

Электронный научный журнал «Век качества» ISSN 2500-1841 <http://www.agequal.ru>

2016, № 3 [http://www.agequal.ru/pdf/2016/AGE\\_QUALITY\\_3\\_2016.pdf](http://www.agequal.ru/pdf/2016/AGE_QUALITY_3_2016.pdf)

**Ссылка для цитирования этой статьи:**

Воронцов Ю.А., Козинец А.В. Пример построения web-сервиса с использованием Apache и MySQL // Электронный научный журнал «Век качества». 2016. №3. С. 75-101. Режим доступа: <http://www.agequal.ru/pdf/2016/316006.pdf> (доступ свободный). Загл. с экрана. Яз. рус., англ.

УДК 004.7

**Пример построения web-сервиса с использованием Apache и MySQL.**

***Воронцов Юрий Алексеевич***

*заведующий кафедрой информационных систем,*

*профессор, доктор технических наук,*

*Московский технический университет связи и информатики*

*125993, Москва, ул. Народного Ополчения, 32, каб.409а*

[\*yvorontsov.1943@mail.ru\*](mailto:yvorontsov.1943@mail.ru)



***Козинец Артур Валерьевич***

*заведующий лабораториями кафедры информационных*

*систем*

*Московский технический университет связи и информатики*

*125993, Москва, ул. Народного Ополчения, 32, каб.407*

[\*kozinets@mtuci2.ru\*](mailto:kozinets@mtuci2.ru)



**Аннотация**

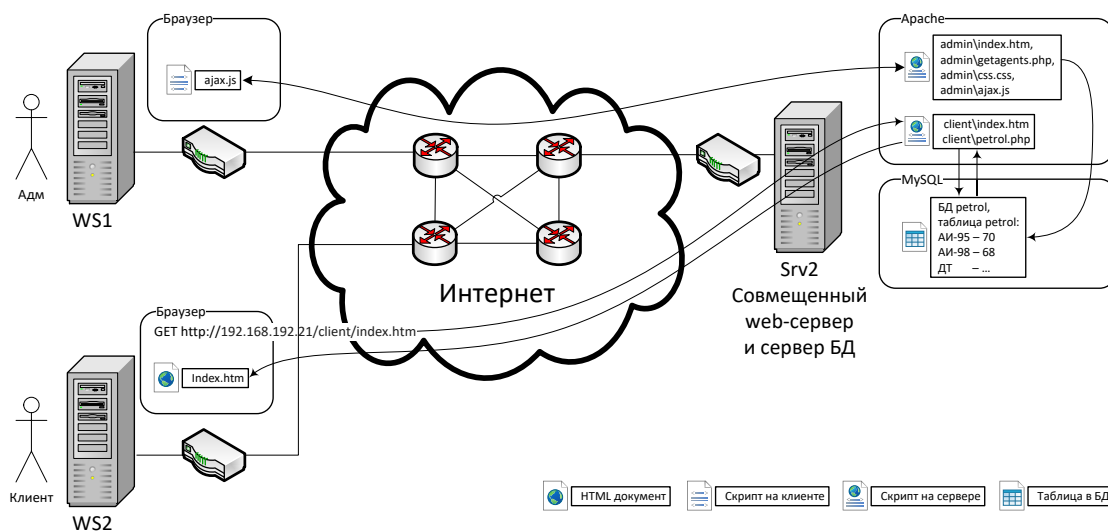
В статье рассматривается пример построения web-сервиса. В качестве средств реализации используются скриптовый язык PHP и веб-сервер Apache. Данные хранятся в базе данных, управляемой с помощью MySQL. При построении информационной системы применены средства виртуализации.

**Ключевые слова:** распределенные информационные системы; веб-сервисы; веб-службы; PHP; Apache; MySQL.

Иногда распределенной информационной системой (РИС) называют такую распределённую систему, в которой функционирует более одного сервера БД. Распределенная информационная система в узком смысле, с позиции программного обеспечения (ПО) – это совокупность программных компонент, взаимодействующих друг с другом. Каждая из таких компонент может рассматриваться как программный модуль (приложение), исполняемый в рамках отдельного процесса. Частным случаем такого приложения являются web-сервисы [1]. Современные распределённые информационные системы рассматриваются как распределенные системы программного обеспечения. Связь между цепочкой взаимодействующих процессов — есть суть распределенных систем.

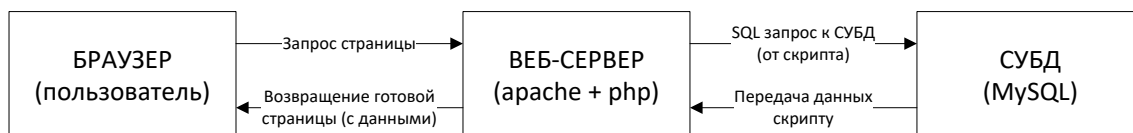
В статье будет рассмотрено решение следующей задачи. Таблица бензин-цена находится в базе данных на сервере. Администратор производит изменения цены в базе через web-страницу управления БД. Изменения цены в базе автоматически отображаются на HTML странице web-сервера. Клиент работает только в режиме чтения таблицы бензин-цена (HTML страница размещается на сервере). Клиент через браузер заходит на HTML страницу вручную по IP адресу. В качестве подхода при реализации задачи необходимо

Концептуальная модель распределенной информационной системы представлена на рисунке 1.



**Рис. 1. Концептуальная модель. Разработано авторами.**

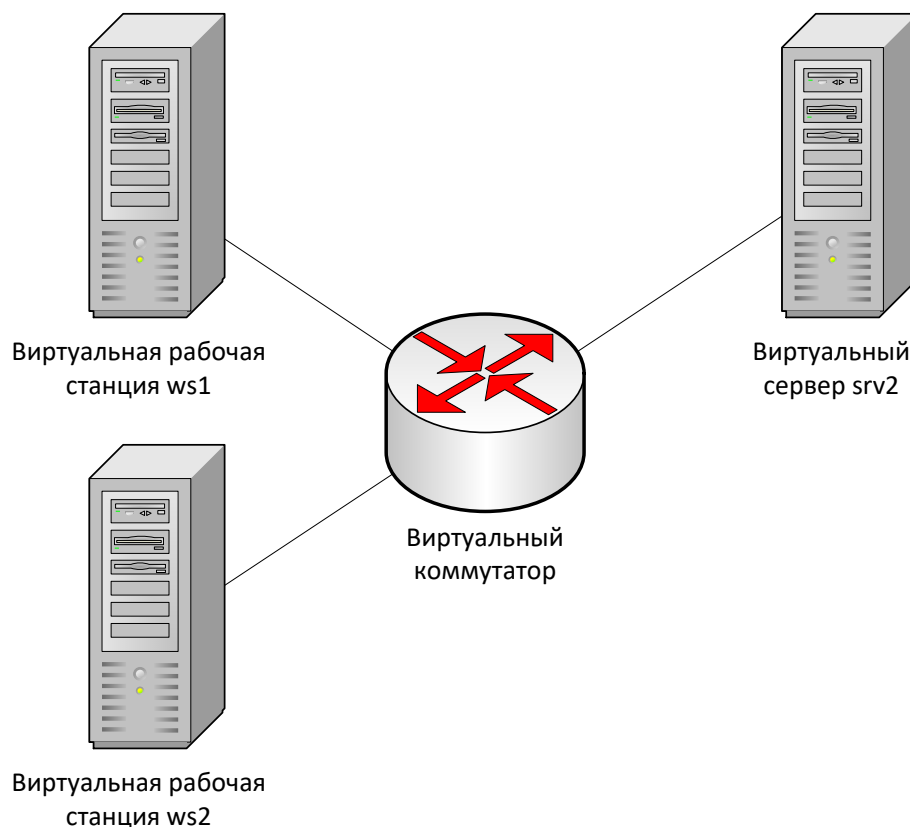
Прикладная архитектура информационной системы представлена на рисунке 2. Система состоит из трех компонентов: браузера, размещающегося на рабочих станциях, веб-сервера и сервера базы данных, размещающихся на сервере.



**Рис. 2. Прикладная архитектура решения. Разработано авторами.**

### Реализация инфраструктуры ИС.

При решении многих задачи заниматься построением полноценной физической инфраструктуры не целесообразно из-за больших финансовых расходов [2]. Мы рассмотрим виртуальную инфраструктуру, которая по своему составу будет соответствовать ИС, описанной выше. Системная архитектура решения представлена на рисунке 3.



**Рис. 3. Системная архитектура решения. Разработано авторами.**

### **Виртуализация сервера**

На персональном компьютере создаем инфраструктуру ИС. В качестве средства виртуализации [3] будем использовать ПО VirtualBox [4, 5, 6]. Для создания виртуальной ОС щелкните кнопку «Создать». Запустить мастер создания новой виртуальной машины. Следуя его инструкциям нужно будет выбрать семейство будущей виртуальной операционной системы и ее версию, а также, ввести ее имя.

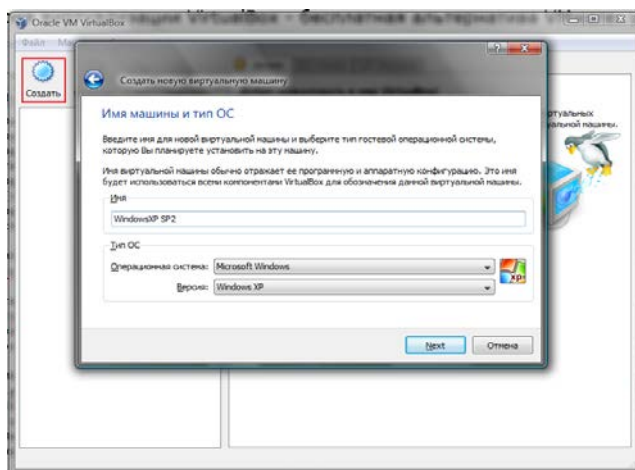


Рис. 4. Создание виртуальной машины [4].

Далее, загрузчик попросит нас указать объем оперативной памяти доступной виртуальной ОС.

Далее нужно будет создать новый жесткий диск, это довольно простая процедура, нужно просто следовать инструкциям мастера создания новых жестких дисков.

После установки необходимо настроить созданную виртуальную машину. Все настройки интуитивно понятны.

На вкладке Носители необходимо выбрать образ Windows Server для ее последующей установки:

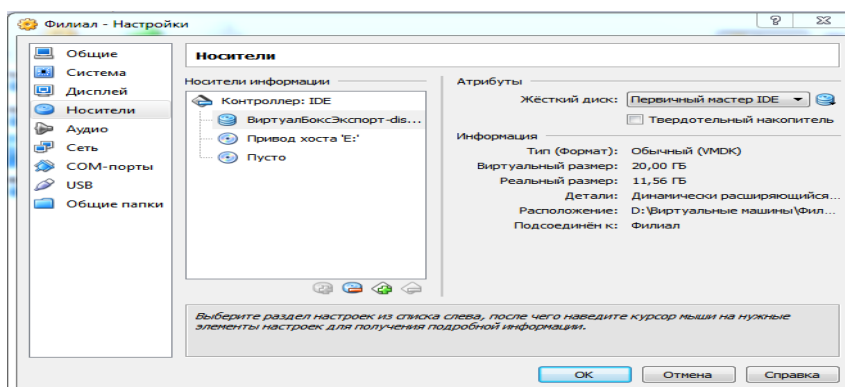


Рис. 5. Настройка накопителей [4].

На вкладке Система необходимо выделить память, поставить приоритет загрузки на Жесткий диск, выделить процессоры для гостевой ОС.

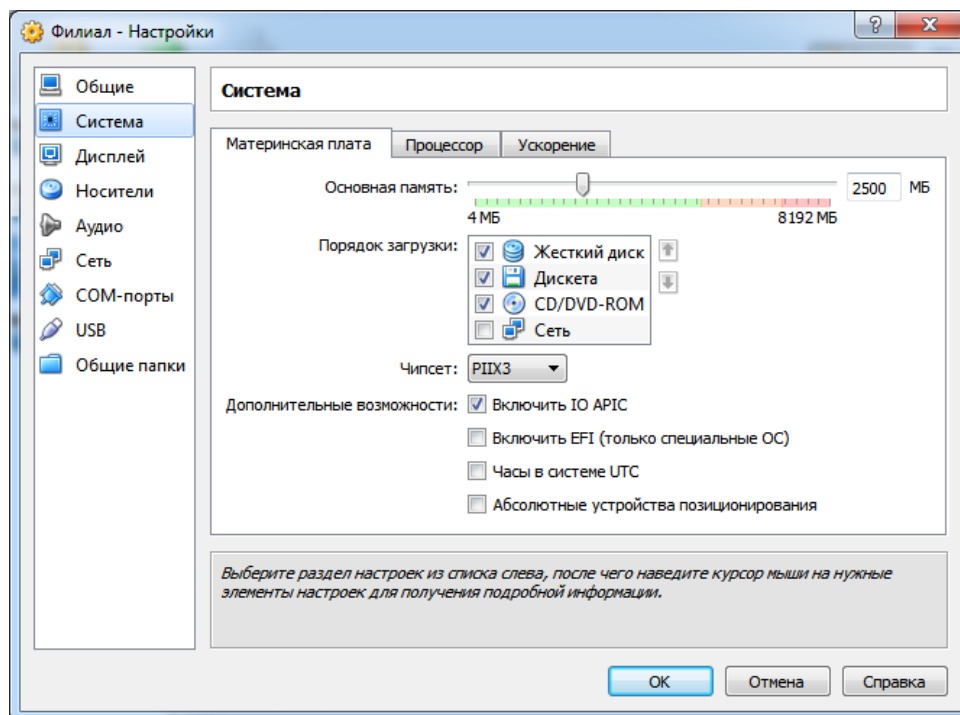


Рис. 6. Настройка системы [4].

Для того, чтобы иметь возможность обмена с основной ОС можно подключить Общую папку:

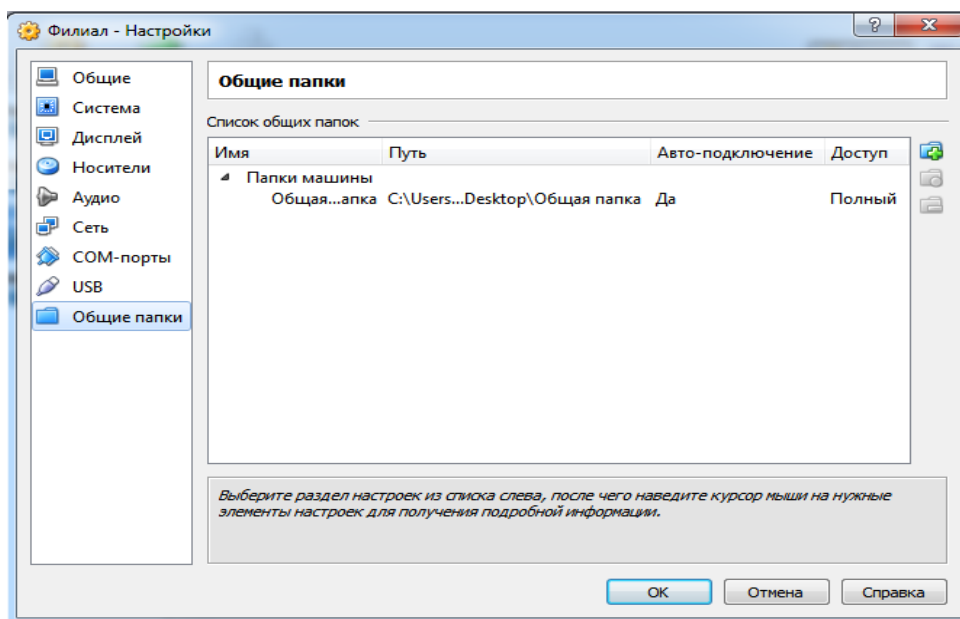


Рис. 7. Общая папка [4].

На вкладке сеть необходимо добавить 2 адаптера: первый для работы с сетью Интернет, второй для соединения между Виртуальными машинами.

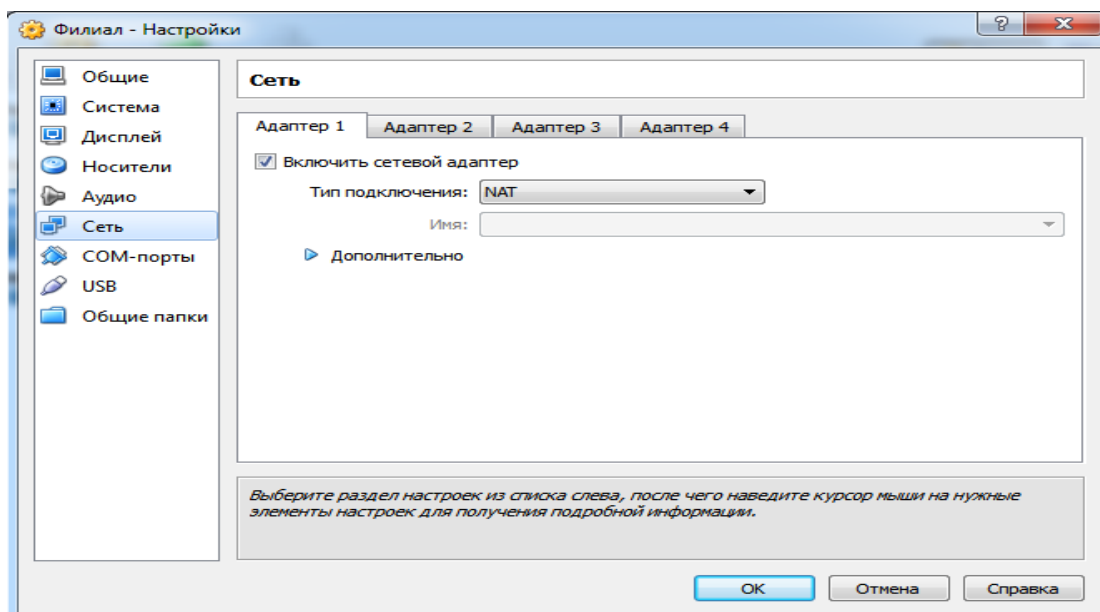
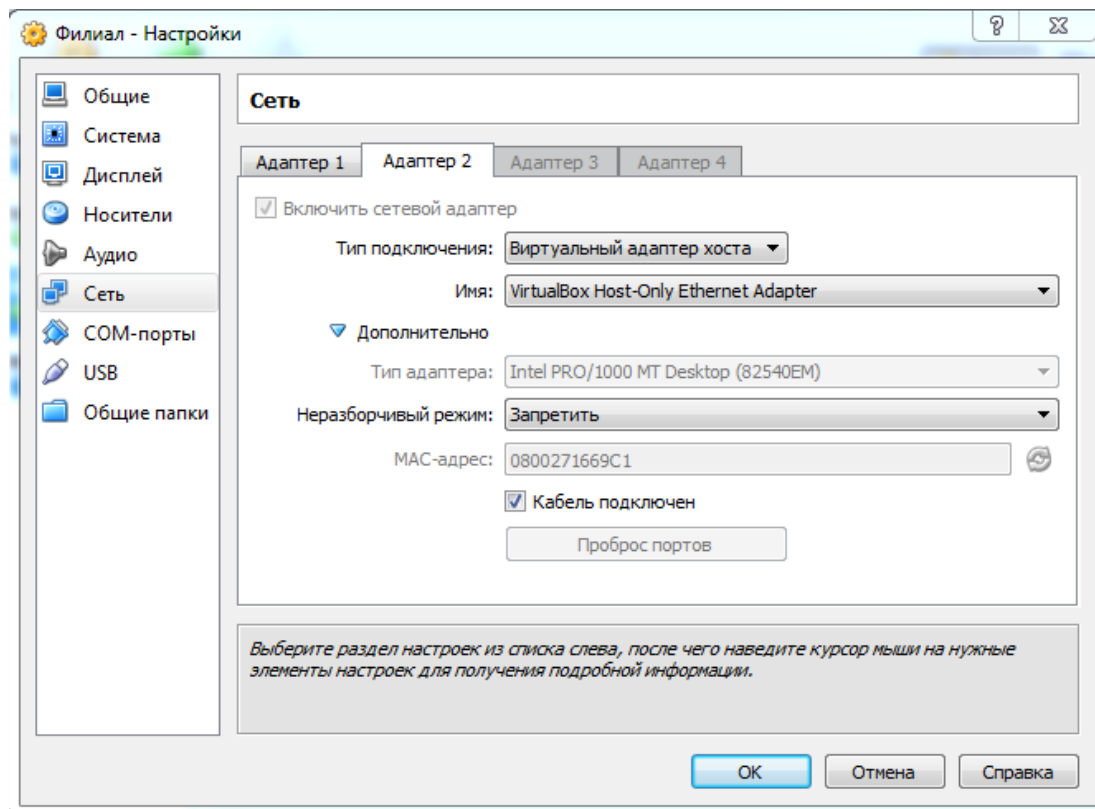


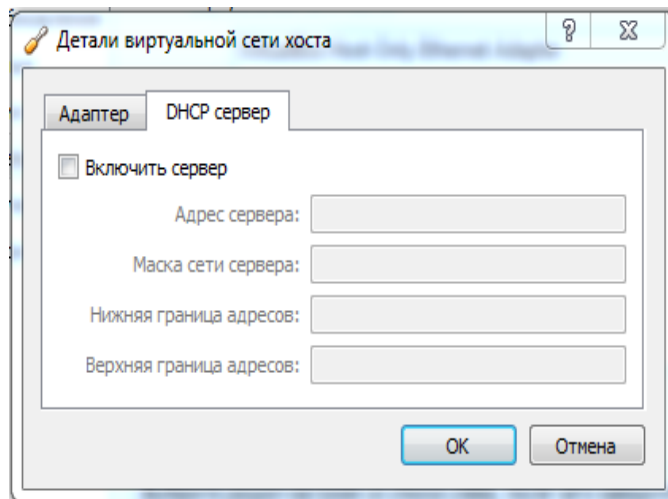
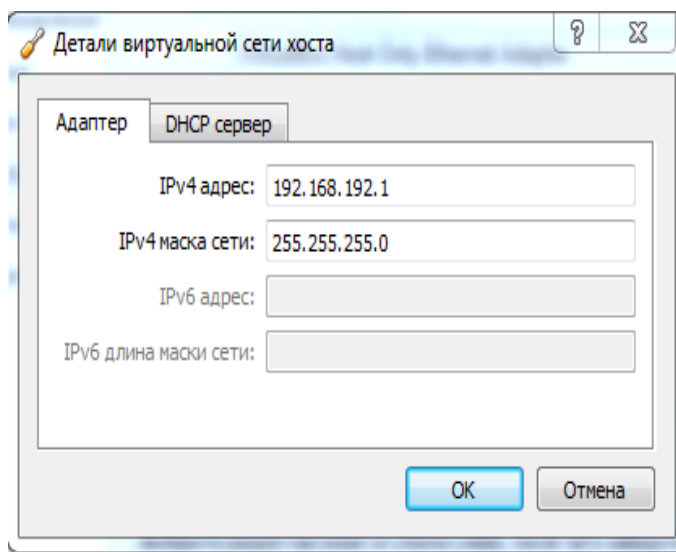
Рис. 8. Первый адаптер [4].



**Рис. 9. Второй адаптер [4].**

### **Настройка сетевого соединения сервера и рабочей станции.**

Для настройки соединения виртуальной машины ws1 и виртуальной машины srv2 необходимо в VirtualBox нажать Файл – Настройки – вкладка Сеть. Вводим IP адрес хоста для Виртуальных машин и отключаем DHCP.



**Рис. 10. Настройка виртуальной сети хоста. [5].**

На виртуальных машинах ws1 и ws2 необходимо настроить подключение по локальной сети [6] в процессе установки гостевых операционных систем. Детально установка операционной системы описана в статье [2]. Для этого

необходимо задать свойства Протокола интернета TCP/IP. Необходимо выбрать постоянный IP для виртуальной машины из сети хоста, заданной в 1 пункте (Сеть 192.168.192.0, Диапазон возможных значений 192.168.192.2 - 192.168.192.254).

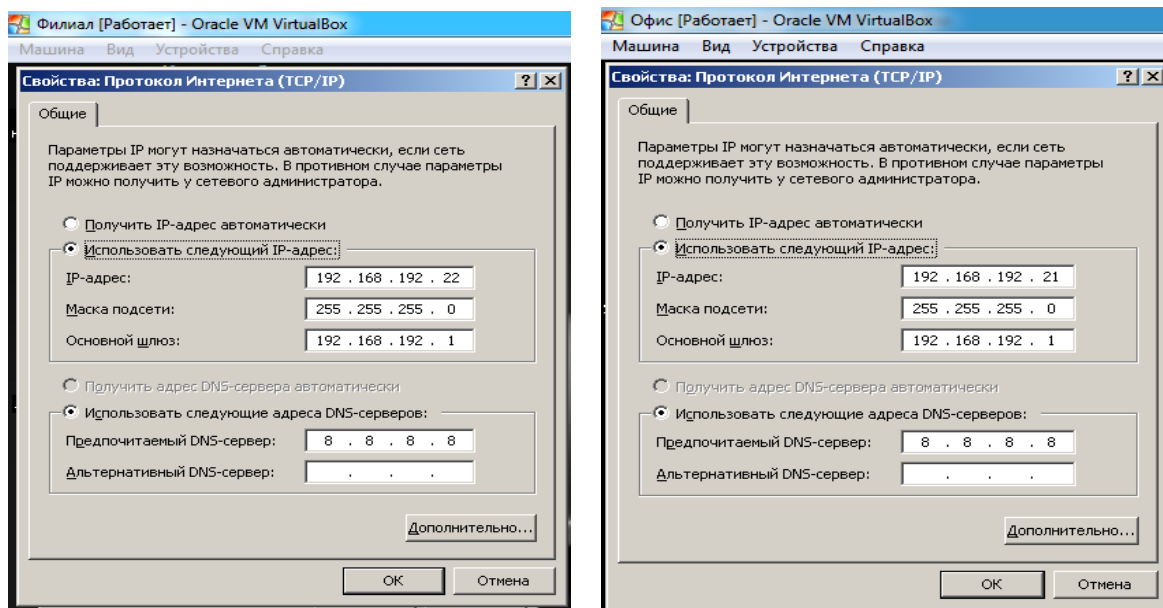


Рис. 11. Настройка локальной сети. Разработано авторами.

Настройка завершена. Проверим доступность Виртуальных машин выполнив из командной строки команду ping:

Доступность рабочей станции (192.168.192.22) с сервера, а доступность сервера (192.168.192.21) с рабочей станции:

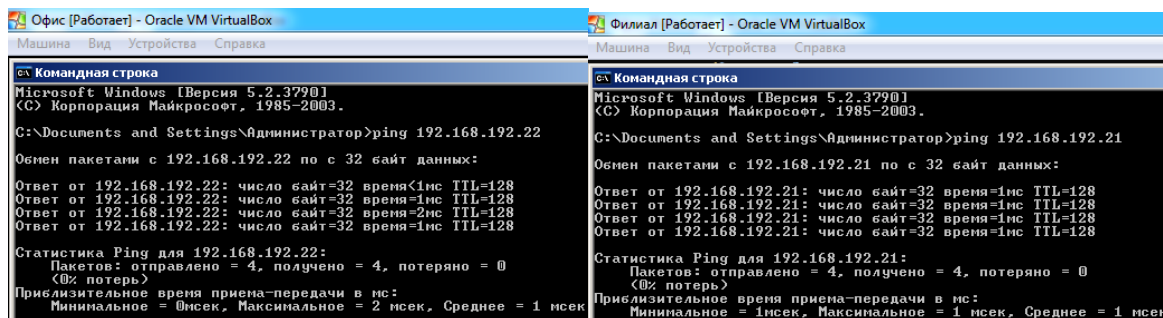


Рис. 12. Проверка локальной сети. Разработано авторами.

Доступность каждой рабочей станции по отношению к другой является необходимым условием для обеспечения работоспособности ИС.

## Программная реализация

Для создания Web-сервиса нужно установить на компьютер локальный сервер Denwer [7, 8]. Denwer — набор дистрибутивов, которые используются Web-разработчиками для разработки сайтов на локальной Windows-машине без необходимости выхода в Интернет.

1. При первом запуске, инсталлятор спрашивает: «Вы действительно хотите установить базовый пакет?», нажимаем «Да».
2. В приветственном окне нажимаем «Enter», как написано.

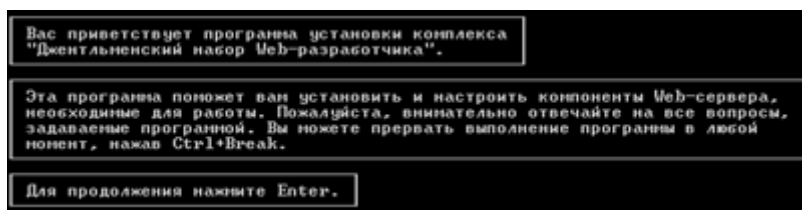


Рис. 13. Подтверждение установки [7].

3. После этого указываем папку, куда будет устанавливаться сервер. Можно оставить по умолчанию «C:\WebServers».

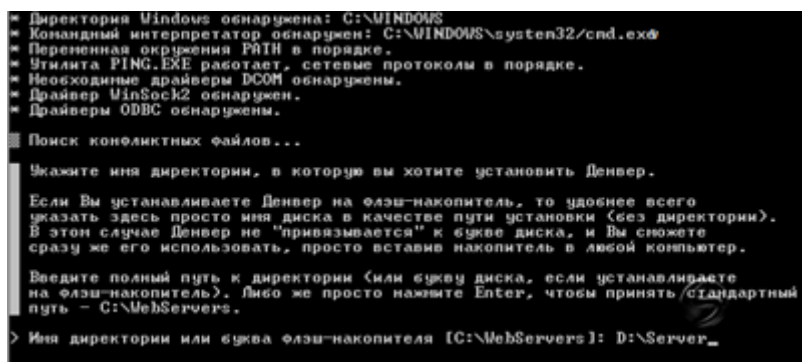


Рис. 14. Путь установки [7].

4. После выбора папки нажимаем «Enter». На вопрос установщика об уверенности в установке в эту директорию, нажимаем «Y» и снова «Enter».

Далее предлагается выбрать букву виртуального диска. Нужно выбрать название диска, которого пока нет на компьютере.

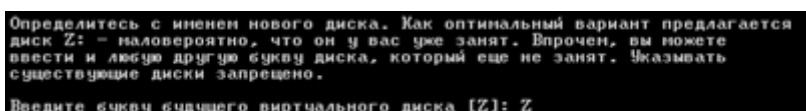


Рис. 15. Настройка виртуального диска [7].

5. Нажимаем «Enter». Последует копирование файлов в ту папку, которую мы указали изначально. После окончания копирования следует окно, в котором нужно ввести “1”.

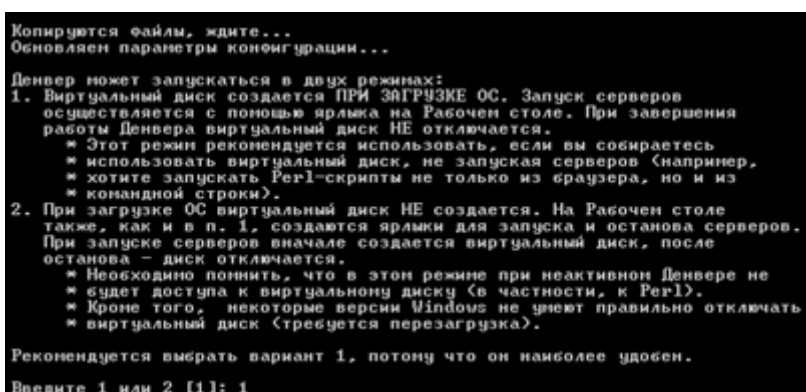
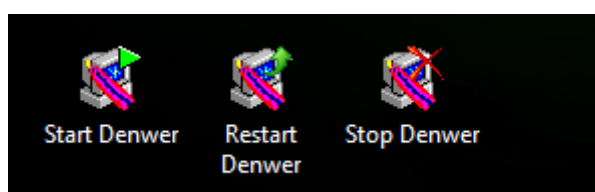


Рис. 16. Подтверждение настройка виртуального диска [7].

6. После этих действий установщик закроется, и откроется браузер с заголовком «Денвер успешно установлен». На рабочем столе появилось 3 ярлыка:



### **Рис. 17. Ярлыки на рабочем столе [7].**

7. Запускаем Денвер. В браузере (Internet Explorer, Opera, Mozilla Firefox и т.д.) вписываем адрес «localhost», в результате должно выйти сообщение с заголовком «Ура, заработало!».

Простой способ открытия сайта. Самый простой способ заставить Денвер открывать по адресу `http://192.168.192.21` определенный сайт — это расположить документы сайта в директории `/home/192.168.192.21/www`.

Не забудьте также перезапустить Денвер.

#### **Создаем БД на MySQL**

`C:\WebServers` Создаем БД на MySQL- для этого в адресной строке браузера вводим `localhost/Tools/phpmyadmin`

Мы попадаем на главную страницу для работы с MySQL. На верхней панели выбираем Базы Данных. В соответствующее поле вводим название БД – `retrol`, в поле сравнение выбираем кодировку `utf-8_general_ic`. После этого на левой панели выбираем `retrol`. Нам открывается окно, в котором перечислены основные таблицы этой БД. Ниже будет область Создать таблицу. В соответствующие поля вводим название таблицы и количество столбцов. После нажатия кнопки ОК откроется окно для создания структуры таблицы. После того, как таблица будет создана, ее название появится в левой части. Нажимаем `retrol` и в открывшемся окне выбираем наверху вкладку Вставить. Откроется окно для заполнения таблицы.

#### **Создаем файловую структуру веб-сервера.**

Рабочий стол сервера 1. Нажимаем «Пуск» - «Проводник». Открываем путь: `C:/WebServers/home/192.168.192.21/www`

Создаем в этой директории две папки: `admin`, `client`

## Папка client

В папке **client** создаем два файла **index.htm** (в блокноте) и **petrol.php** (в блокноте) и сохраняем их. Ниже приведены листинги кодов для этих файлов. Файла **index.htm** следующего содержания:

```
<!DOCTYPE html >
<head>

<title>aj ax</title>
<script type="text/javascript">

function getXmlHttp() {
    var xmlhttp;
    try {
        xmlhttp = new ActiveXObject("Msxml2.XMLHTTP");
    } catch (e) {
        try {
            xmlhttp = new ActiveXObject("Microsoft.XMLHTTP");
        } catch (E) {
            xmlhttp = false;
        }
    }
    if (!xmlhttp && typeof XMLHttpRequest!='undefined') {
        xmlhttp = new XMLHttpRequest();
    }
    return xmlhttp;
}
setInterval(function petrol() {

    var req = getXmlHttp()
    var statusElem = document.getElementById('petrol_tab')

    req.onreadystatechange = function() {

        if (req.readyState == 4) {
            if(req.status == 200) {
                statusElem.innerHTML = req.responseText;
            }
        }

        req.open('GET', 'petrol.php', true); //необходимо
        указать путь к файлу, содержащему серверный скрипт для выборки
        данных из БД.
        req.send(null);
```

```
    }, 100);  
</scri pt>  
<meta      http- equi v="Content- Type"      content="text/html ;  
charset=UTF- 8" >  
</head>  
<body>  
<di v i d="petrol _tab" ></di v>  
</body>
```

Файла **petrol.php** следующего содержания:

```
<?php  
// определяем начальные данные необходимые для соединения с  
БД  
    $db_host = 'local host';  
    $db_name = 'petrol';  
    $db_username = 'root';  
    $db_password = '';  
    $db_table_to_show = 'petrol';  
    $result = mysql_query($sql);  
  
    // соединяемся с сервером базы данных  
    $connect_to_db      =      mysql_connect($db_host,  
$db_username, $db_password)  
        or die("Could not connect: " . mysql_error());  
  
    // подключаемся к базе данных  
    mysql_select_db($db_name, $connect_to_db)  
        or die("Could not select DB: " . mysql_error());  
  
    // выбираем все значения из таблицы "petrol"  
    $qr_result = mysql_query("select * from " .  
$db_table_to_show)  
        or die(mysql_error());  
  
    // выводим на страницу сайта заголовки HTML- таблицы,  
    // количество заголовков должно соответствовать количеству  
столбцов  
    echo '<table border="1" >';  
    echo '<thead>';  
    echo '<tr>';  
    echo '<th>id</th>';  
    echo '<th>petrol</th>';  
    echo '<th>price</th>';  
    echo '</tr>';
```

```
        echo '</thead>';
        echo '<tbody>';

// выводим в HTML-таблицу все данные из таблицы БД
// количество столбцов должно соответствовать, количеству
// столбцов в бд или количеству столбцов, запрашиваемых в
// запросе
        $qr_result = mysql_query
        while($data = mysql_fetch_array($qr_result)) {
            echo '<tr>';
            echo '<td>' . $data['id'] . '</td>';
            echo '<td>' . $data['petrol'] . '</td>';
            echo '<td>' . $data['price'] . '</td>';
            echo '</tr>';
        }

        echo '</tbody>';
        echo '</table>';
    ?>
```

### Папка admin

В папке **admin** создаем 4 файла **index.htm**, **getagents.php**, **css.css**, **ajax.js**  
(все файлы в блокноте) и сохранить

Файла **index.htm** следующего содержания:

```
<script language="javascript" src="ajax.js"></script>
<link rel="stylesheet" href="css.css" type="text/css">
<body onLoad="getagents('id','')">
<div id="hiddenDIV" style="visibility: hidden; background-
color: white; border: 0px solid black;"></div>
</body>
```

Файла **getagents.php** следующего содержания:

```
<?
class clsAJAX
{
// определяем данные для подключения к БД
    public $dbhost;
    public $dbuname;
    public $dbpass;
    public $dbname;
```

```
private $arrTable = "petrol";
public $arrFields = array('id', 'petrol', 'price');
// соединяемся с сервером БД
public function __construct()
{
    $dbhost = "localhost";
    $dbuname = "root";
    $dbpass = "";
    $dbname = "petrol";
    $db = mysql_connect($dbhost, $dbuname, $dbpass);
    if (!$db) { die('There was a problem with the
database, please try back later db.php'); }
    mysql_select_db($dbname, $db);
}

public function showList($id='')
{
    $textout = "";
    $param = $_GET['param'];
    $dir = $_GET['dir'];
    if(strlen($param)>0){

        $sortupimg = '^';
        $sortdownimg = 'v';

        $param = $_REQUEST['param'];
        $dir = $_REQUEST['dir'];

//Генерируем поля таблицы
        if ( $_GET['dir'] == 'desc' )
        {
            $textout .= '<tr class="txtheading">';
            for ( $i = 0 ; $i < count($this-
>arrFields) ; $i ++ )
            {
                $textout .= ($GET['param'] == $this-
>arrFields[$i])?'<td><a href="#" onClick=getagents("'. $this-
>arrFields[$i]. '", "")>'. $this->arrFields[$i]. '</a>
'. $sortdownimg. '</td>': '<td><a href="#"
onClick=getagents("'. $this->arrFields[$i]. '", "")>'. $this-
>arrFields[$i]. '</a></td>';
            }
            $textout .= '<td></td><td></td>';
            $textout .= '</tr>';
        }
        else
        {
            $textout .= '<tr class="txtheading">';
```

```
for ( $i = 0 ; $i < count($this->arrFields) ; $i ++ )
{
    $textout .= ($_GET['param'] == $this->arrFields[$i])?'<td><a href="#" onClick=getagents("' . $this->arrFields[$i]. '", "desc")>' . $this->arrFields[$i]. ' </a>' . $sortupimg. ' </td>':'<td><a href="#" onClick=getagents("' . $this->arrFields[$i]. '", "desc")>' . $this->arrFields[$i]. ' </a></td>';
}
$textout .= '<td></td><td></td>';
$textout .= '</tr>';
}

//заканчиваем генерировать поля

$arrf='';
for ( $i = 0 ; $i < count($this->arrFields) ; $i ++ )
{
    $arrf .= ', ' . $this->arrFields[$i];
}

$q = ' SELECT
' . substr($arrf, 1, strlen($arrf)) . ' FROM ' . $this->arrTable . ' ORDER
BY ' . $param . ' ' . $dir;

$result = mysql_query($q);
while( $myrow = mysql_fetch_array($result)
){
    if ( $id == $myrow[$this->arrFields[0]] )
    {
        //определение поля для вывода значений из БД, необходимо,
        чтобы значения атрибутов //name, id соответствовали именам
        столбцов в БД
        $textout .= '
        <tr
class="txtcontents">
        <td><input
type="text" size="15" class="textbox" name="id" id="id" readonly
value="' . $myrow[$this->arrFields[0]]. '"></td>
        <td><input
type="text" size="15" class="textbox" name="petrol" id="petrol"
value="' . $myrow[$this->arrFields[1]]. '"></td>
        <td><input
type="text" size="25" class="textbox" name="price" id="price"
value="' . $myrow[$this->arrFields[2]]. '"></td>
```

```

                                <td><a href="#"
onCl i ck=saveRecord("save", ' . $myrow[ $this-
>arrFi el ds[0] ]. ', "" . $param. ' ", "" . $di r. ' ") >Save</a> </td>
                                <td>| <a href="#"
onCl i ck=getagents(" . $param. ' ", "" . $di r. ' ") >Cancel </a></td>
                                </tr>
                                ' ;
                                }
                                else
                                {
                                $textout . = ' <tr
class="txtcontents">' ;
                                for ( $i = 0 ; $i < count($this-
>arrFi el ds) ; $i ++ )
                                {
                                $textout . = ' <td
width="18%">' . $myrow[ $this->arrFi el ds[ $i ] ]. ' </td>' ;
                                }
                                $textout . = ' <td><a href="#"
onCl i ck=mani pul ateRecord("update", ' . $myrow[ $this-
>arrFi el ds[0] ]. ', "" . $param. ' ", "" . $di r. ' ") >Update</a> </td>
                                <td>| <a href="#"
onCl i ck=mani pul ateRecord("del ete", ' . $myrow[ $this-
>arrFi el ds[0] ]. ', "" . $param. ' ", "" . $di r. ' ") >Del ete</a></td>
                                </tr>' ;
                                }
                                }
                                } else {
                                $textout=' <tr><td colspan="6">No record
availabl e. . </td></tr>' ;
                                }
                                if ( $_REQUEST[' mode' ] != "new" )
                                {
                                $textout . = ' <tr><td height="20"
val i gn="bottom" class="txtcontents" colspan="6">
                                <a href="#"
onCl i ck=newRecord("new", "" . $param. ' ", "" . $di r. ' ") >New</a>
                                </td></tr>' ;
                                }
                                else if ( $_REQUEST[' mode' ] == "new" )
                                {
                                $textout . = '
                                <tr
class="txtcontents">
/*определение полей для создания новой строки БД, необходимо,
чтобы значения атрибутов
name, id соответствовали именам столбцов в БД*/
                                <td></td>

```



```
if ( $_REQUEST[' mode' ] == "update" )
{
    $obj ->showLi st ( $_REQUEST[' i d' ] );
}

if ( $_REQUEST[' mode' ] == "save" )
{
    $obj ->saveEdi tedRecord();
    $obj ->showLi st ();
}

if ( $_REQUEST[' mode' ] == "newsave" )
{
    $obj ->saveNewRecord();
    $obj ->showLi st ();
}

if ( $_REQUEST[' mode' ] == "new" )
{
    $obj ->showLi st ();
}

if ( $_REQUEST[' mode' ] == "list" )
{
    $obj ->showLi st ();
}
?>
```

Файла **css.css** следующего содержания:

```
. txtheadi ng{
    font- fami ly: Verdana, Ari al , Hel veti ca, sans- seri f;
    font- si ze: 11px;
    font- wei ght: bol d
}

. txtcontents{
    font- fami ly: Verdana, Ari al , Hel veti ca, sans- seri f;
    font- si ze: 11px;
}

. textbox {
    font- fami ly: Verdana, Ari al , Hel veti ca, sans- seri f;
    font- si ze: 11px;
    di spl ay: bl ock;
    border- wi dth: 1px;
}
```

Файла **ajax.js** следующего содержания:

```
function GetXmlHttpRequest(handler)
{
    var objXmlHttp=null

    if (navigator.userAgent.indexOf("Opera")>=0)
    {
        alert("This example doesn't work in Opera")
        return
    }
    if (navigator.userAgent.indexOf("MSIE")>=0)
    {
        var strName="Msxml2.XMLHTTP"
        if (navigator.appVersion.indexOf("MSIE 5.5")>=0)
        {
            strName="Microsoft.XMLHTTP"
        }
        try
        {
            objXmlHttp=new ActiveXObject(strName)
            objXmlHttp.onreadystatechange=handler
            return objXmlHttp
        }
        catch(e)
        {
            alert("Error. Scripting for ActiveX might be
di sabled")
            return
        }
    }
    if (navigator.userAgent.indexOf("Mozilla")>=0)
    {
        objXmlHttp=new XMLHttpRequest()
        objXmlHttp.onload=handler
        objXmlHttp.onerror=handler
        return objXmlHttp
    }
}

//определяем путь к серверному скрипту и передаем параметры
var url = "getagents.php?param=";

function getagents(column, direc) {
    var
myRandom=parseInt(Math.random()*99999999);

    xmlhttp=GetXmlHttpRequest(handlerHttpResponse);
```

```
        xmlHttp.open("GET", url + escape(column) +
"&mode=list&dir=" + direc + "&rand=" + myRandom, true);
        xmlHttp.send(null);
    }
    //определяем функцию на сохранение записи
    function saveRecord(mode, id, param, dir)
    {
        /*определяем поля БД, в качестве значения метода
getElementById указываем id, указанный в серверном скрипте*/
        id = document.getElementById("id").value;
        petrol =
document.getElementById("petrol").value;
        price = document.getElementById("price").value;

        var
myRandom=parseInt(Math.random()*99999999);

        xmlHttp=GetXmlHttpRequest(handlHttpResponse);

        xmlHttp.open("GET", "getagents.php?id="+id+"&petrol="+petrol+"
&price="+price+"&mode="+mode+"&param=" + escape(param) + "&dir=" +
dir + "&rand=" + myRandom, true);
        xmlHttp.send(null);
    }

    function saveNewRecord(mode, param, dir)
    {
        //uid = document.getElementById("id").value;

        petrol =
document.getElementById("petrol").value;
        price = document.getElementById("price").value;

        if ( petrol.length == 0 || price.length == 0 )
        {
            alert("Please enter value for all the fields");
        }
        else
        {
            var
myRandom=parseInt(Math.random()*99999999);
            xmlHttp=GetXmlHttpRequest(handlHttpResponse);

            xmlHttp.open("GET", "getagents.php?petrol="+petrol+"&price="+p
ri ce+"&mode="+mode+"&param=" + escape(param) + "&dir=" + dir +
"&rand=" + myRandom, true);
```

```
        xmlhttp.send(null);
    }
}

function newRecord(mode, param, dir)
{
    var
myRandom=parseInt(Math.random()*99999999);

    xmlhttp=GetXmlHttpRequest(handleHttpResponse);

    xmlhttp.open("GET", "getagents.php?mode="+mode+"&param=" +
escape(param) + "&dir=" + dir + "&rand=" + myRandom, true);
        xmlhttp.send(null);
    }

function manipulateRecord(mode, id, param, dir)
{
    {
    if (confirm("Are you sure you want to "+mode+"
record ?") != 1)
    {
        return false;
    }

    var
myRandom=parseInt(Math.random()*99999999);

    xmlhttp=GetXmlHttpRequest(handleHttpResponse);

    xmlhttp.open("GET", "getagents.php?id="+id+"&mode="+mode+"&par
am=" + escape(param) + "&dir=" + dir + "&rand=" + myRandom, true);
        xmlhttp.send(null);
    }

    function handleHttpResponse() {
        if (xmlhttp.readyState == 4) {

document.getElementById("hiddenDIV").style.visibility="visible";

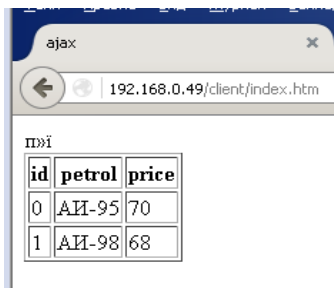
document.getElementById("hiddenDIV").innerHTML='';

document.getElementById("hiddenDIV").innerHTML=xmlhttp.responseText;
        }
    }
}
```

Теперь осталось открыть страницы в Web-обозревателе и посмотреть на результат. В браузере клиента набираем ip адрес сервера с указанием требуемой

страницы index.htm. Должны увидеть актуальную таблицу с марками топлива и их ценами.

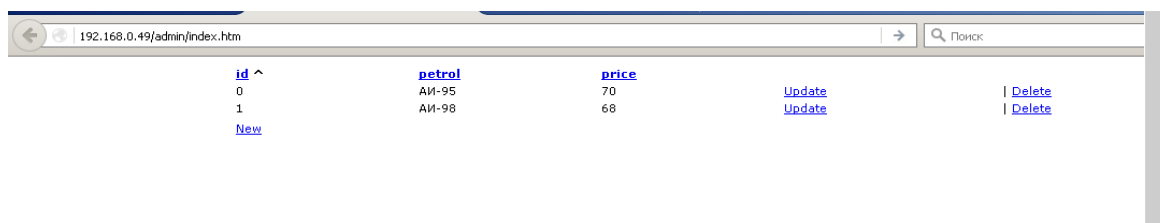
### Тестирование работоспособности ИС



The screenshot shows a web browser window with the address bar containing '192.168.0.49/client/index.htm'. Below the address bar, there is a table with the following data:

id	petrol	price
0	АИ-95	70
1	АИ-98	68

Рис. 18. Форма для клиента. Разработано авторами.



The screenshot shows an administrator interface with a table of fuel prices. The table has columns for 'id', 'petrol', and 'price'. Each row has 'Update' and 'Delete' buttons. There is also a 'New' link at the bottom left.

id	petrol	price	Update	Delete
0	АИ-95	70	Update	Delete
1	АИ-98	68	Update	Delete

Рис. 19. Форма для администратора. Разработано авторами.

### Список литературы

1. Воронцов Ю.А., Козинец А.В. Стандарты веб-сервисов для создания распределенных информационных систем / Ю.А.Воронцов, А.В.Козинец, // Век качества. 2015. №3. С. ...
2. Воронцов Ю.А., Козинец А.В. Пример построения распределенной информационной системы на AJAX с использованием PHP и IIS (Internet Information Services) / Ю.А.Воронцов, А.В.Козинец, // Век качества. 2016. №2. С. ...
3. Введение в виртуализацию [Электронный ресурс]. – Режим доступа: [http://www.team.ru/virt\\_intro.php](http://www.team.ru/virt_intro.php) (дата обращения 11.04.16г.).

4. Компьютерный информационный портал «OS Zone» - Установка и настройка платформы виртуализации Oracle VM VirtualBox [Электронный ресурс]. – Режим доступа: <http://www.oszone.net/13051> (дата обращения 11.04.16г.).
5. Установка VirtualBox [Электронный ресурс]. – Режим доступа: [http://itandlife.ru/technology/emulation\\_and\\_virtualization/virtualbox-setup-and-settings/2147483644/](http://itandlife.ru/technology/emulation_and_virtualization/virtualbox-setup-and-settings/2147483644/) (дата обращения 11.04.16г.).
6. Настройка сети в VirtualBox [Электронный ресурс]. – Режим доступа: <http://rus-linux.net/MyLDP/vm/VirtualBox-networking.html> (дата обращения 11.04.16г.).
7. Блог Максима Зайцева - Установка Денвера (локального сервера) на компьютер [Электронный ресурс]. – Режим доступа: <http://1zaicev.ru/ustanovka-denvera-lokalnogo-servera-na-kompyuter/> (дата обращения 19.10.16г.).
8. Architecture of user applications for network with mobile nodes / Vorontsov Y.A., Farkhadov M.P., Blinova O.V., Abramenzkov A.N. // 18-я международная конференция «Распределенные компьютерные и коммуникационные сети: управление, вычисление, связь» (DCCN-2015). С 460-465.

### **Sample web-service using Apache and MySQL**

***Vorontsov Yuri Alexeevich***

*Head of the Department of information systems,*

*Professor, Doctor of technics,*

*Moscow Technical University of Communications and Informatics*

*#32, Narodnogo Opolcheniya street, Moscow, 123993, Russian Federation*

[\*yvorontsov.1943@mail.ru\*](mailto:yvorontsov.1943@mail.ru)

***Kozinets Arthur Valerievich***

*Head of laboratories of the Department of information systems  
Moscow Technical University of Communications and Informatics  
#32, Narodnogo Opolcheniya street, Moscow, 123993, Russian Federation*  
[kozinets@mtuci2.ru](mailto:kozinets@mtuci2.ru)

**Abstract.** The article describes an example of building a web-service. For the implementation used the scripting language PHP as well as the Apache web server. Data is stored in a database via MySQL. Virtualization is applied when building model of information system.

**Key words:** distributed information systems; web services; PHP; Apache; MySQL.

#### REFERENCES

1. Vorontsov Yu.A., Kozinets A.V. Standarty veb-servisov dlya sozdaniya raspredeleennykh informatsionnykh sistem / Yu.A.Vorontsov, A.V.Kozinets, // Vek kachestva. 2015. №3. S. ...
2. Vorontsov Yu.A., Kozinets A.V. Primer postroeniya raspredelelennoy informatsionnoy sistemy na AJAX s ispol'zovaniem PHP i IIS (Internet Information Services) / Yu.A.Vorontsov, A.V.Kozinets, // Vek kachestva. 2016. №2. S. ...
3. Vvedenie v virtualizatsiyu [Elektronnyy resurs]. – Rezhim dostupa: [http://www.team.ru/virt\\_intro.php](http://www.team.ru/virt_intro.php) (data obrashcheniya 11.04.16g.).
4. Komp'yuternyy informatsionnyy portal «OS Zone» - Ustanovka i nastroyka platformy virtualizatsii Oracle VM VirtualBox [Elektronnyy resurs]. – Rezhim dostupa: <http://www.oszone.net/13051> (data obrashcheniya 11.04.16g.).

5. Ustanovka VirtualBox [Elektronnyy resurs]. – Rezhim dostupa: [http://itandlife.ru/technology/emulation\\_and\\_virtualization/virtualbox-setup-and-settings/2147483644/](http://itandlife.ru/technology/emulation_and_virtualization/virtualbox-setup-and-settings/2147483644/) (data obrashcheniya 11.04.16g.).
6. Nastroyka seti v VirtualBox [Elektronnyy resurs]. – Rezhim dostupa: <http://rus-linux.net/MyLDP/vm/VirtualBox-networking.html> (data obrashcheniya 11.04.16g.).
7. Blog Maksima Zaytseva - Ustanovka Denvera (lokal'nogo servera) na komp'yuter [Elektronnyy resurs]. – Rezhim dostupa: <http://1zaicev.ru/ustanovka-denvera-lokalnogo-servera-na-kompyuter/> (data obrashcheniya 19.10.16g.).
8. Architecture of user applications for network with mobile nodes / Vorontsov Y.A., Farkhadov M.P., Blinova O.V., Abramenzkov A.N. // 18-ya mezhdunarodnaya konferentsiya «Raspredeleynnye komp'yuternye i kommunikatsionnye seti: upravlenie, vychislenie, svyaz'» (DCCN-2015). S 460-465.