

УДК 004.021

РАЗРАБОТКА ВСТРАИВАЕМОГО МОДУЛЯ УЧЕТА ЗАЯВОК УЧАСТНИКОВ КОНКУРСА ГРАФИКИ НА ОСНОВЕ ВЕБ-ТЕХНОЛОГИЙ

Лобашов А.И., Бужинская Н.В.

Нижнетагильский государственный социально-педагогический институт (филиал)
ФГАОУ ВО «Российский государственный профессионально-педагогический университет»

Аннотация:

Сайт является визитной карточкой любой организации. С помощью сайта пользователи узнают интересующую их информацию, задают вопросы, читают отзывы. Эволюция языков веб-программирования привела к открытию новых возможностей. В настоящее время можно не только презентовать информацию об организациях в глобальной сети, но и автоматизировать процессы различной направленности. Актуальным становится использование возможностей языка веб-программирования для разработки базы данных, назначением которой является хранение и обработка информации. Автоматизированный прием онлайн-заявок даёт возможность структурировать поступающие данные, вести их учет и статистику. В статье рассматриваются вопросы разработки встраиваемого модуля для обработки персональных данных участников Международного конкурса компьютерной графики, который ежегодно проводится в филиале РГППУ г. Нижнего Тагила. Для разработки данного модуля использовались HTML, jQuery, PHP. База данных создавалась с помощью MySQL. Данный процесс можно представить в виде реализации трех взаимосвязанных этапов: создание базы данных для хранения и учета заявок, создание форм для приема заявки и вывода информации. Каждый этап характеризуется своими целями и задачи, однако переход на следующий этап невозможен без решения задачи предыдущего.

Ключевые слова: веб-программирование, форма, онлайн-заявки, глобальные сети, модуль

**EMBEDDED DEVELOPMENT MODULE OF ACCOUNTING OF DEMANDS OF
PARTICIPANTS OF COMPETITION SCHEDULES WITH THE USE OF WEB
TECHNOLOGIES**

Lobashov A. I., Buzhinskaya N. V.

Nizhny Tagil state socio-pedagogical Institute (branch) of the Russian state vocational pedagogical university

Annotation:

The site is a business card of any organization. With the help of the site, users will find out the information they are interested in, ask questions, read reviews. The evolution of web programming languages has led to the discovery of new features. At present, it is possible not only to present information about organizations in the global network, but also to automate processes of various types. It becomes relevant to use the capabilities of the web programming language to develop a database, the purpose of which is to store and process information. Automated acceptance of online applications allows you to structure incoming data, keep their records and statistics. The article deals with the development of an embedded module for processing personal data of the participants of the International Competition for Computer Graphics, which is held annually at the branch of the RGPU of Nizhny Tagil. HTML, jQuery, PHP were used to develop this module. The database was created using MySQL. This process can be represented as the implementation of three interrelated steps: the creation of a database for storing and recording applications, the creation of forms for receiving an application and outputting information. Each stage is characterized by its goals and objectives, but the transition to the next stage is impossible without solving the problem of the previous one.

Keywords: web programming, form, online applications, global networks, module

Общая постановка задачи

Ежегодно в филиале РГППУ г. Нижнего Тагила проводится международный конкурс компьютерной графики. В этом году было принято решение о разработке встраиваемого модуля, предназначенного для хранения и обработки on-line заявок участников.

Разработку встраиваемого модуля стоит поделить на несколько этапов.

Этап 1. Создание базы данных для хранения и учёта заявок.

Этап 2. Создание формы для реализации приёма заявки.

Этап 3. Создание формы для вывода информации.

В основе разработки базы данных лежит определение её структуры. База данных должна быть нормализованной, а автоматизация реализована с помощью организации связей между таблицами. На этом этапе необходимо учесть, что все таблицы должны соотноситься только с одной темой, а каждое поле должно соответствовать заполняемому данным. Кроме

того, важно определить первичные ключевые поля с автоматически заполняемыми значениями. При этом должны быть выбраны место расположения базы данных с подходящей системой управления.

Для разработки формы следует определить её конструкцию с указанием многократно используемых объектов. Установить переменные, способы их передачи и подключаемые функции. Также должна быть создана основа формы на языке HTML.

Формы должна иметь возможность быть встроенными в сайт международного конкурса компьютерной графики. Следовательно, для решения данной задачи необходимо подключить стилевые таблицы, специальным файлом, соответствующие графическому оформлению сайта. Формы должны передавать в базу данных информацию посредством языка PHP. Схема работы модуля представлена на рисунке 1.

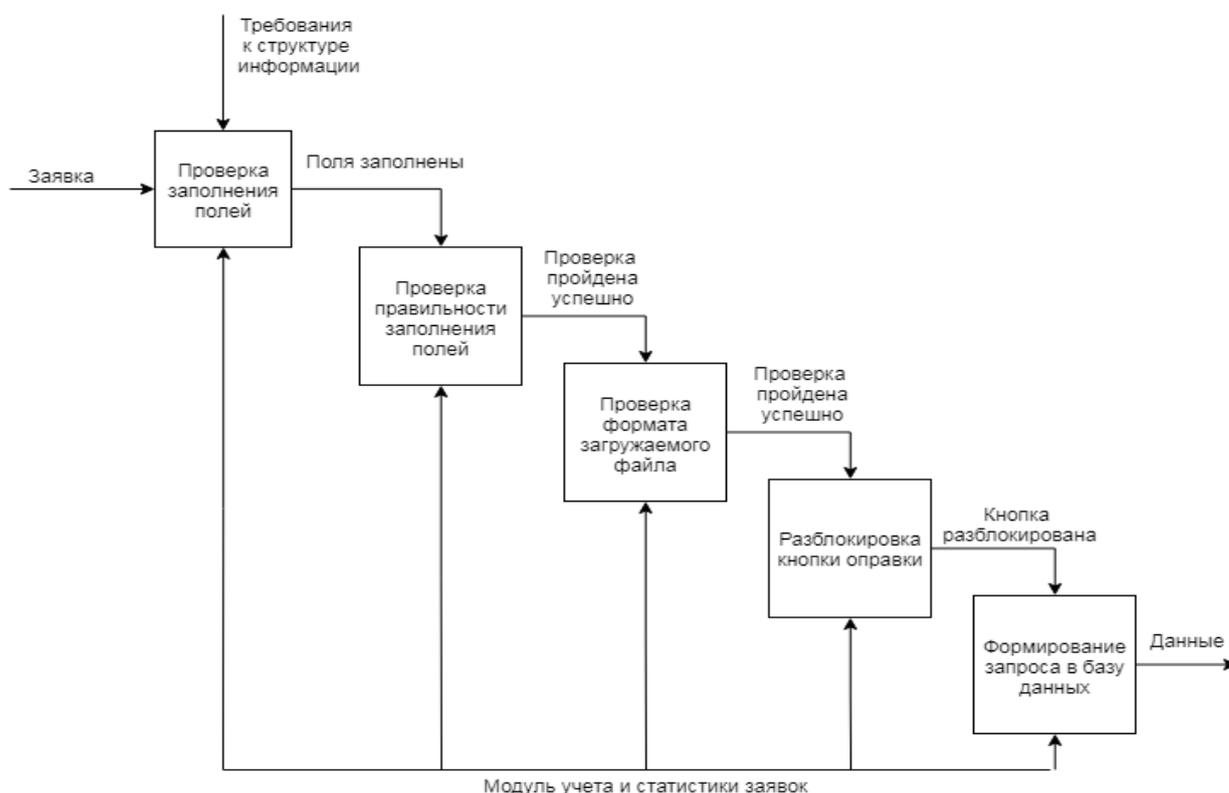


Рис. 1. Диаграмма работы модуля

Решение поставленной задачи

Этап 1. Разработка базы данных

Сама база данных имеет имя Zaiavki и состоит из четырех таблиц, связанных между собой.

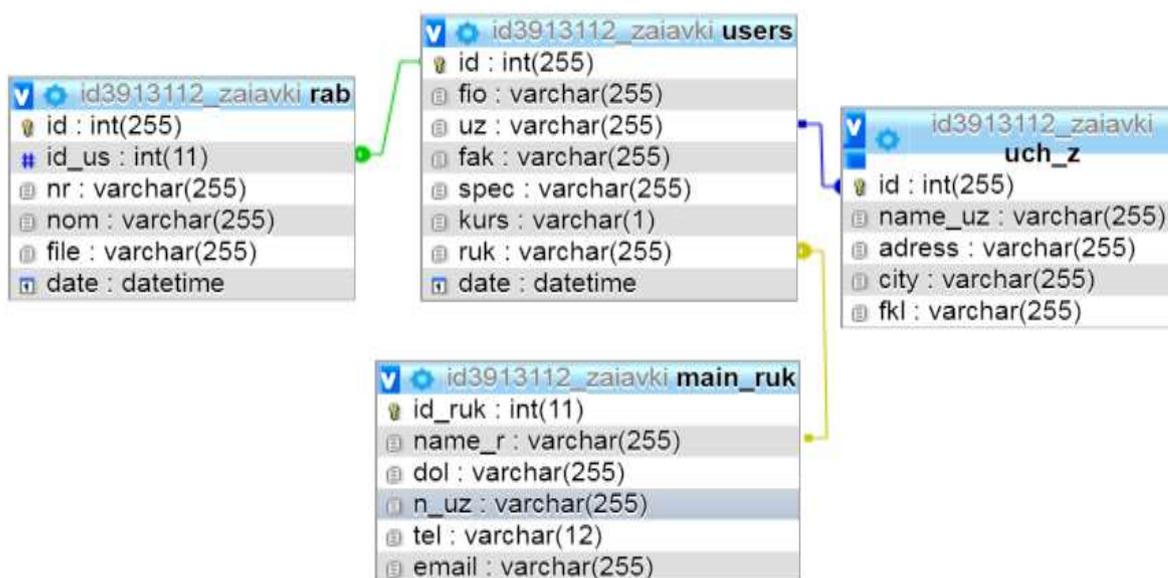


Рис. 2. Схема базы данных

Первая таблица хранит в себе сведения об участнике конкурса и именуется Users. Она включает: id – уникальный идентификатор каждого заявителя, fio – фамилия имя отчество участника конкурса, uz – название учебного заведения, spec – специальность на которой обучается участник, kurs – курс на котором обучается участник, ruk – фамилия имя отчество руководителя, fak – факультет, date – дата и время приёма заявки.

Вторая таблица содержит информацию о работах участника конкурса под названием Rab. Поля: id – идентификатор каждой работы, id_us – идентификатор автора работы, nom – номинация, nr – название работы, file – путь и измененное имя загруженного файла.

Третья таблица содержит данные об учебном заведении под названием Uch_uz. Поля: id – идентификатор каждого учебного заведения, name_uz – название учебного заведения, city – город, adress – адрес учебного заведения, fkl – факультет.

Четвертая таблица содержит данные о руководителе имеет наименование Main_ruk. Поля: id – идентификатор каждого руководителя, name_r – фамилия имя отчество руководителя, dol – должность, n_uz – название учебного заведения, email – электронная почта руководителя, tel – контактный телефон.

В каждой таблице поля id – это ключевые поля с автоматическим заполнением и типом integer. Поле date для корректной передачи имеет тип datetime. Все остальные поля имеют тип varchar с длиной 255 символов и со сравнением кодировки utf8_unicode_ci.

Этап 2. Разработка формы заявки

Форма подачи заявки участника состоит из HTML-основы, PHP- кода и jQuery-кода. Оформление и некоторые особенности задаются каскадными стилевыми таблицами.

HTML-основа состоит из: текстовых полей для заполнения (input type="text"), поля с выпадающим списком для выбора номинации (select), поля выбора файла для загрузки файла на сервер (input type="file"), элемента checkbox для соглашения с политикой конфиденциальности сайта, ссылок на пользовательское соглашение и на политику конфиденциальности (a href), а также кнопки для отправки формы (input type="submit").

Для возможности отправки файла на сервер и сведений в базу данных форма должно иметь тип enctype="multipart/form-data" и метод передачи method="post" [3].

Рассмотрим фрагмент кода, представленный в листинге 1.

Листинг 1

```
<form name="register" action="zaia.php" method="post" enctype="multipart/form-data">
<p><label> Номинация<br>
<select name="nom" class="input" id="nom" ><option value="Цифровое
фото">Цифровое фото</option><option
value="Коллаж/Фотомонтаж">Коллаж/Фотомонтаж</option>... </select></label></p>
<p><label> Название работы<br>
<input type="text" name="nr" value="" size="20" class="input" id="nr"
onkeyup="checkParams()" value="" autocomplete="on"></label></p>
...
<p><label> Загрузите файл <input type="file" name="image" class="button"
/></label></p>
<p>(поддерживаются форматы jpg,png,gif,bmp,jpeg,tiff)</p>
<hr align="center" width="500" size="2" />
<p><a href="https/..."></a></p>
<p><a href="https://..."> Политика конфиденциальности </a></p>
<p><label> Согласен с политикой конфиденциальности сайта <input type="checkbox"
name="check" id="check" value="1" required></label></p>
...
<input type="submit" name="register" id="submit" value="Отправить" disabled />
autocomplete="on" – элемент автозаполнения включен
onkeyup="checkParams()" – элемент для запуска jQuery – кода.
```

Вид формы представлен на рисунке 4.

Рис. 3. Вид формы заявки

Изначально кнопка отправки отключена (disabled). Такая особенность задействована для реализации jq-скрипта, который проверяет заполненность всех полей, и CSS-кода, который проверяет соглашение заполняющего с политикой конфиденциальности [2]. Это исключает возможность отправки неполной формы. Если все поля заполнены корректно, то кнопка становится доступной.

Рассмотрим CSS-код, представленный в листинге 2.

Листинг 2

```
<style type="text/css">
  form:invalid input[type="submit"] { opacity: 0.25;}
</style>
```

Данный код позволяет сделать кнопку прозрачной.

Рассмотрим фрагмент кода jQuery.

Листинг 3

```
<script>
function checkParams() {
  var fio = $('#fio').val();
```

```

var email = $('#email').val();
...
if($('input').length != 0 && email.length != 0 && ...) {
    $('#submit').removeAttr('disabled');
} else {
    $('#submit').attr('disabled', 'disabled');}
</script>

```

Данный код проверяет длину, заранее определенных переменных, которые соотносятся с текстовыми полями. Если все поля заполнены корректно, то выключается атрибут disabled на кнопке отправки формы.

PHP-код обрабатывает форму путем подключения к базе данных, созданием SQL-запросов и отправки переименованного файла на сервер [4].

В начале инициализируем файл connection.php для подключения к базе данных и определим нужные переменные, передаваемые методом POST [5].

Листинг 4

```

<?php require_once("scripts/connection.php");?>
<?php
if(isset($_POST["register"])){
    $fio = htmlspecialchars(trim($_POST['fio']));
    $email = htmlspecialchars(trim($_POST['email']));

```

Далее выполняем запросы к таблицам подключенной базы данных и загружаем файл на сервер, с предварительной проверкой расширения и переименованием.

Листинг 5

```

$sql = "INSERT INTO users (fio, uz, spec, kurs, ruk, fak, date) VALUES ('$fio', '$uz',
'$spec', '$kurs', '$ruk', '$fak', NOW())";
$result=mysqli_query($con, $sql);
...
$filetypes =
array('.jpg', '.JPG', '.Jpg', '.gif', '.GIF', '.Gif', '.bmp', '.BMP', '.Bmp', '.png', '.PNG', '.Png', 'TIFF', 'Tiff', 'tiff', 'JPE', 'Jpeg', 'jpeg');
$upload = 'images/';
$filename = $_FILES['image']['name'];
$ext = substr($filename, strpos($filename, '.'), strlen($filename)-1);
if(!in_array($ext,$filetypes))
    die ('<script>window.location.href = "error1.html"</script>');
if(move_uploaded_file($_FILES['image']['tmp_name'],$upload.uniqid('file_').$ext))

```

```
...
$file='images/'.uniqid('file_'. $ext);
```

```
...
```

Переменная `$filetypes` хранит в себе массив с возможными расширениями, загружаемых файлов. Далее идет проверка расширения загружаемого файла. Если расширение не равно одному из значений массива соединение приостанавливается и осуществляется переход на страницу ошибки. Если с расширением все в порядке, то имя файла заменяется на сгенерированное с помощью функции `uniqid`.

Функция `move_uploaded_file` перемещает переименованный файл в заданную директорию.

Этап 3. Создание формы для вывода информации

В данной форме будет осуществляться поиск участников по их именам. Для этого будет динамически генерироваться таблица с возможными значениями.

Рассмотрим HTML-код формы.

Листинг 5

```
<form name="form" id="form" action="names.php" method="post" style="margin:0px">
<label>Введите часть имени<input type="text" name="fio" value="" size="32"
style="width:100%" /></label>
<label><input type="submit" name="names" value="Найти" /> </label>
<label><input type="button" value="На главную"
onClick='location.href="https://ntgspigraf.000webhostapp.com"'></label>
```

Форма состоит из одного текстового поля и кнопок «Найти» и «На главную».

Листинг 6

```
$fio = htmlspecialchars(trim($_POST['fio']));
if(!empty($fio)){
$sql="SELECT fio,uz,fak,spec,city,ruk
FROM users, uch_z
where users.uz=uch_z.name_uz AND fio like '%" . mysqli_escape_string($con,$fio) . '%"
GROUP BY fio";
```

Первоначально определим переменную `$fio`, которая будет означать Ф.И.О. участника конкурса.

`SELECT` определяет имена столбцов, подключаемых к выборке из базы данных.

`FROM` определяет таблицы, подключаемые к выборке.

`WHERE` задает ограничение на строки табличного выражения из предложения.

like – это оператор SQL устанавливающий соответствие символьной строки с шаблоном.

mysqli_escape_string(\$con,\$fio) – это функция используется для создания допустимых в SQL строк, которые можно использовать в SQL выражениях. Заданная строка кодируется в экранированную SQL строку, используя текущий набор символов подключения.

GROUP BY объединяет ряды, имеющие одинаковое свойство с применением агрегатных функций.

В дальнейшем нам нужно создать заголовок таблицы и саму, динамически заполняемую, таблицу.

Рассмотрим фрагмент PHP-кода заголовка таблицы.

Листинг 7

```
echo '<thead>';
echo '<tr>';
echo '<th>Ф.И.О.</th>';
echo '<th>Город</th>';
echo '<th>Учебное заведение</th>';
echo '<th>Специальность</th>';
echo '<th>Факультет</th>';
echo '<th>Руководитель</th>';
echo '</tr>';
echo '</thead>';
```

Тем самым будет создано шесть заголовков столбцов.

Рассмотрим фрагмент PHP-кода вывода табличных данных.

Листинг 8

```
while($data = mysqli_fetch_array($result))
{   echo '<tr>';
    echo '<td>' . $data['fio'] . '</td>';
        echo '<td>' . $data['city'] . '</td>';
    echo '<td>' . $data['uz'] . '</td>';
    echo '<td>' . $data['spec'] . '</td>';
        echo '<td>' . $data['fak'] . '</td>';
        echo '<td>' . $data['ruk'] . '</td>';
    echo '</tr>';}
else {echo'Записи о таком пользователе нет. Попробуйте изменить запрос';} } else {
echo "Введите часть фио!";} }
```

Заполнение таблицы из базы данных производится циклом `while ($data = mysqli_fetch_array($result)) [1]`.

Где `mysqli_fetch_array` – функция которая выбирает одну строку из результирующего набора и помещает ее в ассоциативный массив. Данный цикл будет выполняться до тех пор, пока в базе есть данные удовлетворяющие запросу `$result`.



Введите часть имени

Найти На главную

Рис. 4. Страница поиска по имени



Ф.И.О.	Город	Учебное заведение	Специальность	Факультет	Руководитель
Леонардо да Винчи	Флоренция	ФСКМ	Художник	ФСБ	Кац Семен Петрович

Рис. 5. Создаваемая таблица

Вывод

Реализуемые формы позволяют производить системный учёт заявок участников их работ, руководителей и учебных заведений. Эти формы и базу данных можно встроить в сайт или же разместить на стороннем хостинге. При их работе были использованы скриптовый язык PHP для отправки сведений в базу данных, JavaScript – для обеспечения интерактивности разрабатываемых компонентов, CSS – для оформления внешнего вида форм, jQuery – для создания обработчиков событий. В процессе разработки также необходимо учитывать, что встраиваемый модуль будет располагаться на сайте конкурса графики, поэтому его внешний вид должен соответствовать дизайну сайта.

Список литературы:

1. Введение в jQuery [Электронный ресурс]. URL: <https://html5book.ru/vvedenie-v-jquery/> (Дата обращения 04.01.2019).
2. Веру Л. Секреты CSS. Идеальные решения ежедневных задач страниц [Электронный ресурс].

URL:

<https://orkhanalyshov.com/media/HTMLCSS/LeaVerouCSSsecretsBetterSolutionsToEverydayWebDesignProblems.pdf> (Дата обращения 10.01.2019).

3. Гасанов Э. В. Практикум по созданию Интернет-проектов. Основы языка программирования PHP. Часть 1 [Электронный ресурс]. URL: http://biblioclub.ru/index.php?page=book_view_red&book_id=230535 (Дата обращения 23.12.2018).

4. МакГрант М. PHP7 для начинающих с пошаговыми инструкциями. М: ЭКСМО-ПРЕСС, 2018. 258 с.

5. HTML, CSS, JavaScript, jQuery, HDOM, AJAX [Электронный ресурс]. URL: <http://www.wisdomweb.ru/JQ/jquery-first.php> (Дата обращения 13.02.2019).