Белгородский Государственный Технологический Университет им. В. Г. Шухова

Кафедра программного обеспечения вычислительной техники  
и автоматизированных систем

## Лабораторная работа №6 по теме: «Использование AJAX-запросов в веб-приложениях»

**Выполнил:**  
студент группы ПВ-41  
Адаменко И. И.

**Проверил:**  
старший преподаватель  
Гарибов А. И.

Белгород  
2015

**Цель работы:** научиться работать с AJAX-запросами в контексте веб-приложений.

# Задание

Разработать веб-приложение на технологии PHP или ASP.NET с использованием AJAX-запросов.

# Теоретическая часть

## AJAX

AJAX (Asynchronous Javascript and XML — «асинхронный JavaScript и XML») — подход к построению интерактивных пользовательских интерфейсов веб-приложений, заключающийся в «фоновом» обмене данными браузера с веб-сервером. В результате, при обновлении данных веб-страница не перезагружается полностью, и веб-приложения становятся быстрее и удобнее.

Отличия стандартного подхода к построению веб-приложений от подхода с использованием AJAX можно представить в виде следующего сравнения:

В классической модели веб-приложения:

* Пользователь заходит на веб-страницу и нажимает на какой-нибудь её элемент.
* Браузер формирует и отправляет запрос серверу.
* В ответ сервер генерирует совершенно новую веб-страницу и отправляет её браузеру и т. д. После чего браузер полностью перезагружает всю страницу.

При использовании AJAX:

* Пользователь заходит на веб-страницу и нажимает на какой-нибудь её элемент.
* Скрипт (на языке JavaScript) определяет, какая информация необходима для обновления страницы.
* Браузер отправляет соответствующий запрос на сервер.
* Сервер возвращает только ту часть документа, на которую пришёл запрос.
* Скрипт вносит изменения с учётом полученной информации (без полной перезагрузки страницы).

У подхода с использованием AJAX есть как преимущества, так и недостатки.

Преимущества:

* Экономия трафика
* Уменьшение нагрузки на сервер
* Ускорение реакции интерфейса
* Почти безграничные возможности для интерактивной обработки

Недостатки:

* Отсутствие интеграции со стандартными инструментами браузера
* Динамически загружаемое содержимое не доступно поисковым системам
* Усложнение проекта
* Требуется включённый JavaScript в браузере
* Риск фабрикации запросов другими сайтами

# Практическая часть

В качестве предметной области был выбран небольшой сервис по расчёту корней квадратных уравнений (по коэффициентам) и площади треугольника (по длине сторон).

Серверная часть, где производится весь расчёт, была написана на PHP. Клиентская же часть использует JavaScript для создания AJAX-запроса к серверу, в котором содержатся введённые в форму данные. В ответ она получает строку, которую выдаёт пользователю в качестве ответа.

Ниже представлен код AJAX-запроса с поясняющими комментариями:

1. /\*
2. В этом примере AJAX-запрос отправляется в случае,
3. когда пользователь нажимает кнопку «Отправить»
4. в форме для ввода данных
5. \*/
6. $('form').submit(**function**(e) {
7. /\*
8. Чтобы не произошла перезагрузка страницы,
9. стандартное поведение браузера предотвращается
10. \*/
11. e.preventDefault();
13. /\*
14. Создание AJAX-запроса с помощью jQuery
15. \*/
16. $.ajax({
17. // адрес, на который отправляется запрос
18. url: location.href + 'solvers/' + $(**this**).attr('action'),
19. // метод, с помощью которого отправляется запрос
20. type: 'GET',
21. // данные запроса
22. data: {
23. a: $('input[name=a]', $(**this**)).val(),
24. b: $('input[name=b]', $(**this**)).val(),
25. c: $('input[name=c]', $(**this**)).val()
26. }
27. })
28. // выполняется, когда запрос завершился удачно
29. .done(**function**(response) {
30. alert('Ответ: ' + response + '.');
31. })
32. // выполняется, когда запрос завершился неудачно
33. .fail(**function**(jqXHR) {
34. // проверка кода ошибки
35. **if** (jqXHR.status == 400) {
36. alert(jqXHR.responseText)
37. } **else** {
38. alert('Что-то пошло не так. Перезагрузите страницу и попробуйте ещё раз!');
39. }
40. });
41. });

Таким образом, страница не перезагружается, а пользователь получает решение введённой задачи.

# Скриншоты

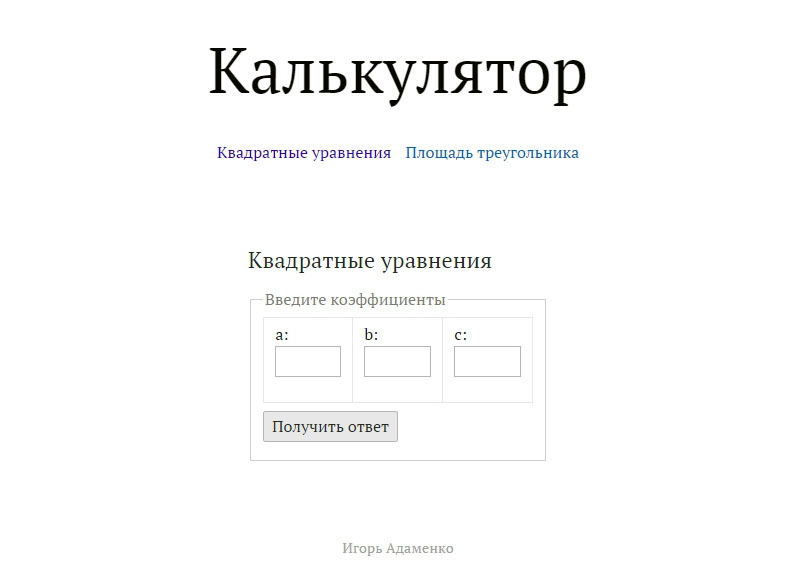


Рисунок 1. Страница, на которой используется указанный выше код.



Рисунок 2. Результат работы указанного выше кода.

# Вывод

В ходе этой лабораторной работы я изучил технологию AJAX и научиться работать с AJAX-запросами в контексте веб-приложений.