной направленности, поддерживающих онлайн базы данных, например, мультязычная интернет-энциклопедия Википедия.

В основе таких ресурсов лежит технология «вики» (wiki) [1] создания веб-сайтов, предоставляющих пользователям средства многократного редактирования страниц, текстов; учёта версий страниц; визуализации изменений в данных в режиме реального времени и т.п. Их функциональность обеспечивается «вики-движком» – специализированной системой управления контентом сайта (CMS).

Анализ известных систем управления, направлений развития интернет-технологий, спектра средств, устройств удаленного, мобильного доступа позволяет сделать вывод о перспективности развития возможностей вики-сайтов, их CMS в части обеспечения доступа пользователей через разнотипные клиентские интерфейсы.

Здесь рассматривается проект соответствующей системы для совместного накопления и использования информации, обеспечивающей авторизацию, регистрацию пользователей, поддержку просмотра, создания и редактирования страниц, механизмов ведения истории изменений и т.п.

При этом решаются задачи:

- выбора архитектуры системы, моделей логического представления системы и ее составляющих, форматов передачи данных и т. д.;
- обеспечения механизмов инверсии управления, конфигурации маршрутизации и механизмов кроссдоменных запросов для целей обеспечения безопасности;
- выбора средств разработки, языков разметки, СУБД и т. д., обеспечивающих нужные характеристики функционирования.

Расширяемость, масштабируемость системы обеспечивается разделением интерфейсов, выбором в качестве модели построения архитектуры системы модели MVC (Модель-Представление-Контроллер) [2], предусматривающей логическую декомпозицию системы на «слабо связанные» компоненты - модели данных, их представление, визуализацию (пользовательский интерфейс) и процессы взаимодействия с пользователем (аппарат событий).

Для обеспечения потенциальной масштабируемости системы в соответствии с меняющимися требованиями в качестве интерфейса программирования приложения выбран интерфейс REST (на базе фреймворка Microsoft ASP.NET Web API), отличающийся безызычностью, простотой реализации, многоплатформенностью [3].

Средства взаимодействия пользователей с сайтом следует реализовать с помощью инструментов и технологий HTML, CSS и Javascript с использованием библиотеки Backbone.js и jQuery, подхода AJAX для поддержки «фонового» обмена данными браузера с веб-сервером, что повышает быстродействие веб-приложений.

Для обеспечения приемлемой скорости работы браузеров при реализации механизма отображения данных следует использовать HTML-шаблоны со специальным языком разметки, поддерживаемым библиотекой Dust.js [4].

Передачу данных между сервером и клиентом с учетом применения при разработке средств разных языков программирования и необходимости обеспечивать высокую языконезависимость предлагается осуществлять с использованием текстового формата данных JSON (JavaScript Object Notation). Формат отличается простотой в использовании, прозрачным синтаксисом, безызычностью и оперирует «универсальными» структурами данных, потенциально поддерживаемыми современными языками программирования [5].

В качестве языка разметки для создания страниц выбран «облегченный» язык Markdown [6], обладающий простым синтаксисом и позволяющий включать секции XHTML в Markdown-документ. В качестве модели базы данных и модели доступа к базе данных выбраны соответственно реляционная модель и механизм ADO.NET с использованием библиотеки Dapper [7], что обеспечивает поддержку SQL-запросов и независимость от особенностей конкретной базы данных.

Соответственно в качестве средств разработки и реализации системы использованы: – язык C# для написания серверной части приложения; – HTML для разметки веб-страниц; – язык JavaScript для реализации программного доступа к объектам приложений, обеспечения интерактивности веб-страниц; – фреймворк ASP.NET MVC Framework для создания веб-приложений на базе архитектуры MVC; – Microsoft SQL Server 2008 R2 в качестве СУБД.

Проведено макетирование системы, реализованной в виде набора модулей. В том числе, модуль, содержащий реализацию API, коды слоя отображения (архитектуры MVC), код доступа к базе данных; модуль, содержащий базовые компоненты, классы поддержки функциональности системы; модуль, содержащий набор скриптов на языке SQL для генерации таблиц баз данных, а также сценарии их ввального заполнения и др.

Выполнено модульное, интеграционное и нагрузочное тестирование макета системы на базе библиотек Rhino Mocks, NUnit, SpecFlow [8–10]. В процессе тестирования установлено, что запрос к базе данных на обработку 100 записей (чтение записей из базы данных и передачи в формате JSON по HTTP на клиентскую сторону - javascript сценарий, исполняющийся в браузере), выполняется менее чем за 0,1 секунды, что говорит о высоком быстродействии системы.

Таким образом, в работе представлены результаты анализа организации известных средств управления вики-сайтами, проектные решения перспективной системы управления данными на базе вики-технологии, данные о ее макетировании и испытаниях.

ЛИТЕРАТУРА

1. Википедия [Электронный ресурс]. – 2010. – Режим доступа: [http:// wikipedia.org](http://wikipedia.org).
2. Архитектура MVC [Электронный ресурс]. – 2011. – Режим доступа: <http://en.wikipedia.org/wiki/Model-view-controller>.
3. Интерфейс REST [Электронный ресурс]. – 2012. – Режим доступа: http://en.wikipedia.org/wiki/Representational_state_transfer.
4. Библиотека Dust.js [Электронный ресурс]. – 2012. – Режим доступа: <http://akdubya.github.com/dustjs>.
5. Официальная документация формата JSON [Электронный ресурс]. – 2010. – Режим доступа: <http://www.json.org>.
6. Спецификация языка Markdown [Электронный ресурс]. – 2012. – Режим доступа: <http://daringfireball.net/projects/markdown>.
7. [Электронный ресурс]. – 2010. – Режим доступа: <http://code.google.com/p/dapper-dot-net>.
8. [Электронный ресурс]. – 2011. – Режим доступа: <https://github.com/ayende/rhino-mocks>.
9. Официальный сайт библиотеки NUnit [Электронный ресурс]. – 2012. – Режим доступа: <http://www.nunit.org>.
10. Официальный сайт библиотеки SpecFlow [Электронный ресурс]. – 2012. – Режим доступа: <http://www.specflow.org/specflownew>.