

WEB-СРЕДА РАЗРАБОТКИ PASCALABC.NET

Ю.В. Белякова, С.С. Михалкович

Белякова Юлия Вячеславовна

Михалкович Станислав Станиславович

Южный федеральный университет

Факультет математики, механики и компьютерных наук

Ростов-на-Дону

Julbin@yandex.ru

Аннотация. С развитием Интернета все более удобным становится использование для различных целей специализированных Интернет-сервисов (например, Google Docs). В этом отношении разработка программ — не исключение. В данной работе описан специализированный Интернет-сервис, предназначенный для разработки программ на языке Паскаль прямо через браузер — Web-среда программирования PascalABC.NET. Рассмотрены ее основные возможности, реализация, а также преимущества перед обычными интегрированными средами. Приведены варианты использования.

Ключевые слова: Интернет-сервис, разработка, мобильность, пользователь, файл.

Annotation. Using of specialized Internet services (for example Google Docs) is becoming more convenient with the development of the Internet. In this respect programming is not an exception. A specialized Internet service designed for programming in Pascal directly via the browser — programming Web-environment PascalABC.NET is described in this paper. Its main features, implementation, and the advantages over general integrated environments are reviewed. Use cases are also given.

Keywords: Internet service, development, mobility, user, file.

ВВЕДЕНИЕ

На сегодняшний день все более важными условиями успешной разработки программ становятся мобильность (возможность моментального запуска программ на любом компьютере без установки соответствующего программного обеспечения), кроссплатформенность, а также сетевой доступ к файлам. Возможным решением данных задач является создание специализированного Интернет-сервиса — мобильного дополнения к обычной интегрированной среде, а именно *Web*-среды программирования с личным архивом программ.

В данной работе рассматривается Web-среда программирования **PascalABC.NET** [1]. PascalABC.NET [2] — это язык, совместимый с Delphi

Object Pascal и ориентированный на платформу .NET. Web-среда функционирует непосредственно через браузер и доступна по адресу <http://pascalabc.net/WDE/>.

Сервис, предоставляемый Web-средой программирования, существенно отличается от большинства Web-приложений. Во-первых, здесь необходима постоянная синхронизация многочисленных операций, производимых на клиенте и на сервере, так как практически любая команда приложения требует соответствующих действий на сервере. Отдельную сложность представляет синхронизация файловых операций на сервере с состоянием Web-редактора, а также браузера файлов на клиенте. Во-вторых, необходимо корректно обрабатывать большое число возможных ошибок. К стандартным сетевым ошибкам и ошибкам обращения к базе данных добавляются ошибки сервера, связанные с операциями над файлами и каталогами. В-третьих, дополнительные трудности вызывает высокая степень взаимодействия приложения с компилятором языка (например, для обеспечения интерактивного ввода/вывода). И, наконец, необходимо обеспечивать многоплановую защиту доступа от несанкционированных операций.

1. СУЩЕСТВУЮЩИЕ WEB-СРЕДЫ

На сегодняшний день Web-среды разработки распространены сравнительно мало. Большинство из них связаны с разработкой на языках *JavaScript*, *HTML* и *CSS*, которые исполняются непосредственно браузером. Другим типом подобных сервисов являются *online-компиляторы*. Здесь текст программы из окна браузера отправляется на сервер, где передается соответствующему компилятору, а уже результат выполнения программы возвращается клиенту.

Далее рассмотрим три лучших, на наш взгляд, Web-среды:

- *IDEOne* [3]. Система позволяет компилировать программы на 40 языках, компиляторы которых расположены на сервере. Результаты компиляции программ возвращаются на отдельной странице. Основным недостатком данной среды является отсутствие интерактивности — нет возможности использовать систему ввода.
- *WebDevStudio* [4]. Достаточно функциональная Web-среда разработки, позволяющая компилировать проекты на языках C, C++ и Java. После компиляции можно скачать полученный exe- или class-файл на локальный компьютер и исполнить его. Однако, выполнение проекта через браузер невозможно.
- *CodeRun Studio* [5]. Наиболее функциональная система из рассмотренных. Позволяет разрабатывать и исполнять Web-приложения на языках C#, PHP и JavaScript. Возможность разработки консольных приложений, к сожалению, отсутствует.

В целом, список возможностей существующих систем достаточно велик. Однако ни одна из рассмотренных нами Web-сред не поддерживает

интерактивную систему ввода/вывода (ввод/вывод во время выполнения программы). Для нас же интерактивность при работе с консольными приложениями, позволяющая максимально приблизить процесс разработки в Web-среде к привычному нам в настольных IDE, является одной из первоочередных задач.

2. ОСНОВНЫЕ ВОЗМОЖНОСТИ WEB-СРЕДЫ PASCALABC.NET

Web-среда разработки PascalABC.NET представляет собой Web-аналог интегрированной среды разработки программ на языке *Паскаль*. Список ее возможностей можно разделить на две основные группы: базовые функции среды программирования и ее возможности как Web-приложения.

Рассмотрим основную функциональность Web-среды.

- Как и в обычной среде, программу можно *компилировать* и *выполнять*. Процесс выполнения программы также можно досрочно остановить.
- Для разработки используется достаточно мощный Web-редактор с подсветкой синтаксиса, который позволяет работать с несколькими файлами одновременно, как и в обычной среде.
- Поддерживается интерактивный ввод-вывод.
- При обнаружении ошибок компиляции или выполнения курсор редактора позиционируется в соответствующее место программы.
- Поддерживается работа с модулями и библиотеками.
- Пользователь имеет возможность не только компилировать и выполнять программы через браузер, но и скачивать соответствующие exe- или dll-файлы на локальный компьютер.
- Доступна возможность *публикации* файлов в общий каталог (публикуемая программа должна компилироваться). Соответствующий файл получает специальное имя и становится доступен по адресу <http://pascalabc.net/WDE/?file=name.pas>. Кроме того, любой пользователь может открыть такой файл непосредственно в редакторе, используя соответствующую кнопку. Также доступен список всех опубликованных файлов.
- Поддерживается регистрация пользователей. Использовать систему могут как зарегистрированные, так и незарегистрированные пользователи. Последние, однако, работают во «временном» каталоге, доступном лишь в течение одного сеанса, в котором могут создавать и сохранять файлы исходных текстов.
- Зарегистрированные пользователи получают в свое распоряжение *постоянный каталог* на сервере. Кроме того, зарегистрированные пользователи имеют возможность переименовывать и удалять файлы и папки, а также создавать дополнительные папки и подпапки внутри своего каталога (имеется простая навигация по папкам).
- Доступен легко изменяемый администратором набор примеров программ на языке PascalABC.NET.

3. ТЕХНИЧЕСКОЕ ОПИСАНИЕ

Система является *ASP.NET-приложением*. Широко используется технология *AJAX*, позволяющая минимизировать число видимых перегрузок страницы, тем самым ускорив работу приложения, а также максимально приблизить процесс программирования к привычному в настольных интегрированных средах. Для реализации клиентского слоя обширно используется язык *JavaScript*.

Рассмотрим реализацию основных подсистем Web-среды.

- *Компиляция и выполнение программ*. Компиляция и выполнения программ производятся на сервере, где расположен соответствующий компилятор. Браузер пользователя синхронизируется с сервером посредством специальных запросов *AJAX*.

Рассмотрим данную схему подробнее. При нажатии кнопки «компилировать» или «выполнить» происходит следующее: текст пользовательской программы извлекается средствами редактора и отправляется в асинхронном запросе на сервер. Для этого используется *AJAX*-технология методов страницы: *JavaScript*-код вызывает статический метод сервера в виде `PageMethods.<имя метода>(<параметры метода>, <callback-функция>)`. Далее, на сервере запускается компилятор, происходит выполнение программы, и результаты возвращаются клиенту как параметры `<callback-функции>`, которая и вызывается на клиенте по завершению метода сервера. В зависимости от результата (ошибка компиляции, ошибка выполнения, данные для вывода) совершаются нужные действия. Например, при возникновении ошибки выполнения в окне вывода появляется ее текст, а курсор редактора позиционируется в соответствующую строку. Если ошибка произошла не в открытом файле, то нужный файл предварительно открывается.

- *Ввод-вывод*. Рассмотрим следующую программу:

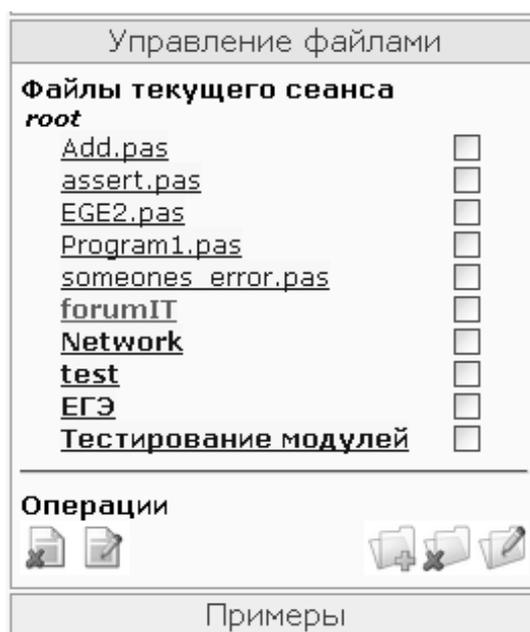
```
var x: integer;  
begin  
    read(x);  
    writeln('x = ', x);  
end.
```

Как и в общем случае, при запросе на выполнение текст программы из окна редактора отправляется на сервер, где передается компилятору. Когда выполнение откомпилированной программы доходит до оператора `read`, процесс приостанавливается, и клиенту возвращается сигнал о том, что ожидается ввод. На странице среды появляется соответствующее окно ввода. После того как пользователь завершит ввод данных, вновь вызывается *AJAX*-метод страницы, который получает на вход введенную строку. Далее, уже на сервере, она передается ожидающему процессу, выполнение возобновляется и продолжается до следующего оператора ввода, вывода, ошибки выполнения или успешного завершения программы.

Наличие в программе оператора `write` означает, что в процессе выполнения программы клиент получит от сервера сигнал об имеющихся данных для вывода, которые и будут отображены в соответствующем окне приложения.

Для синхронизации операций ввода/вывода используется метод опроса: каждые 300 мс на сервер отправляется запрос на наличие новых данных. Если такие данные есть, то они возвращаются на сторону клиента и тут же отображаются.

- *Работа с файлами.* Основной задачей клиентского слоя приложения является обеспечение корректной работы системы управления файлами, например: отслеживание несохраненных и открытых файлов, предотвращение выполнения некорректных команд, обеспечение соответствия реальной файловой структуре сервера. Кроме того, именно клиентская сторона предоставляет визуальную компоненту работы с файловой системой. Например, браузер файлов реализован самостоятельно, при этом существенно использованы объектно-ориентированные возможности языка JavaScript.



Непосредственно физические изменения в пользовательском каталоге сервера выполняются с использованием все тех же методов страницы AJAX.

Браузер файлов

Клиент отправляет серверу необходимую информацию (например, при сохранении это – полное имя файла и его содержимое), на сервере запрошенная операция выполняется, а результат возвращается клиенту.

- *Защита.* Значительную роль в работе приложения играет защита пользовательских данных: в качестве механизма хранения паролей выбрано хеширование, а также используются средства ASP.NET для защиты от SQL-инъекций.

С другой стороны, необходимо также обеспечить защиту сервера от умышленных (или случайных) потенциально опасных действий пользователя: например, попытки программным образом испортить файлы сервера. Данная возможность исключена — процессы выполнения пользовательских программ ограничены в доступе своим каталогом. Таким образом, файлы сервера не только нельзя испортить, но и даже невозможно программно получить информацию о существующих файлах и каталогах.

И, наконец, все процессы, запускаемые в среде, контролируются: в случае зависания процесса на сервере, через некоторое время он прерывается.

4. НЕКОТОРЫЕ ТЕХНИЧЕСКИЕ ПРОБЛЕМЫ

В процессе разработки мы столкнулись с некоторыми техническими ограничениями ASP.NET. Например, удаление или переименование папки в каталоге приложения приводит к немедленному завершению сеанса. Указанная проблема оказалась весьма распространенной, однако, в качестве единственного решения предлагалась перезагрузка страницы, что неприемлемо ввиду низкой скорости. Было реализовано решение искусственно возобновлять сеанс, используя еще один AJAX-механизм ASP.NET — компонент UpdatePanel. Как только сессия работы приложения завершается, на этой панели программно генерируется событие, вызывающее отправку на сервер запроса. В результате выполнения данного запроса начинается новая сессия, при этом никаких визуальных изменений на странице не происходит.

Еще одна проблема — невозможность использования стандартных компонентов ASP.NET для авторизации пользователей. Причина заключается в том, что данные компоненты используют перегрузку страницы, которая, как было сказано выше, неприемлема. Дополнительной сложностью авторизации является необходимость синхронизированных изменений на клиенте и сервере. В итоге решением стало использование набора HTML-элементов, размещенных на AJAX-компоненте UpdatePanel. Работа с базой данных пользователей производится через специальный класс ASP.NET — Membership. Несмотря на то, что все операции авторизации (установка статуса пользователя или работа с Cookies) приходится совершать вручную, использование данного класса, инкапсулирующего работу с базой, в значительной степени повышает безопасность приложения. Для синхронизации клиента и сервера используются серверные события для обратной связи (объект Sys.WebForms.PageRequestManager).

ЗАКЛЮЧЕНИЕ

Таким образом, заявленная задача создания Web-среды программирования PascalABC.NET выполнена в полном объеме. Web-среда удовлетворяет базовым возможностям IDE, таким как: компиляция, выполнение, интерактивный ввод/вывод, позиционирование на ошибках, хотя и уступает по функциональности современным интегрированным средам. Однако она обладает рядом других важных преимуществ:

- запуск программы на любом компьютере без предварительной установки соответствующего программного обеспечения
- хранение программ во внешнем хранилище с возможностью моментального запуска
- кроссплатформенность
- скорость выполнения
- всегда актуальная версия системы
- возможность создания специализированного архива программ

Указанные свойства, прежде всего, позволяют использовать Web-среду PascalABC.NET в качестве мобильного дополнения к основной среде [5], а в некоторых случаях она может выступать и в качестве единственного инструмента разработки.

В настоящее время Web-среда программирования PascalABC.NET используется в учебном процессе при обучении основам программирования студентов специальности «Информационные технологии» мехмата ЮФУ.

Кроме этого, Web-среда используется как специализированный файловый архив программ с возможностью публикации ссылок на запускаемые файлы на различных сайтах и форумах (аналогично Google Docs).

Web-среда может использоваться:

- как компонент при реализации дистанционного обучения программированию
- для организации автоматической проверки задач на олимпиадах по программированию
- в качестве тонкого клиента для реализации облачных вычислений

СПИСОК ЛИТЕРАТУРЫ

1. Ю. В. Белякова, И. В. Бондарев, С. С. Михалкович. Первое сообщение о Web-среде разработки PascalABC.NET. / Научно-методическая конференция «Современные информационные технологии в образовании: Южный Федеральный округ». Ростов-на-Дону, с.58-59.
2. Н. Н. Водолазов, С. С. Михалкович, А. В. Ткачук. Опыт разработки учебного языка программирования для платформы .NET / Научно-методическая конференция «Современные информационные технологии в образовании: Южный Федеральный округ». Ростов-на-Дону, с.71-72.
3. Электронный ресурс <http://ideone.com/>.
4. Электронный ресурс <http://gayuba5.datsi.fi.upm.es/~iortiz/webdevstudio/>.
5. Электронный ресурс <http://www.coderun.com/ide/>.
6. Электронный ресурс <http://pascalabc.net/>.