

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего
профессионального образования

«Томский государственный университет систем управления и
радиоэлектроники». (ТУСУР)

УТВЕРЖДАЮ

Заведующий кафедрой
«Управление инновациями»

_____ /А.Ф.Уваров
(подпись) (ФИО)
" _____ " _____ 2012 г.

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ К САМОСТОЯТЕЛЬНОЙ РАБОТЕ

по дисциплине

Программирование

Составлена кафедрой

«Управление инновациями»

Для студентов, обучающихся

по направлению подготовки 220600.62 «Инноватика»

по специальности 220601.65 «Управление инновациями»

Форма обучения

очная

Составитель

к.т.н.,

Титков Антон Вячеславович

" 20 " мая 2012 г

Томск 2012 г.

СОДЕРЖАНИЕ

1. ВВЕДЕНИЕ	3
2. ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ №1	4
3. ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ №2	5
4. ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ №3	8
12. РЕКОМЕНДУЕМАЯ ЛИТЕРАТУРА	11

Введение

Изучение дисциплины «Программирование» имеет важное значение в специальной подготовке студентов по направлению «Инноватика» и специальности «Управление инновациями» и предоставляет им инструментарий презентации в интернет своих разработок, своей работы, деятельности инновационной фирмы путем размещения информации на веб-сайтах.

Цель данного пособия состоит в выработке навыков в программировании веб-сайтов и их разработке. Для полноценного понимания и усвоения материала необходимо предварительно изучить дисциплину "Информатика".

Для углубленного изучения и освоения материала целесообразно выполнение лабораторных работ, наряду с другими различными формами обучения студентов: тесты, задачи, упражнения, которые используются при проведении практических занятий в университете, выполнении контрольных и аудиторных работ, а также при самостоятельном изучении данных дисциплин.

Одним из наиболее интенсивных способов изучения дисциплины является самостоятельное выполнение лабораторных работ, на которых вырабатываются навыки построения веб-сайтов.

Предлагаемые лабораторные работы позволяют глубже освоить теоретические и практические вопросы, понять принципы программирования веб-сайтов и научиться создавать свои интернет приложения.

Индивидуальное задание №1. Работа с хостингом.

Цель: научиться работать с хостингами.

Задание: создать веб-сайт, используя html, и разместить его на бесплатном хостинге.

Ход работы.

Требуется создать веб-сайт, состоящий из нескольких страниц. Первым делом необходимо зарегистрироваться на бесплатном хостинге. В качестве бесплатного хостинга может быть <http://zetahost.ru/>.

Для регистрации хостинга на <http://zetahost.ru/> потребуется перейти по ссылке. В поле, где написано домен пишем имя для вашего сайта и если такого имени нет, то регистрируем его (рис. 1).



The image shows a registration form for a domain. The form is set against a light beige background with a dark border. It contains the following fields and elements:

- Ваш домен:** A text input field containing "bestsite" and a dropdown menu showing ".1cs.su".
- Название проекта:** A text input field containing "Your Site Name".
- Ваш E-mail:** A text input field containing "thisis@your.email".
- E-mail еще раз:** A text input field containing "thisis@your.email".
- Пароль:** A text input field with masked characters "*****".
- Пароль еще раз:** A text input field with masked characters "*****".
- reCAPTCHA:** A section with the text "Введите то, что видите:" and a text input field containing "SOLITARY, idsmark". To the right of the input field are icons for refresh, volume, and mute. Further right is the reCAPTCHA logo with the text "reCAPTCHA™ stop spam. read books."
- Зарегистрироваться:** A button at the bottom of the form.
- Вспомнить пароль** and **Войти:** Two links at the bottom of the form.

Рис. 1 – Регистрация домена

Завершив регистрацию, на указанный почтовый адрес придет письмо с логином и паролем. Перейдите по адресу <http://cp.zetahost.ru> и введите логин и пароль из письма и попадете административный интерфейс (рис 2).

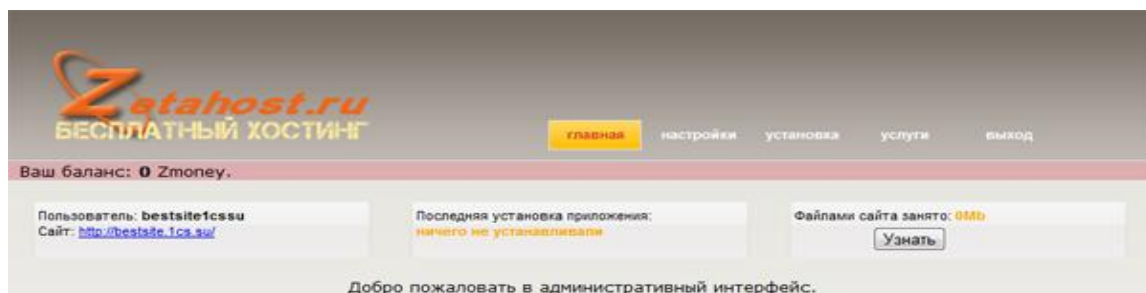


Рис. 2 – Личный кабинет

Теперь можно перейти на созданный сайт по ссылке <http://bestsite.1cs.su/>, где bestsite – имя вашего домена. Здесь появится автоматически сгенерированная страница. Вполне возможно, что вы увидите надпись «Если Вы открыли свой сайт и видите эту страницу - это означает, что Ваш сайт еще не до конца создан. Попробуйте обновить страницу через несколько минут.». Подождите некоторое время пока она изменится (это занимает до нескольких часов).

Для того чтобы разместить ваш сайт на хостинге необходимо воспользоваться ftp-клиентом. Для этих целей подойдет Total Commander. Требуется установить его на компьютере и нажать на «Соединиться с ftp-сервером» в меню «Сеть» либо нажать комбинацию клавиш ctrl+f. В появившемся окне нужно нажать на кнопку «Добавить». Заполните поля «Имя соединения», «Сервер», «Учетная запись» и «Пароль». Здесь «Имя соединения» может быть любым, остальные поля заполняются в соответствии с параметрами доступа по ftp из страницы настройки (рис. 3).

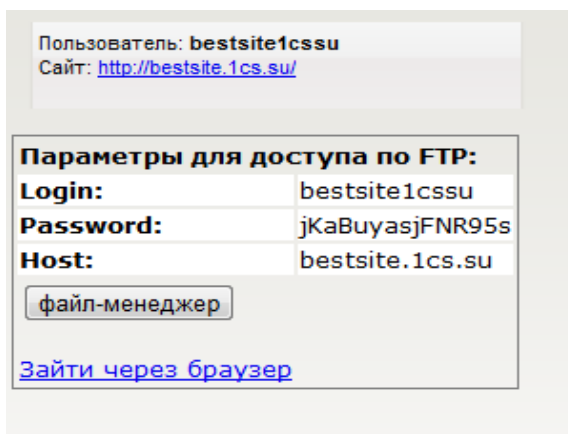


Рис. 3 – Настройки FTP

Подключитесь к ftp, затем перейдите в папку public_http и загрузите туда сайт. Перейдите по ссылке <http://bestsite.1cs.su/> и увидите ваш сайт.

Индивидуальное задание №2. Создание изображений в HTML5 .

Цель: научиться работать с canvas в html5, а также создавать анимацию.

Задание: нарисовать анимированный рисунок, используя технологию canvas.

Ход работы.

Для работы с изображениями в html5 используются две технологии Canvas и SVG. Первая используется для создания растровой графики, а вторая – векторной. Целью индивидуального задания является ознакомление с технологией html5 и создание анимированного изображения при помощи Canvas.

В качестве примера, воспользуемся изображением Марио (рис. 4).

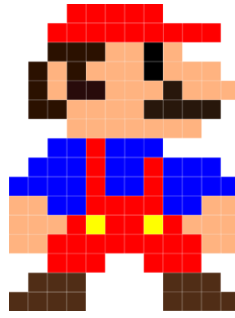


Рис. 4 – пример изображения.

Из примера видно, что изображение состоит из квадратов разного цвета, поэтому можно составить матрицу этого изображения. Элементом матрицы может быть число от 0 до 7, где число это цвета квадрата. Для данного изображения она будет выглядеть следующим образом:

```
var mario = [  
  [0,0,0,1,1,1,1,0,0,0,0],  
  [0,0,1,1,1,1,1,1,1,0],  
  [0,0,6,6,6,7,7,5,7,0,0],  
  [0,6,7,6,7,7,7,5,7,7,0],  
  [0,6,7,6,6,7,7,7,6,7,7],  
  [0,6,6,7,7,7,6,6,6,6,0],  
  [0,0,0,7,7,7,7,7,7,0,0],  
  [0,0,3,3,1,3,3,3,0,0,0],  
  [0,3,3,3,1,3,3,1,3,3,0],  
  [3,3,3,3,1,3,3,1,3,3,3],  
  [7,7,3,3,1,1,1,1,3,3,7],  
  [7,7,7,3,4,1,1,4,3,7,7],  
  [7,7,3,3,1,1,1,1,3,3,7],  
  [0,0,3,3,1,0,0,1,3,3,0],  
  [0,6,6,6,0,0,0,0,6,6,6],  
  [6,6,6,6,0,0,0,0,6,6,6]];
```

Необходимо инициализировать в самом начале переменные:

```
var canvas; // элемент canvas.  
var ctx; // контекст canvas.  
var cellSize = 24; // размер квадрата.  
var animate = 0; // переменная для анимации персонажа.
```

В тег body добавляем событие `onload="drawCanvas()"`, а также тег `canvas` с атрибутом `id="drawing"`.

В функции `drawCanvas` получаем элемент `canvas` и его контекст, так же устанавливаем размеры канвы, на которой будет будущее изображение, и при помощи функции `setInterval` делаем заикливание для анимации. В функцию передаем два параметра: функцию `drawHero` и время задержки, равное 500 миллисекунд.

```
function drawCanvas(){  
  canvas = document.getElementById("drawing");  
  ctx = canvas.getContext('2d');  
  canvas.width = 12 * cellSize;  
  canvas.height = 16 * cellSize;  
  gameLoop = setInterval(drawHero, 500);  
}
```

Внутри функции drawHero очищаем канву при помощи ctx.clearRect(0, 0, canvas.width, canvas.height) – очищаем прямоугольник, левый верхний угол которой расположен в точке (0; 0) и длина и высота которого canvas.width и canvas.height соответственно. Инкремент переменной animate нужен для изменения состояния персонажа. В нашем случае Марио должен стоять и время от времени приседать и моргать. Для этого зададим условие, которое будет изменять позицию марио при animate кратном числу 3. Также составим новую матрицу для изменения положения персонажа.

```
var marioSit = [
  [0,0,0,0,0,0,0,0,0,0],
  [0,0,0,1,1,1,1,0,0,0],
  [0,0,1,1,1,1,1,1,1,0],
  [0,0,6,6,6,7,7,5,7,0,0],
  [0,6,7,6,7,7,7,5,7,7,0],
  [0,6,7,6,6,7,7,6,7,7,7],
  [0,6,6,7,7,7,6,6,6,6,0],
  [0,0,0,7,7,7,7,7,7,0,0],
  [0,0,3,3,1,3,3,3,0,0,0],
  [0,3,3,3,1,3,3,1,3,3,0],
  [3,3,3,3,1,3,3,1,3,3,3],
  [7,7,3,3,1,1,1,1,3,7,7],
  [7,7,7,3,4,1,1,4,3,7,7],
  [7,7,3,3,1,1,1,1,3,7,7],
  [0,6,6,6,0,0,0,0,6,6,6],
  [6,6,6,6,0,0,0,0,6,6,6]];
```

Следом за условным оператором будет выполняться функция drawPix, куда будем передавать матрицу, описывающую персонажа. При достижении переменной animate значения 20, обнуляем ее.

```
function drawHero(){
  ctx.clearRect(0,0,canvas.width,canvas.height);
  animate++;
  if (animate % 3 == 0) {
    drawPix(marioSit);
  } else {
    drawPix(mario);
  }
  if (animate == 20) animate = 0;
}
```

Функция drawPix осуществляет перебор всех элементов матрицы при помощи двойного цикла for и в зависимости от значения элемента матрицы создает квадрат и закрашивает его нужным цветом. Так как черным цветом обозначены только глаза, можно заставить Марио моргать, при помощи тернарного оператора, т.е. если animate будет делиться на 5 без остатка, то вместо черного цвета установить цвет кожи.

```
function drawPix(arr) {
  for (var j = 0; j < 16; j++)
  for (var i = 0; i < 12; i++) {
    if (arr[j][i] == 0) continue;
    switch (arr[j][i]) {
      case 1:
        color = "#FF0000";
        break;
      case 3:
        color = "#0000FF";
        break;
```

```

case 4:
    color = "#F0FF00";
    break;
case 5:
    color = (animate % 5 == 0)? "#efbb84" :
"#000000";
    break;
case 6:
    color = "#9e3f00";
    break;
case 7:
    color = "#efbb84";
    break;
}
x = i * cellSize;
y = j * cellSize;
ctx.fillStyle = color;
ctx.fillRect(x, y, cellSize, cellSize);
}
}

```

Индивидуальное задание №3. Установка CMS на сервер.

Цель: научиться устанавливать и работать с CMS.

Задание: установить CMS на хостинг.

Ход работы.

В качестве хостинга будем использовать все тот же <http://zetahost.ru/>, так как на нем предоставляют доступ к базе данных.

Загружаем с официального сайта Joomla последнюю версию и загружаем их на хостинг по ftp. Для работы с Joomla необходима база данных. Ее можно создать в административной панели в меню настройки. После создания базы данных в настройках появятся параметры для работы с базой данных (рис 5.)

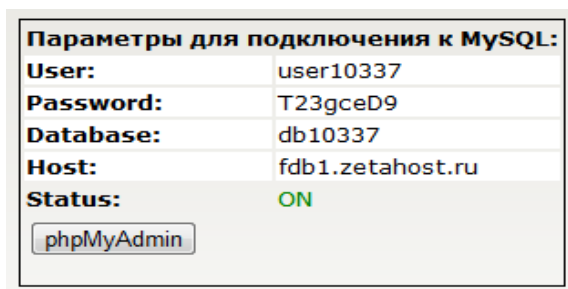


Рис. 5 - Параметры подключения к базе данных

Теперь заходим на сайт. На экране появится мастер установки Joomla. Первая страница предлагает выбрать язык установки (рис. 6).

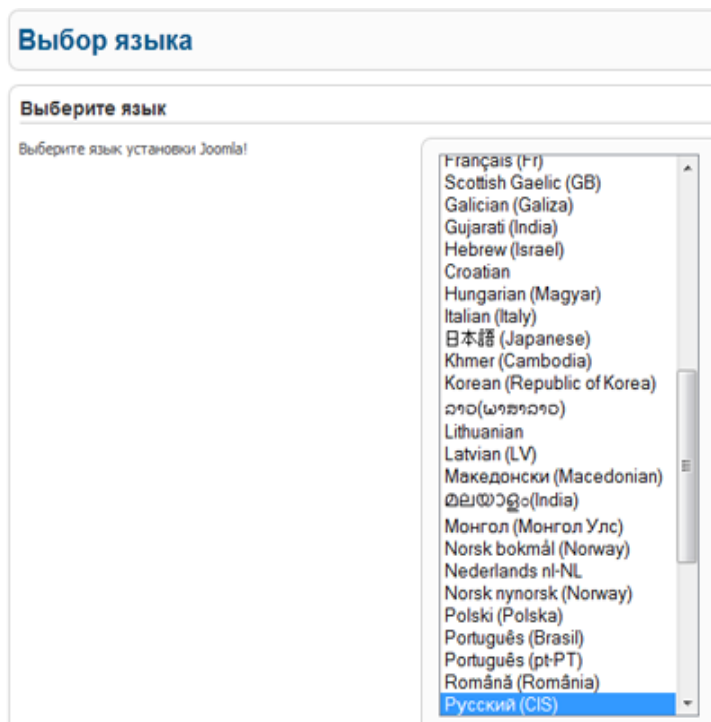


Рис. 6 – Выбор языка

Выбираем нужный язык и ждем далее. На странице «Начальная проверка» нужно, чтобы все установки в верхней таблице поддерживались. Если все хорошо, то ждем далее, попадаем на шаг с описанием лицензии, снова далее. Для конфигурации базы (рис. 7) данных берем настройки, которые получили выше.

Рис. 7 – Конфигурация базы данных

Шаг с конфигурацией ftp можно пропустить. На странице конфигурация сайта пишем его название и логин и пароль от административной панели. На последнем шаге необходимо удалить директорию installation.

Теперь можно еще раз зайти на сайт. Перед вами должна появиться главная страница (рис. 8).

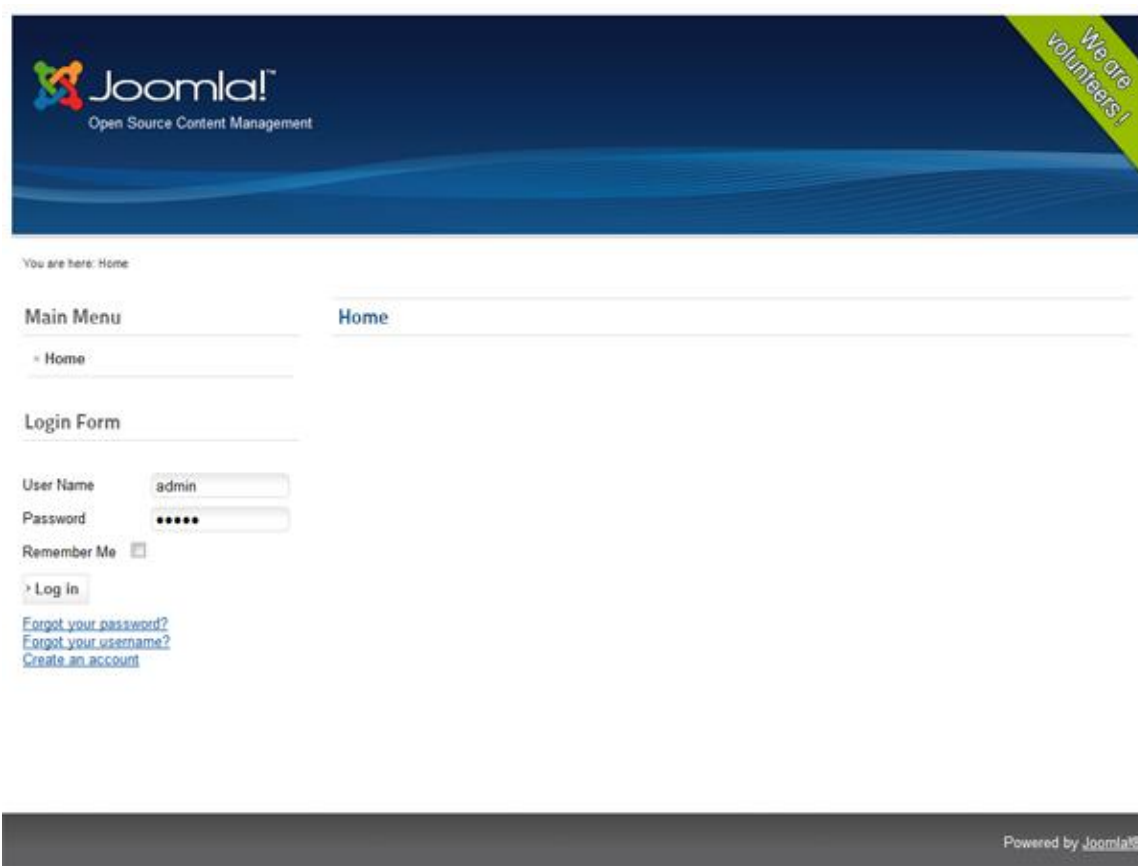


Рис. 8 – Главная страница

Чтобы перейти в панель администратора необходимо в строке адреса ввести «адрес_вашго_сайта/administrator/». В открывшемся окне необходимо ввести логин и пароль администратора, нажать кнопку «Login».

Разместите несколько новостей и добавьте их в меню.

Рекомендуемая литература

Основная литература:

1. Мировые информационные ресурсы : учебное пособие / С. Л. Миньков ; Федеральное агентство по образованию, Томский государственный университет систем управления и радиоэлектроники, Кафедра автоматизированных систем управления. - Томск : ТМЦДО, 2009. - 185 с.
2. Лаптев, Валерий Викторович. С++. Объектно-ориентированное программирование : учебное пособие / В. В. Лаптев. - СПб. : Питер, 2008. - 457[7] с.
3. Павловская, Татьяна Александровна. С/С++. Программирование на языке высокого уровня : Учебник для вузов / Т. А. Павловская. - СПб. : Питер, 2008. - 460[4] с.

Дополнительная литература

1. Практические занятия по PHP4 : Краткий курс / В. А. Будилов; Ред. С. Л. Корякин-Черняк. - СПб. : Наука и Техника, 2001. - 352 с.
2. Знакомьтесь: World Wide Web : Учебное пособие: Пер. с нем. / Матиас Нольден. - Киев : ВНУ, 1996. - 336 с.
3. JavaScript, XML и объектная модель документа : учебное пособие / В. А. Будилов. - СПб. : Наука и Техника, 2001. - 348[4] с.
4. Д. Скляр, А. Трахтенберг PHP. «Рецепты программирования PHP». Издательства: Русская Редакция, БХВ-Петербург, 2007 г., 736 стр.
5. М.Браун, Д.Ханикатт «HTML в подлиннике», издательство: ВНУ, 2002, 1024 стр., перевод с английского
6. Муссиано, Кеннеди «HTML и XHTML. Подробное руководство», издательство: Символ-Плюс, 2008г., 752 стр, перевод с английского.
7. PHP: hypertext preprocessor. URL: <http://php.net> (дата обращения 31.09.2011)
8. Флэнаган Д. «JavaScript. Подробное руководство» Издательство: Символ-Плюс, 2008г., 992 стр, перевод с английского.