

Федеральное государственное автономное  
образовательное учреждение  
высшего образования  
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Космических и информационных технологий  
институт

Вычислительная техника  
кафедра

УТВЕРЖДАЮ  
Заведующий кафедрой  
А. И. Легалов  
\_\_\_\_\_ подпись инициалы, фамилия  
« \_\_\_\_ » \_\_\_\_\_ 2016 г.

**БАКАЛАВРСКАЯ РАБОТА**

09.03.01 Информатика и вычислительная техника  
код и наименование специализации

Разработка веб-сервера для регистрации участников олимпиады ACM по  
программированию. Разработка пользовательского интерфейса  
тема

Пояснительная записка

Руководитель	_____	<u>ст.преп.</u>	<u>Л. В. Макуха</u>
	<small>подпись, дата</small>	<small>должность, ученая степень</small>	<small>инициалы, фамилия</small>
Выпускник	_____		<u>Д. Г. Слепцов</u>
	<small>подпись, дата</small>		<small>инициалы, фамилия</small>
Нормоконтролер	_____	<u>доцент, к.т.н.</u>	<u>В. И. Иванов</u>
	<small>подпись, дата</small>	<small>должность, ученая степень</small>	<small>инициалы, фамилия</small>

Красноярск 2016

Федеральное государственное автономное  
образовательное учреждение  
высшего образования  
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Космических и информационных технологий  
институт

Вычислительная техника  
кафедра

УТВЕРЖДАЮ  
Заведующий кафедрой

\_\_\_\_\_ А. И. Легалов  
подпись      инициалы, фамилия

« \_\_\_\_ » \_\_\_\_\_ 2016 г.

**ЗАДАНИЕ**

**НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ**

в форме \_\_\_\_\_ бакалаврской работы \_\_\_\_\_

Студенту Слепцову Денису Геннадьевичу

фамилия, имя, отчество

Группа КИ12-09Б Направление (специальность) 09.03.01

номер

код

Высокопроизводительные вычислительные системы на базе больших ЭВМ

наименование

Тема выпускной квалификационной работы

Разработка веб-сервера для регистрации участников олимпиады ACM по программированию. Разработка пользовательского интерфейса

Утверждена приказом по университету № \_\_\_\_\_ от \_\_\_\_\_

Руководитель ВКР Л. В. Макуха, старший преподаватель кафедры ВТ

инициалы, фамилия, должность, учёное звание и место работы

Исходные данные для ВКР Расписание соревнований, данные для регистрации, образец бейджиков

Перечень разделов ВКР анализ задания на выпускную квалификационную работу, разработка архитектуры и основных технических решений

Перечень графического материала структурные схемы Xmind, презентация в формате Power Point, демонстрационное видео

Руководитель ВКР

\_\_\_\_\_   
подпись

Л.В. Макуха

инициалы, фамилия

Задание принял к исполнению

\_\_\_\_\_   
подпись

Д. Г. Слепцов

инициалы, фамилия

«\_\_» \_\_\_\_\_ 2016 г.

## РЕФЕРАТ

Пояснительная записка содержит 51 страницу, 21 рисунок, 10 источников.

Объектом исследования является разработка и создание web-сайта.

При создании сайта были рассмотрены языки веб программирования (PHP, HTML, JavaScript, CSS). Также были рассмотрены различные системы управления содержимым, которые позволяют в упрощенной виде создавать, редактировать и управлять контентом на сайте. Существует множество разнообразных CMS, каждая из которых отличается какой-то своей особенностью, после анализа различных возможностей, наиболее оптимальной CMS была выбрана платформа WordPress.

Результатом данной работы является разработка веб-сервера, осуществляющего функцию информирования об олимпиаде по программированию ACM, проводимой в СФУ ИКИТ, регистрации на олимпиаду участников и упрощения работы организаторов олимпиады.

Итогом будет являться веб-сервер, на котором потенциальные участники олимпиады смогут просмотреть актуальную информацию, а также зарегистрироваться для участия в олимпиаде. Для организаторов олимпиады будут доступны просмотр списков, зарегистрированных участников, также списки оформленных на участников документов, которые автоматически заполняются при регистрации.

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	5
1 Аналитическая часть.....	7
1.1 Анализ задания на выпускную квалификационную работу.....	7
1.2 Анализ существующих веб-сайтов .....	8
1.2.1 НИИ «Высшая школа экономики.....	8
1.2.2 Онлайн-олимпиада «Юный предприниматель» .....	10
1.3 Выбор средств разработки .....	11
1.3.1 Анализ основных языков программирования.....	11
1.3.1.1 Язык программирования PHP .....	11
1.3.1.2 Язык программирования Ruby .....	14
1.3.1.3 Язык программирования Python .....	15
1.3.2 Язык программирования HTML.....	16
1.3.3 СУБД MySQL .....	17
1.3.4 Разметка стилей CSS.....	18
1.3.5 Язык программирования JavaScript .....	18
1.3.6 Обоснование выбора средств разработки .....	18
1.4 Техническое задание.....	19
1.4.1 Наименование работы .....	19
1.4.2 Функции и цели создания работы .....	20
1.4.2.1 Функции работы .....	20
1.4.2.2 Цель создания работы .....	20
1.4.3 Требования к дизайну веб-сервера.....	20
1.4.4 Требования к структуре веб-сервера .....	21
1.4.5 Требования к административной части веб-сервера.....	21
1.4.6 Требования к разделению доступа.....	21
1.4.7 Требования к информационному обеспечению.....	22
1.4.7.1 Требования к хранению данных .....	22
1.4.7.2 Требования к языкам программирования.....	22

1.4.8 Требования к программному обеспечению.....	22
1.4.8.1 Требования к программному обеспечению серверной части .....	22
1.4.8.2 Требования к клиентскому программному обеспечению .....	23
1.4.9 Требования к эргономике и технической эстетике .....	23
2 Разработка архитектуры и основных технических решений .....	24
2.1 Функциональная структура веб-сервера .....	24
2.2 Информационная структура веб-сервера .....	25
2.2.1 Диаграмма прецедентов .....	25
2.3 Структура веб-сервера.....	26
2.4 Описание создания страниц веб-сервера.....	28
2.5 Разработка функции регистрации .....	28
2.5.1 Структура функции регистрации .....	28
2.5.2 Программная реализация функции регистрации.....	29
2.5.3 Результат работы функции регистрации .....	34
2.6 Разработка личного кабинета .....	36
2.6.1 Структура личного кабинета .....	36
2.6.2 Программная реализация Личного кабинета.....	37
2.6.3 Результат работы Личного кабинета.....	40
2.7 Перенос веб-сервера на CMS.....	41
2.7.1 Описание установки CMS Wordpress.....	31
2.7.2 Настройка темы веб-сервера.....	42
2.7.3 Создание страниц веб-сервера.....	43
2.7.4 Создание новостей .....	44
2.7.5 Подключение плагинов .....	45
2.7.6 Регистрация участников.....	45
2.8 Запуск проекта.....	46
3 Руководство пользователя.....	47
3.1 Требования к программному и серверному обеспечению.....	47
3.2 Требования к клиентскому программному обеспечению.....	47
3.3 Перенос веб-сервера на хостинг СФУ .....	47

ЗАКЛЮЧЕНИЕ .....	49
СПИСОК СОКРАЩЕНИЙ.....	50
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ .....	51

## ВВЕДЕНИЕ

Характерными особенностями современного человечества является: воспроизводство, хранения, передача и обработка имеющейся и получаемой информации. Сейчас, обычные люди не в состоянии представить свою жизнь без плазменных телевизоров, смартфонов, персональных компьютеров, различных гаджетов, но главное без интернета.

Интернет с ног на голову перевернул человеческие представления о работе с информацией, да и с компьютером вообще. С развитием технологий гипертекстовой разметки в интернете на всемирную паутину «обрушилось» огромное количество различных сайтов по всевозможным тематикам.

Различные компании большие и маленькие стали заводить себе сайты, на которых они предлагают свои услуги, заключают сделки.

стали появляться различные форумы для общения, перерастающие со временем в огромные интернет порталы, на которых общаются люди с разных уголков планеты. Сегодня, можно не выходя из дома делать покупки, оплачивать счета, самообразовываться и само реализовываться в каких то сферах.

В наше время любые общественные мероприятия, конференции, лекции, олимпиады, которые проводятся с целью привлечения заинтересованных людей из различных мест проживания, должны иметь интернет ресурс, где у людей будет возможность записаться, зарегистрироваться и получить необходимую информацию о мероприятиях дистанционно, не выходя из дома.

Потребность в практическом применении дипломной работы заключается в том, что на сегодняшний день на базе Сибирского Федерального Университета Института Космических и Информационных Технологий проводится международная олимпиада по программированию АСМ. Но регистрация участников проходит затруднительно, так как каждый участник для записи должен прибыть на место проведения олимпиады. В связи с этим возникла необходимость в создании электронного интернет ресурса, через



который можно было бы осуществлять учет и регистрацию участников, поэтому тема работы достаточно актуальна.

Целью выпускной квалификационной работы является разработка данного веб-сервера.

Разработка веб-сервера должна включать в себя:

- Утверждение технического задания на разработку веб-сервера;
- Определение общей структурной схемы веб-сервера;
- Создание графических элементов, макета веб-сервера;
- Разработка программного кода, модулей и других необходимых элементов веб-сервера;
- Создание базы данных;
- Тестирование и размещение веб-сервера в сети.

## **1 Аналитическая часть**

### **1.1 Анализ задания на выпускную квалификационную работу**

Целью данной работы является разработка веб-сервера для информирования, регистрации участников олимпиады по программированию АСМ, проводимой в СФУ ИКИТ, а также упрощения работы организаторов олимпиады.

В данном веб-сервере должны быть решены следующие возможности:

- Публикация различных материалов по проведению олимпиады (новостей, объявлений, документов);
- Регистрация участников олимпиады;
- Личный кабинет участников;
- Внесение данных об участниках олимпиады в базу данных веб-сервера;
- Внесение изменений в данных об участниках, занесенных в базу данных;
- Автоматическое формирование шаблонов документов для участников и организационного комитета;
- Реализация разделения олимпиад.

Задачи:

- Анализ существующих аналогичных по назначению веб-серверов;
- Выбор средств разработки;
- Техническое задание.

## **1.2 Анализ существующих веб-серверов**

Этапы олимпиады по программированию АСМ проводят не только в Красноярске, но также в таких городах как Санкт-Петербург и Барнаул, однако они не имеют соответствующие интернет ресурсы, поэтому анализ веб-серверов будет проведен среди других различных олимпиад.

На сегодняшний день существует множество различных веб-серверов по проведению олимпиад с онлайн регистрациями.

Ниже приведены некоторые подобные веб-серверы:

- Национальный исследовательский университет «Высшая школа экономики»;
- Онлайн-олимпиада «Юный предприниматель»;
- Открытая Московская онлайн олимпиада по математике.


### **1.2.1 НИУ «Высшая школа экономики»**

На базе Национального исследовательского университета «Высшей школы экономики» ежегодно проводят олимпиаду для студентов и выпускников. Победители и призеры получают льготы при зачислении в НИУ ВШЭ на магистерские программы, соответствующие профилю олимпиады. В олимпиаде могут принять участие студенты последних курсов бакалавриата или специалитета, а также выпускники вузов независимо от гражданства. Олимпиада проводится по широкому кругу направлений и профилей, что позволяет учесть профессиональные интересы потенциальных участников олимпиады, ориентированных на обучение на программах магистерской подготовки и продолжение академической и профессиональной карьеры [1].

Для участия необходимо пройти онлайн регистрацию. На рисунке 1 приведен скриншот анкеты, которую необходимо заполнить потенциальному участнику.

**ВЫСШАЯ ШКОЛА ЭКОНОМИКИ**  
 НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ

**Приветствуем участников олимпиады для студентов и выпускников вузов!**



Фамилия:  Все поля обязательны для заполнения.  
 Имя:  Фамилию, имя и отчество необходимо писать русскими буквами.  
 Отчество:  Если у вас в паспорте нет отчества, поставьте -.  
 Дата рождения:  Дата рождения задается в формате ДД.ММ.ГГГГ.  
 Пол:    
 E-mail:   
 Контактный телефон:  Обязательно указывайте код населенного пункта!  
 Домашний телефон:

Я даю свое согласие на обработку персональных данных  
 Я подтверждаю достоверность предоставленных данных

[Перейти на сайт олимпиады](#)

Разработка ООО "Мельница Технологий"

Рисунок 1 – Анкета регистрации

После отправки заявки на электронный адрес приходит письмо с логином и паролем для входа в личный кабинет. На рисунке 2 изображен скриншот личного кабинета.

**Участник олимпиады студентов**

Фамилия:	<b>Иванов</b>	Логин:	<b>IVANOV1</b>
Имя:	<b>Иван</b>	Адреса электронной почты:	<b>ivan.ivanov@hse.ru; ivan.ivanov@hse.com</b>
Отчество:	<b>Иванович</b>	Контактные телефоны:	<b>+79802349876</b>
Дата рождения:	<b>10.05.1995</b>	Домашний телефон:	
Пол:	<b>Мужской</b>		

Рисунок 2 – Личный кабинет

## 1.2.2 Онлайн-олимпиада «Юный предприниматель»

Онлайн-олимпиада «Юный предприниматель» представляет собой одностраничный веб-сервер, дизайн которого оформлен с упором на детские тона. Целью проведения олимпиад на данном веб-сайте является развитие у детей с раннего возраста коммуникативных способностей. На рисунке 3 изображен скриншот общего дизайна сайта.

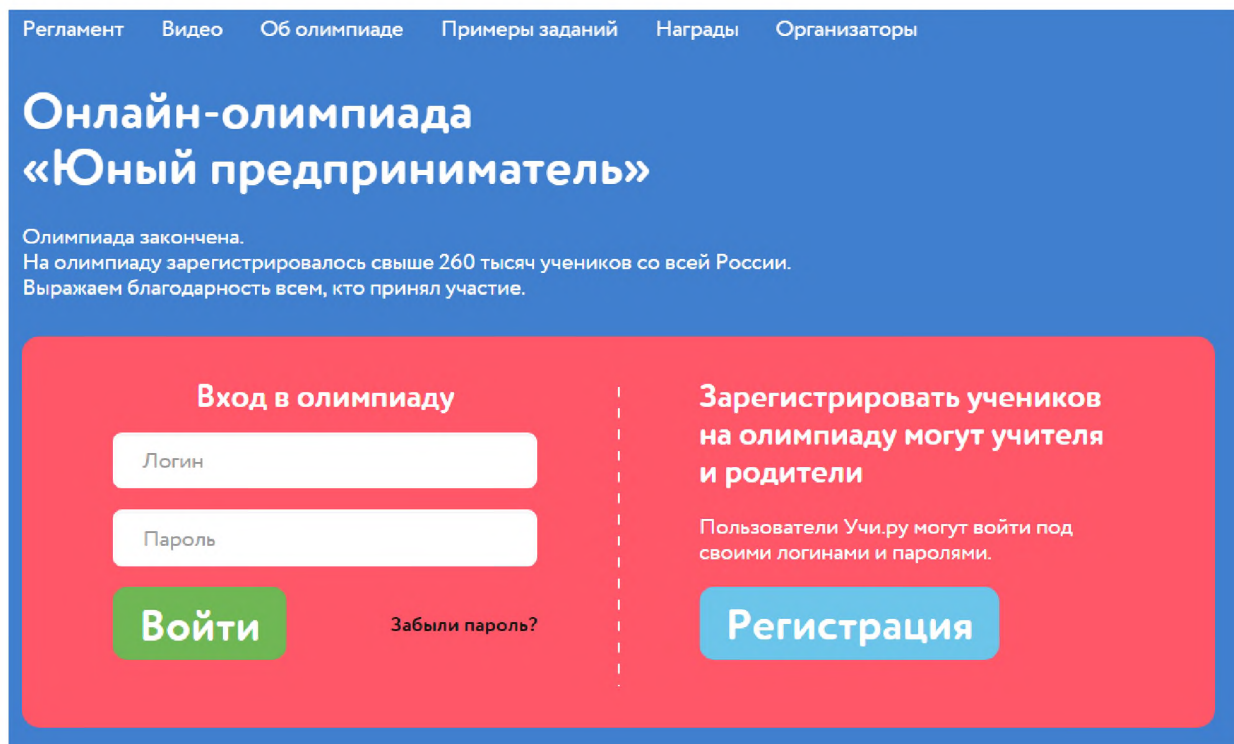


Рисунок 3 – Дизайн сайта Онлайн-олимпиады «Юный предприниматель»

Регистрация представляет собой поэтапный анимированный переход в ходе, которого участник заполняет все необходимые для регистрации поля.

## **1.3 Выбор средств разработки**

### **1.3.1 Анализ основных языков программирования**

На сегодняшний день существует множество языков программирования для разработки веб-серверов и не существует ни одного языка, который бы превосходил все остальные. Превосходство какого-либо языка программирования может проявляться только в контексте какой-либо задачи. Но многие задачи могут быть эффективно решены с помощью любого современного популярного языка программирования. Часто, выбор языка и определяется тем, какими знаниями обладают программисты, готовые реализовать проект. Также независимо от того, какой язык программирования будет выбран, в основе любого сайта лежит язык гипертекстовой разметки – HTML [2].

Однако отнести HTML к языкам программирования можно лишь с большой натяжкой, это все-таки язык разметки, но он является основой, без которой веб-сайт невозможно представить.

Для выбора подходящего языка разработки необходимо рассмотреть наиболее распространенные языки в веб-программировании, такие как:

- PHP+MySQL;
- Ruby;
- Python.

Данные языки выбраны для анализа в виду того, что они являются одними из самых популярных языков на сегодняшний день.

#### **1.3.1.1 Язык программирования PHP**

PHP (Hypertext Preprocessor") – это широко используемый язык программирования общего назначения с открытым исходным кодом. PHP разработан специально для ведения Web-разработок и может внедряться в

HTML-код. За последние годы PHP прошел долгий путь от набора Perl скриптов, позволяющих следить за посетителями страницы, и до, написанного уже на языке C, одного из самых известных и широко распространенных скриптовых языков для создания сложных, динамических, интерактивных web-сайтов. PHP лучше всего охарактеризовать как работающий на стороне сервера встроенный язык web сценариев, позволяющий разработчикам быстро и эффективно создавать динамические web-приложения. С позиций грамматики и синтаксиса PHP напоминает язык программирования C, хотя разработчики включили в него некоторые весьма полезные средства из других языков программирования, в том числе из Perl, Java и C++. Среди ценных заимствованных возможностей – поддержка регулярных выражений, мощные средства работы с массивами, объектно-ориентированная методология и обширная поддержка работы с различными базами данных [3].

История PHP начинается осенью 1994 года. Когда Расмус Лерддорф (Rasmus Lerdorf) начал работать над тем, что впоследствии стало PHP, единственной целью, которая была у него в мыслях, выяснить, кто читает его резюме. В то время, являясь независимым подрядчиком, Лерддорф рассылал потенциальным работодателям свое мини-резюме с URL ссылкой на его полную версию. Чтобы следить за посетителями, он создал CGI скрипт на Perl-е, который вставлялся как специальный тег в HTML код его страницы, и собирал информацию о посетителях. Чтобы произвести впечатление на потенциальных работодателей, он позволил любому посетителю страницы просматривать собираемую статистику посещений. Он назвал этот код для сбора статистики «PHP-Tools for Personal Home Page», поскольку сам использовал его на своей персональной домашней странице (personal home page). Несколько человек поинтересовались тем, как они могли бы получить этот инструмент, и Лерддорф принял решение предоставить его другим лицам. «Это чудо программного обеспечения. Вы можете дать это и тем не менее оставить это себе», остроумно заметил Лерддорф. В то время движения Open Source не существовало. «Тогда оно назвалось freeware». Ближе к концу 1995

года Лердорф открыл для людей первый список рассылки по PHP, чтобы можно было обмениваться идеями, исправлениями ошибок и кодом [3].

Основным преимуществом PHP является его практичность. PHP предоставляет разработчику возможности для эффективного и качественного решения поставленных задач. Для PHP можно выделить 4 основных характеристики:

- Традиционность;
- Эффективность;
- Простота;
- Гибкость.

**Традиционность.** В языке PHP используются конструкции, взятые из языков Си, Perl. Таким образом, код на PHP схож с Си, что облегчает разработчикам изучение языка, так как язык Си является основой в программировании.

**Эффективность.** «Движок» PHP – транслирующий интерпретатор. Он позволяет обрабатывать сценарии высокой скоростью.

**Простота.** PHP - язык, который может быть встроен непосредственно в html -код страниц, которые, в свою очередь будут корректно обрабатываться PHP -интерпретатором. Мы можем использовать PHP для написания CGI-сценариев и избавиться от множества неудобных операторов вывода текста. Мы можем привлекать PHP для формирования HTML-документов, избавившись от множества вызовов внешних сценариев.

Большое разнообразие функций PHP избавят вас от написания многострочных пользовательских функций на С или Pascal [4].

**Гибкость.** PHP – встраиваемый, гибкий по отношению к потребностям разработчиков, язык. Сценарии PHP перед отправкой клиенту, компилируются на стороне сервера.

Также, стоит отметить, что PHP является свободным программным обеспечением, что плодотворно сказалось на развитии языка.



К недостаткам языка стоит отнести:

- Слабая безопасность.
- Объекты передаются по значению, а не по ссылке.

### **1.3.1.2 Язык программирования Ruby**

Ruby – динамический императивный объектно-ориентированный язык программирования, разработанный Юкихио Матсумото [5].

Он характеризуется динамической типизацией и автоматическим управлением памятью [5].

Преимущества:

- Открытое программное обеспечение;
- Кроссплатформенность;
- Возможность внедрения в HTML;
- Легкая интеграция баз данных;
- Простой интерфейс для создания многопоточных приложений;
- Широкие возможности для внедрения массивов;
- Встроенный отладчик;
- Хорошая безопасность.

Недостатки:

- Освоение языка затруднительно на начальных этапах изучения;
- Недостаточное количество документации на русском языке;
- Слабая производительность;
- Мало хостов;
- Слабое развитие языка.

### 1.3.1.3 Язык программирования Python

Python – это объектно-ориентированный, интерпретируемый, переносимый язык сверхвысокого уровня.

Программирование на Python позволяет получать быстро и качественно необходимые программные модули. Интерпретатор Python может быть перенесён на любую платформу, будь то Unix, Windows, Linux, RiscOS, MAC, Sun. При написании кода на Питоне разработчики особо не задумываются о платформе. Таким образом, Python представляет серьёзную угрозу для Java, обеспечивая лёгкую переносимость, одновременно сочетая в себе средства, доступа к ресурсам операционной системы. В отличие от Java Питон не столь строг к использованию объектов, но реализуются они столь просто, что любой программист легко понимает сущность объектно-ориентированного подхода. Кроме этого, модули Python могут быть с лёгкостью использованы в программах, написанных на C++ и, что облегчает освоение Python. Python идеален в качестве cgi скриптов для веб-страниц, так как использует быстрый, эффективный и мощный интерпретатор. Питон может служить как в качестве языка для обучения, так и в качестве языка для разработки больших систем. Он прост, мощен и прост в освоении [6].

Преимущества:

- Открытое программное обеспечение;
- Просто в изучении на начальных этапах;
- Большое сообщество разработчиков;
- Элегантный синтаксис.

Недостатки:

- Плохая поддержка многопоточности;
- Плохая поддержка баз данных;
- Плохая скорость работы.

### 1.3.2 Язык программирования HTML

Передача информации в сети интернет происходит при помощи протоколов прикладного уровня, выполняющих различные функции (пересылка файлов, гипертекстовой информации, почты). Распределённая система, которая предоставляет доступ к различным документам и файлам, находящимся на разных компьютерах, подключённых к интернету, ее еще называют «Всемирной паутиной» основана на протоколе HTTP (Hyper Text Transfer Protocol - протокол передачи гипертекстовой информации). Для подготовки и работы с гипертекстовыми документами был разработан язык программирования HTML (Hyper Text Markup Language – язык разметки гипертекстовых документов), предоставляющий широкие возможности по форматированию и структурной разметке документов, организации связей между различными документами, средства включения графической и мультимедийной информации. HTML-документы просматриваются с помощью специальной программы – браузера [7].

HTML-документ состоит из текста, представляющего собой содержание документа, и тегов, определяющих его структуру и внешний вид при отображении браузером [7].

HTML - документ должен содержать следующие поля:

```
<html>  
<head>  
<title>Название</title>  
</head>  
<body>  
<p>Тело документа  
</body>  
</html>
```

Тэг является ключевым словом и заключается в угловые скобки. Бывают одинарные теги и парные, парные тэги имеют начальный открывающий тэг и

закрывающий тэг, а действие тэга заключается между этими двумя границами. Тэгам также можно присваивать параметры, например, цвет текста, цвет фона, размер шрифта и т.д.

Весь документ заключается в теги `<html>` и имеет две основных части - заголовок и тело. Заголовок описывается открывающим тегом `<head>` и закрывающим тэгом `</head>`, в заголовок обычно включают название документа и другие различные параметры. Тело описывается в теги `<body></body>` и содержит всю основную информацию. Для удобства отображения используют теги форматирования документа, вот основные тэги:

- `<p>` – начало нового абзаца;
- `<br>` – перевод строки в пределах текущего абзаца;
- `<u></u>` – выделение текста подчеркиванием;
- `<a href="URL">...</a>` – вставка ссылки, где URL – полный или относительный адрес документа;
- `` – отображение изображения.

### 1.3.3 СУБД MySQL

MySQL – это одна из самых популярных и самых распространенных СУБД в Интернете. Важным плюсом СУБД MySQL – возможность работы с API (Application Program Interface). API обеспечивает легкий доступ из программы пользователя к СУБД, также данная СУБД является свободным программным продуктом.

MySQL отличается хорошей скоростью работы, надежностью, гибкостью. Работа с ней, как правило, не вызывает больших трудностей. Поддержка сервера MySQL автоматически включается в поставку PHP [5].

### **1.3.4 Разметка стилей CSS.**

Аббревиатура CSS расшифровывается, как (Cascading Style Sheets – каскадные таблицы стилей). С помощью HTML создается структура документов. Со временем возникла необходимость использования в HTML все новых и новых атрибутов визуального оформления, которые сильно «захламляли» исходный код. В связи с этим был предложен другой, более перспективный вариант развития – создание отдельного языка стилевой разметки CSS [8].

### **1.3.5 Язык программирования JavaScript**

Javascript – это язык программирования, с помощью которого веб-страницам придается интерактивность. С его помощью создаются приложения, которые включаются в HTML-код (например, анкеты или формы регистрации, которые заполняются пользователем). Часто Javascript путают с языком программирования Java, однако общего между ними очень мало. К тому же, некоторые сравнивают Javascript с языками Python, Self, Ruby. Однако это особенный язык, который существует сам по себе [9].

### **1.3.6 Обоснование выбора средств разработки**

При выборе языка программирования стоит выбирать между PHP и Python, так как эти языки являются более традиционными и имеющими множество модификаций и документации, по сравнению с «молодым» Ruby.

PHP является наиболее популярным языком, так как за время его использования были созданы множество готовых программных решений и документации, что облегчает работу разработчикам, также PHP поддерживает множество баз данных, в особенности с СУБД MySQL, PHP+MySQL считается самой популярной «связкой» для управления сайтами. Также плюсом PHP

служит то, что данный язык поддерживается подавляющим большинством хостингов. Трудно делать выбор между PHP и Python, оба языка имеют сильные достоинства, но PHP более популярен в разработке веб-приложений, поэтому языком разработки был выбран PHP.

Выбор MySQL как основную СУБД обусловлен множеством ее плюсов:

- Свободное распространение;
- Поддержка большинства хостингов;
- Удобство настройки;
- Кроссплатформенность;
- «Дружеские взаимоотношения» с языком PHP.

Использование CSS позволяет значительно сократить программный код и сделать его более читабельным. Также CSS позволяет задавать параметры, недоступные HTML. И, например, CSS позволяет легко менять внешний вид, стиль веб-страниц.

Актуальность JavaScript в том, что он поддерживается и интегрируется практически во все браузеры, что делает выбор языка JavaScript универсальным.

## **1.4 Техническое задание**

### **1.4.1 Наименование работы**

Полное наименование работы:

Веб-сервер олимпиады по программированию ACM, проводимой в СФУ ИКИТ. Разработка пользовательского интерфейса.

Краткое наименование:

Веб-сервер олимпиады по программированию.

## **1.4.2 Функции и цели создания работы**

### **1.4.2.1 Функции работы**

Веб-сервер олимпиады должен выполнять следующие функции:

- Публикация различных материалов по проведению олимпиады (новостей, объявлений, документов);
- Регистрация участников олимпиады;
- Личный кабинет участников;
- Внесение данных об участниках олимпиады в базу данных веб-сайта;
- Внесение изменений в данных об участниках, занесенных в базу данных;
- Автоматическое формирование шаблонов документов для участников и организационного комитета;
- Реализация разделения олимпиад.

### **1.4.2.2 Цель создания работы**

Основной целью разработки и внедрения данной работы, является разработка веб-сервера для информирования, регистрации участников олимпиады по программированию АСМ, проводимой в СФУ ИКИТ, а также упрощения работы организаторов олимпиады.

### **1.4.3 Требования к дизайну веб-сервера**

При разработке дизайна веб-сервера должны быть использованы преимущественно светлые стили, также стили, присущие символике СФУ и ИКИТ в частности. Основные разделы сайта должны быть доступны с первой страницы. На первой (главной) странице не должно быть большого объема текстовой информации. Дизайн должен быть практичным и лаконичным.

#### **1.4.4 Требования к структуре веб-сервера**

Общая структура веб-сервера должна иметь вид:

- Главная;
- Новости;
- Расписание соревнований;
- Регистрация;
- Личный кабинет;
- Контакты.

#### **1.4.5 Требования к административной части веб-сервера**

Для получения доступа к административной части веб-сервера необходимо указать определенный адрес в строке браузера и пройти авторизацию.

Страница административной части должна содержать следующие пункты меню:

- Печать документов;
- Редактирование новостей;
- Списки зарегистрированных.

#### **1.4.6 Требования к разделению доступа**

Все разделы веб-сервера должны открываться для просмотра без аутентификации пользователя. При попытке входа в закрытый раздел у пользователя не прошедшего аутентификацию, должен быть запрошен логин и пароль. После прохождения этапа аутентификации система должна проверять полномочия пользователя на доступ к запрошенному разделу. Если доступ пользователю разрешен, то ему должен быть открыт доступ к запрашиваемому разделу.



## **1.4.7 Требования к информационному обеспечению**

### **1.4.7.1 Требования к хранению данных**

Данные веб-сервера, такие как анкеты участников, данные по администрированию, новости должны храниться под управлением реляционной СУБД. Исключения могут составлять файлы данных, предназначенные для просмотра и скачивания.

### **1.4.7.2 Требования к языкам программирования**

Для реализации статических страниц и стилей должны использоваться языки HTML и CSS.

Для реализации интерактивных элементов клиентской части должен использоваться язык JavaScript.

Для реализации динамических страниц должен использоваться язык PHP.

## **1.4.8 Требования к программному обеспечению**

### **1.4.8.1 Требования к программному обеспечению серверной части**

Для функционирования веб-сервера необходимо следующее программное обеспечение:

- Операционная система – Unix;
- Веб-сервер – Apache версии не ниже 1.3.26;
- СУБД – MySQL версии не ниже 3.23.

#### **1.4.8.2 Требования к клиентскому программному обеспечению**

Веб-сервер должен быть доступен для работы во всех основных браузерах.

#### **1.4.9 Требования к эргономике и технической эстетике**

Интерфейс должен быть понятен, выполнен в одном стиле и удобен пользователю. Не должно быть элементов, оказывающих влияние на психику человека.

На всех страницах должен отображаться логотип СФУ.

## **2 Разработка архитектуры и основных технических решений**

### **2.1 Функциональная структура веб-сервера**

Целью работы является автоматизация процесса регистрации и информирования потенциальных участников олимпиады по программированию АСМ, проводимой в СФУ ИКИТ. При ограниченности времени или невозможности личного приезда на место проведения регистрации, у потенциального участника будет возможность, не выходя из дома, узнать необходимую информацию и заполнить анкету с регистрацией.

При возникновении вопросов, будет реализована система «Обратной связи». Для оформления заявки необходимо заполнить форму, где нужно указать интересующий вопрос или пожелания и адрес электронной почты, на который ответит администратор.

Всю необходимую информацию об олимпиаде можно прочесть на главной странице, актуальные новости на странице «Новости», расписание новостей на странице «Расписание соревнований».

Регистрация осуществляется путем заполнения всех полей формы регистрации и подтверждения регистрации через email.

В административной части веб-сервера будет реализовано разделение прав. Администратору будет доступно редактирование и добавление новостей, информации на главной странице, просмотр и редактирование списка зарегистрированных участников и бейджиков. Члену оргкомитета будут доступны функции, аналогичные администратору, за исключением редактирования списка, зарегистрированных участников.

## 2.2 Информационная структура веб-сервера

Перед проектированием веб-сервера необходимо составить информационную и структурную модели. Реализация информационной модели может быть выполнена с помощью UML- языка.

### 2.2.1 Диаграмма прецедентов

Варианты использования предназначены в первую очередь для определения функциональных требований к системе и управляют всем процессом разработки.

В разрабатываемом веб-сервере выделены 3 роли пользователь, администратор и член Оргкомитета. Диаграмма прецедентов представлена на рисунке 4.



Рисунок 4 – Диаграмма прецедентов

Пользователь имеет функции:

- просмотр актуальных новостей;
- просмотр расписания соревнований;
- регистрация для участие в олимпиаде;

- просмотр корректности введенных данных.

Администратор имеет функции:

- создание и редактирование информации;
- просмотр списка зарегистрированных участников;
- создание и редактирование расписания соревнований;
- редактирование списка зарегистрированных участников;
- печать списка зарегистрированных участников.

Член Оргкомитета имеет функции:

- просмотр списка зарегистрированных участников;
- просмотр актуальных новостей;
- просмотр расписания соревнований;
- печать списка зарегистрированных участников.

### 2.3 Структура веб-сервера

Структура описывает функционал системы. На рисунке 5 отображена схема общей структуры сайта, которая учитывает всю специфику разрабатываемого приложения.

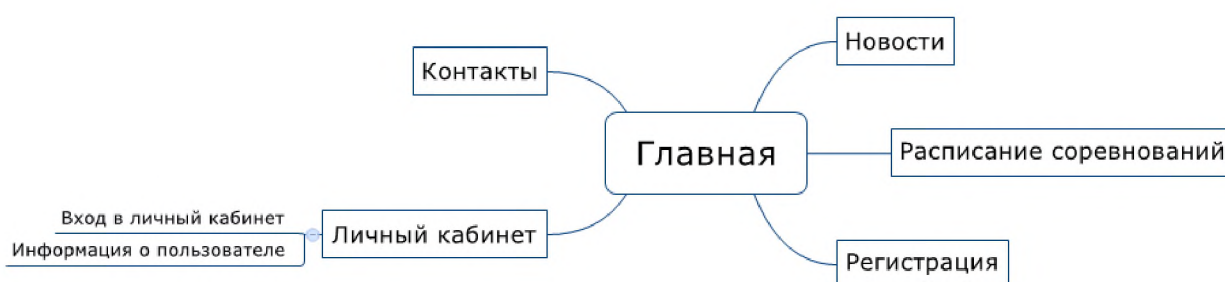


Рисунок 5 – Общая структура сайта

На главной странице отображается основная информация об олимпиаде.

Страница «Новости» содержит все актуальные новости, которые добавляет администратор.

Страница «Расписание соревнований» содержит таблицу с расписанием соревнований.

На странице «Регистрация» находится регистрационная анкета, которую должен заполнить участник для регистрации в олимпиаде, краткая инструкция к дальнейшим действиям.

Страница «Личный кабинет» содержит страницу с авторизацией пользователей, если участник зарегистрировался, заполнив анкету, то у него будут индивидуальные логин и пароль, после ввода которых, ему будет доступна страничка личного кабинета, где будет находиться образец бейджика с заполненной об участнике информацией.

Страница «Контакты» содержит контакты, по которым можно связаться с организаторами олимпиады.

На рисунке 6 отображена часть сайта, с которой будут работать администраторы и члены оргкомитета.

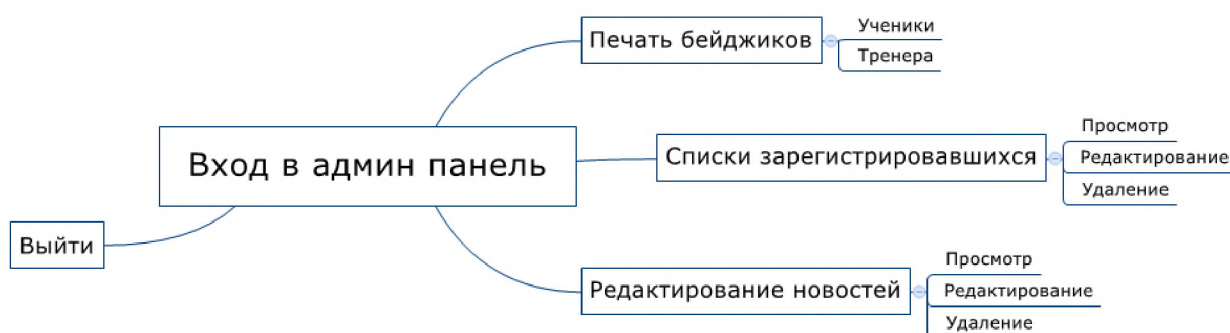


Рисунок 6 – Административная панель

При входе в административную панель необходимо ввести логин и пароль администратора или члена оргкомитета.

Страница «Печать бейджиков» позволяет загрузить на свое устройство списки бейджиков, зарегистрированных участников и тренеров, в формате doc.

Страница «Списки зарегистрировавшихся» содержит список зарегистрированных участников с заполненными анкетами при регистрации.

Члены оргкомитета могут просматривать анкеты, а администратор может редактировать и удалять пользователей из списка. Список формируется из базы данных.

Страница «Редактирование» содержит список с заголовками новостей, которые администратор может просматривать, редактировать, удалять и добавлять новые новости. Список новостей также же формируется из базы данных.

Страница «Выйти» позволяет выйти из авторизованного режима.

## **2.4 Описание создания страниц веб-сервера**

На первых этапах разработки был разработан макет веб-сервера. Затем были разработаны остальные страницы веб-сервера и подключена база данных в соответствие с техническим заданием. Следует рассмотреть наиболее важные страницы веб-сервера.

## **2.5 Разработка функции регистрации**

### **2.5.1 Структура функции регистрации**

Одной из важных функций, разрабатываемого веб-сервера, является функция регистрации. Для реализации этой функции была разработана структура регистрации (рисунок 7).

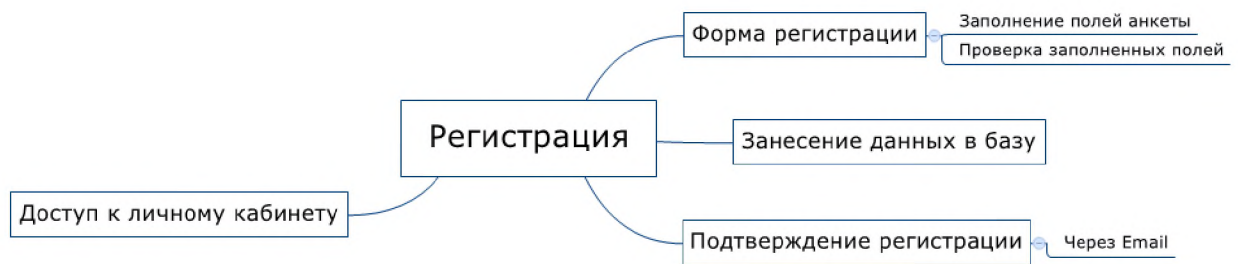


Рисунок 7 – Структура функции регистрации

На странице «Регистрация» будет размещена форма регистрации, а в конце формы будет написано о согласии пользователем на обработку персональных данных и кнопка о завершении регистрации. При нажатии на кнопку «Зарегистрироваться» будет проводиться проверка заполненных полей. Если все данные введены корректно, то участнику будет выведено сообщение о необходимости подтвердить регистрацию, перейдя на указанную участником электронную почту, на которую автоматически будет отправлено письмо о завершении регистрации в виде ссылки с пояснительным текстом. При переходе по ссылке участнику будет открыт доступ к собственному Личному кабинету.

### 2.5.2 Программная реализация функции регистрации

Проанализировав структуру функции регистрации, были разработаны два файла «reg\_uch.php» и «script\_post.php». Рассмотрим содержимое первого файла «reg\_uch.php».

Сперва, была подключена «шапка» веб-сервера.

```
<div id="body">
```

```
<? include_once("header.php"); ?>
```

Затем нижняя часть:

```
<? include_once("footer.php"); ?>
```



После был подключены стили дизайна веб-сервера:

```
<div class="clear"></div>
<div class="mw">
<div class="wh-block bordered" style="padding-left: 80px;">
<h2>ФОРМА РЕГИСТРАЦИИ УЧАСТНИКА</h2>
<div class="thumbs">
```

Следующим шагом собственно становится подключение базы данных: Соединение с базой данных осуществлялось через функцию «*mysql\_connect*» Аргументами функции являются имя сервера, логин и пароль:

```
<?php
if (!mysql_connect(localhost, c933322l_bd, stalker)) {
echo "Ошибка подключения к серверу MySQL";
exit;
}
//функция mysql_select_db() выбирает текущую базу данных
mysql_select_db($db);
?>
```

После была разработана форма регистрации.

```
<form name="form1" action="script_post.php" method="post" id="form_reg">
```

После отправки формы будет вызван файл «script\_post.php»

Далее были описаны все поля для регистрации, ниже приведен фрагмент программного кода с описанием разработки полей.

```
<div class="form_reg"> <p style="font-size: 15px; padding: 8px; color: #0099CC;
">Фамилия</p>
<input type="text" name="fam" size="20" maxlength="50" value="">
</div>
<div class="form_reg"> <p style="font-size: 15px; padding: 8px; color: #0099CC;">Имя</p>
<input type="text" name="name" size="20" maxlength="50" value="">
</div>
<div class="form_reg"> <p style="font-size: 15px; padding: 8px; color:
#0099CC;">Отчество</p>
<input type="text" name="otch" size="20" maxlength="50" value="">
```

</div>

<div class="form\_reg"> <p style="font-size: 15px; padding: 8px;color: #0099CC;">Фамилия  
(на английском)</p>

<input type="text" name="fam\_en" size="20" maxlength="50" value="">

</div>

<div class="form\_reg"> <p style="font-size: 15px; padding: 8px;color: #0099CC;">Имя (на  
английском)</p>

<input type="text" name="name\_en" size="20" maxlength="50" value="">

</div>

Заключительным шагом в разработке документа является завершение регистрации.

<div style="padding-left: 166px; padding-bottom: 20px; padding-top: 25px;color: #FF6500;">  
<p>\*Нажимая на кнопку регистрации вы соглашаетесь на обработку персональных  
данных</p>

</div>

Ниже описана разработка кнопки «Зарегистрироваться».

<div style="padding-left: 300px; padding-bottom: 20px; padding-top: 25px;">

<input type="submit" name="submit" value="Зарегистрироваться" onclick="return  
Formdata(this.form)" style="border: 0px; cursor: pointer;color: white;background:  
#FF6500;font-size: 18px; border-radius: 4px;">

</div>

Во втором файле «script\_post.php» реализована проверка и внесение данных о новом пользователе в базу данных.

Первым шагом было подключение базы данных, как в предыдущем документе. Затем был получен массив данных из анкеты, заполненной участником. Ниже приведен фрагмент программного кода, получения массива данных.

```
if(isset($_GET["hesh"]))
```

```
else
```

```

$fam = $_POST["fam"];
$name = $_POST["name"];
$otch = $_POST["otch"];
$fam_en = $_POST["fam_en"];
$name_en = $_POST["name_en"];
$otch_en = $_POST["otch_en"];
$date = $_POST["date"];
$uch_zav = $_POST["uch_zav"];
$email = $_POST["email"];
$password = $_POST["password"];
$number = $_POST["number"];
$tren_fam = $_POST["tren_fam"];
$tren_nam = $_POST["tren_nam"];
$tren_otch = $_POST["tren_otch"];
$tren2_fam = $_POST["tren2_fam"];
$tren2_nam = $_POST["tren2_nam"];
$tren2_otch = $_POST["tren2_otch"];
$teacher = $_POST["teacher"];
$browser = $_POST["browser"];

```

Для успешного завершения регистрации была реализована система подтверждения регистрации через email.

```

$uid = md5(uniqid(rand(), true));
$per = "script_post?hesh=";
$per .= $uid;
$to = (string) $email;
$subject = "Подтверждение email пользователя";
$message = "
<html>
<body>
<p>Для продолжения регистрации участника олимпиады по программированию
пройдите по ссылке, <br>если вы не регистрировались на сайте олимпиады, то
проигнорируйте это сообщение</p>
$per
</body>

```

```

</html>";
$headers = "Content-type: text/html; charset=utf-8 \r\n";
$headers .= "From: От кого письмо <from@example.com>\r\n";
$headers .= "Reply-To: reply-to@example.com\r\n";
mail($to, $subject, $message, $headers);
}

```

Далее реализована проверка при переходе по ссылке из сообщения на электронной почте. Т.е. происходит проверка переменной из строки браузера и переменной в базе данных, если переменные совпадают, то в базе ставится 1, это означает, что регистрация подтверждена.

```

if(isset($_GET["hesh"]))
{
    $per = $_GET["hesh"];
    $prov1 = 0;
    $strSQL = "SELECT hesh FROM uch WHERE hesh = '$per'";
    // Выполнить запрос (набор данных $rs содержит результат)
    $rs = mysql_query($strSQL);
    // Цикл по recordset $rs
    // Каждый ряд становится массивом ($row) с помощью функции
    «mysql_fetch_array»
    while($row = mysql_fetch_array($rs)) {
        // Записать значение столбца FirstName (который является теперь массивом
        $row)
        $prov1 = 1;
    }
    if ($prov1 == 1)
        $result = mysql_query ("UPDATE uch SET pod='1' WHERE hesh = '$per'");
    if ($result == 'true')
        echo "<p style = 'font-size: 20px;'>Вы успешно зарегистрировались<br>";
        echo "<a href = 'vhod_uch'>Перейти в личный кабинет</a></p>";
}

```

Затем происходит занесение данных в базу данных и ожидание подтверждения, а также вывод информации об успешной регистрации.

Ниже описан фрагмент программного кода о занесении в базу данных и выводе информации об успешной регистрации.

```

$prov = 0;
$strSQL = "SELECT email FROM uch WHERE email = '$email'";
$rs = mysql_query($strSQL);
while($row = mysql_fetch_array($rs)) {
$prov = 1;
if ($prov == 0) {
if (isset($teacher))
if (isset($_POST["name"]) and isset($_POST["fam"]) and isset($_POST["otch"]) and
isset($_POST["name_en"]) and isset($_POST["fam_en"]) and isset($_POST["otch_en"]) and
isset($_POST["date"]) and isset($_POST["uch_zav"]) and isset($_POST["email"]) and
isset($_POST["password"]) and isset($_POST["number"]) and isset($_POST["teacher"])) {
//Вставляем данные, подставляя их в запрос
$result = mysql_query ("INSERT INTO uch
(name,fam,otch,name_en,fam_en,otch_en,date,uch_zav,email,password,number,tren,tren2,paren
t,tip,hesh)VALUES
('$name','$fam','$otch','$name_en','$fam_en','$otch_en','$date','$uch_zav','$email','$password
','$number','$tren','$tren2','$teacher','$browser','$suid')");
if ($result = 'true'){
echo "Вы успешно зарегистрировались";
echo " <a href = 'eg_uch'>Зарегистрировать нового ученика</a>";
echo " <a href = 'vhod'>Перейти в личный кабинет</a>";
}
}
}
}

```

### 2.5.3 Результат работы функции регистрации

Результатом работы будет являться зарегистрированный участник, данные о котором будут храниться в базе данных, а участнику будет доступен Личный кабинет. Далее рассмотрен визуальный процесс регистрации. На рисунке 8 изображен процесс заполнения анкеты.

### ФОРМА РЕГИСТРАЦИИ УЧАСТНИКА

Фамилия	Имя	Отчество
Иванов	Иван	Иванович
Фамилия (на английском)	Имя (на английском)	Отчество (на английском)
Ivanov	Ivan	Ivanovich
Дата рождения	Название учебного заведения	
20.12.1994	МБОУ Новоленинская СОШ	
Email	Пароль	Номер телефона
lastmoran123@gmail.com	.....	89333222445
Фамилия Тренера	Имя Тренера	Отчество Тренера
Макуха	Любовь	Витальевна
Фамилия второго Тренера	Имя второго Тренера	Отчество второго Тренера
-	-	-
Кем вы являетесь?		
<input type="radio"/> Студент		
<input checked="" type="radio"/> Студент ИКИТ		
<input type="radio"/> Школьник		
<small>*Нажимая на кнопку регистрации вы соглашаетесь на обработку персональных данных</small>		
<input type="button" value="Зарегистрироваться"/>		

Рисунок 8 – Заполнение формы регистрации

При нажатии на кнопку регистрации появляется окно с просьбой подтвердить регистрацию (рисунок 9).

### ПОДТВЕРЖДЕНИЕ РЕГИСТРАЦИИ УЧАСТНИКА

Вам отправлено письмо на почту, для подтверждения регистрации перейдите по ссылке в сообщении.

Рисунок 9 – Подтверждение регистрации

На электронный ящик пришло сообщение о завершении регистрации, перейдя по ссылке, участник попадает на следующую страницу (рисунок 10).

## ПОДТВЕРЖДЕНИЕ РЕГИСТРАЦИИ УЧАСТНИКА

Вы успешно зарегистрировались

[Перейти в личный кабинет](#)

Рисунок 10 – Завершение регистрации

## 2.6 Разработка личного кабинета

### 2.6.1 Структура личного кабинета

После завершения регистрации участнику рекомендуется перейти в личный кабинет для просмотра и проверки корректности информации, заполненной им при регистрации. Для разработки личного кабинета была составлена структура (рисунок 11).

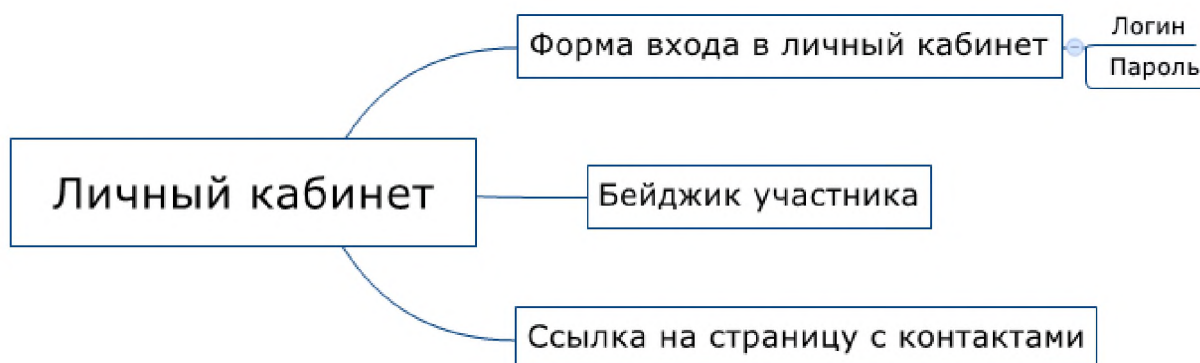


Рисунок 11 – Структура личного кабинета

Для входа в личный кабинет будет разработана форма входа с вводом логина и пароля. В сам личном кабинете будет отображен образец бейджика с заполненными данными об этом участнике, которые будут браться из базы данных. Если участник найдет ошибки в заполненном бейджике, то в личном кабинете будет опубликована информация по действиям, которые он должен совершить для исправления ошибок.

## 2.6.2 Программная реализация Личного кабинета

В соответствии с разработанной структурой был разработан файл «vhod\_uch.php». Следует рассмотреть некоторые фрагменты данного файла.

Ниже приведено программное описание формы ввода логина и пароля в личном кабинете.

```
php?>
<form name="forma1" action="vhod_uch.php" method="post" id = "form_reg">
<div style="padding-left: 249px;padding-bottom: 20px; padding-top: 25px;">
<div class="form_reg"> <p style="font-size: 15px; padding: 8px; color: #0099CC ;">
Email</p>
//Ввод логина
<input type="text" name="email" size="20" maxlength="50" value="">
</div>
<br>
<div class="form_reg"> <p style="font-size: 15px;padding: 8px;color: #0099CC;">Пароль</p>
//Ввод пароля
<input type="password" name="password" size="20" maxlength="50" value="">
</div>
</div>
<div style="padding-left: 320px;padding-bottom: 20px; padding-top: 25px;">
//Кнопка войти
<input type="submit" name="submit" value="Войти" onclick="return Formdata (this.form)"
style="border: 0px; cursor: pointer; color: white; background: #FF6500;font-size: 18px;
border-radius: 4px;">
</div>
<div style="padding-left: 250px;padding-bottom: 20px; padding-top: 25px;">
<p ><center style="margin-right: 357px;font-size: 15px;color: #0099CC;">Если вы еще не
зарегистрировались<br> то перейдите на страницу регистрации</center></p><br>
//Ссылка на вкладку с регистрацией
<a class="en" href="reg_uch">Зарегистрироваться</a>
```



После того как пользователь вводит логин и пароль должна происходить проверка существования данного пользователя в базе данных.

Ниже представлен код проверки.

```
if (isset($_POST["email"]) and isset($_POST["password"]))
{
    $email = $_POST["email"];
    $pass = $_POST["password"];
    $a = 0;
    $result1 = mysql_query("SELECT email, password, name, fam, pod FROM uch");
    while ($row = mysql_fetch_array($result1, MYSQL_ASSOC)) {
        if($email == $row['email'] and $pass == $row['password'] and $row['pod'] == '1'){
            $a = 1;
            $name = $row["name"];
            $fam = $row["fam"];
            $email = $row["email"];
        }
    }
    mysql_free_result($result1);
    if($a == 1)
    {
        $_SESSION['name'] = $name;
        $_SESSION['fam'] = $fam;
        $_SESSION['email'] = $email;
        echo "Здравствуйте, " . $_SESSION['name']. " " . $_SESSION['fam']. "!";
    }
}
```

Если логин и пароль не совпадают с данными базы данных, то пользователю об этом будет сказано.

?>

*<p style="font-size: 14px;color: red;margin-top: 12px;margin-left: 200px;">Неверный логин или пароль, попробуйте еще раз</p>*

После успешного входа участнику становится доступным Личный кабинет. В связи с требованиями структуры личного кабинета, участнику будет выводиться образец бейджика и некоторая информация с ссылкой на контакты.

Ниже представлена программная реализация вывода бейджика.

```
if (isset($_SESSION['name']))
{
    $sem = $_SESSION['email'];
    $result = mysql_query("SELECT * FROM uch WHERE email = '$sem'");
    while ($row = mysql_fetch_array($result, MYSQL_ASSOC)) {
        <table>
        <tr style="border: 1px black solid;">
        <td></td>
        <td style="vertical-align: middle;"><center>Международная олимпиада ACM <br>no
        программированию <br>четвертьфинал</center></td>
        <td></td>
        </tr>
        <tr style = " border-left: 1px black solid;border-right: 1px black solid;">
        <td colspan="3" style="vertical-align: middle;text-align: center;font-size: 18px;font-weight:
        500;"><? echo $row['uch_zav']; ?></td>
        <tr style = " border-left: 1px black solid;border-right: 1px black solid;">
        <td colspan="3" style="vertical-align: middle;text-align: center;font-size: 25px;font-weight:
        500;"><? echo $row['fam']; ?> <br> <? echo $row['name']; ?>&nbsp; <? echo $row['otch'];
        ?></td>
        <tr style = " border-left: 1px black solid;border-right: 1px black solid;">
        <td style="font-size: 24px;">Образец</td>
        <td style="font-size: 18px;font-weight: 500;padding-top: 12px;text-align:
        right;">Участник</td>
        <tr style="border: 1px black solid;">
        <td colspan="2"></td>
        <td style="font-size: 29px;font-weight: 500;vertical-align: middle;color: rgba(156, 39, 176,
        0.94);">2016</td>
        </table>
        <div style="text-align:center;padding-left: 106px;padding-bottom: 20px;padding-top: 25px;font-
        size:18px;">
        <p> Если у вас остались вопросы или вы хотите изменить данные, то воспользуйтесь
        контактной формой
        <br><a href = "kontakty"> Контакты</a></p>
        </div>
```

### 2.6.3 Результат работы функции личного кабинета

На рисунке 12 изображена форма входа в личный кабинет.

ФОРМА ВХОДА В ЛИЧНЫЙ КАБИНЕТ УЧАСТНИКА

Email

Пароль




Войти

Если вы еще не зарегистрировались то перейдите на страницу регистрации

Зарегистрироваться

Рисунок 12 – Форма входа в личный кабинет

После корректного ввода логина и пароля участнику будет доступен личный кабинет с образцом бейджика и ссылкой на контакты (рисунок 13).

	Международная олимпиада ACM по программированию четвертьфинал	
МБОУ Новоленинская СОШ		
Иванов		
Иван Иванович		
Образец		Участник
	International Collegiate Programming Contest	2016

Если у вас остались вопросы или вы хотите изменить данные, то воспользуйтесь контактной формой [Контакты](#)

Рисунок 13 – Личный кабинет

## 2.7 Перенос веб-сервера на CMS

В целях улучшения функциональности и упрощения работы с веб-сервером было принято решение о переносе разработанного веб-сервера на CMS. Средой для переноса была выбрана Wordpress.

### 2.7.1 Описание установки CMS Wordpress

Для установки Wordpress (далее wp) необходимо в панели управления хостингом создать базу данных. Затем, среди установочных файлов найти файл «*wp-config-sample.php*», переименовать его в «*wp-config.php*» и изменить некоторые поля данного файла. Поля для изменения представлены на рисунке 14.

```
// ** Параметры MySQL: Эту информацию можно получить у вашего хостинг-провайдера ** //  
/** Имя базы данных для WordPress */  
define('DB_NAME', 'database_name_here');  
  
/** Имя пользователя MySQL */  
define('DB_USER', 'username_here');  
  
/** Пароль к базе данных MySQL */  
define('DB_PASSWORD', 'password_here');  
  
/** Имя сервера MySQL */  
define('DB_HOST', 'localhost');  
  
/** Кодировка базы данных для создания таблиц. */  
define('DB_CHARSET', 'utf8');  
  
/** Схема сопоставления. Не меняйте, если не уверены. */  
define('DB_COLLATE', '');
```

Рисунок 14 – Конфигурация файла *wp-config.php*

После редактирования полей необходимо перенести все установочные файлы wp на хостинг. Затем в браузере необходимо ввести адрес – «<http://wptest.ru/wp-admin/install.php>».

В открывшемся окне необходимо заполнить поле с названием веб-сайта и адресом электронной почты, после чего нажать кнопку «Установить WordPress».

Затем необходимо было войти в панель управления wp (рисунок 15).

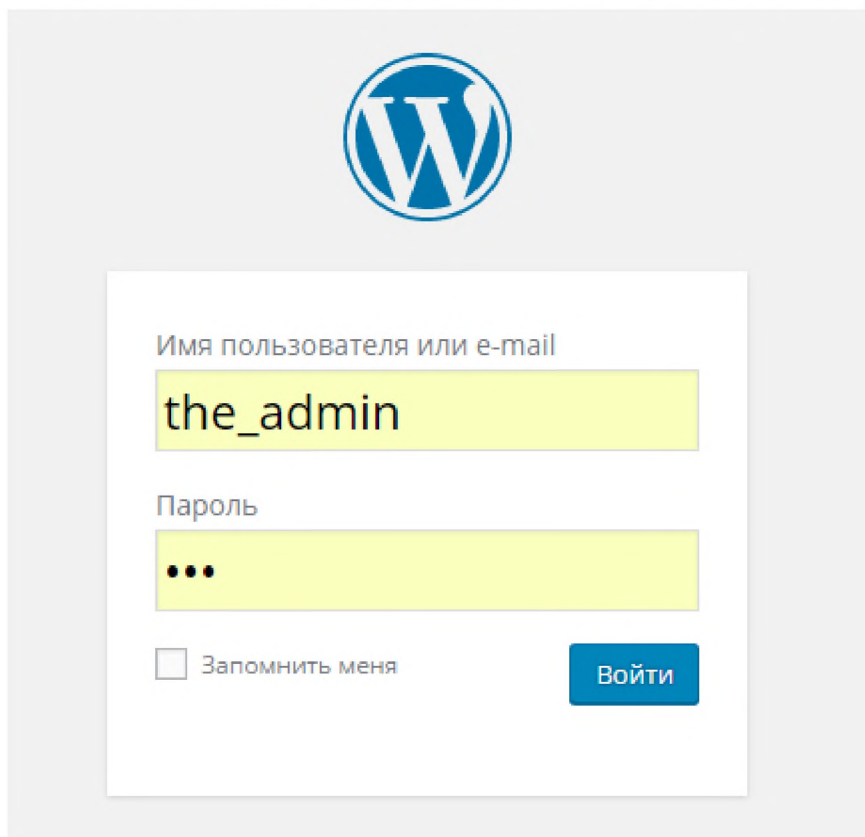


Рисунок 15 – Вход в панель управления wp

### **2.7.2 Настройка темы веб-сервера**

Темой оформления была выбрана стандартная тема wp под названием «courage» (рисунок 16).

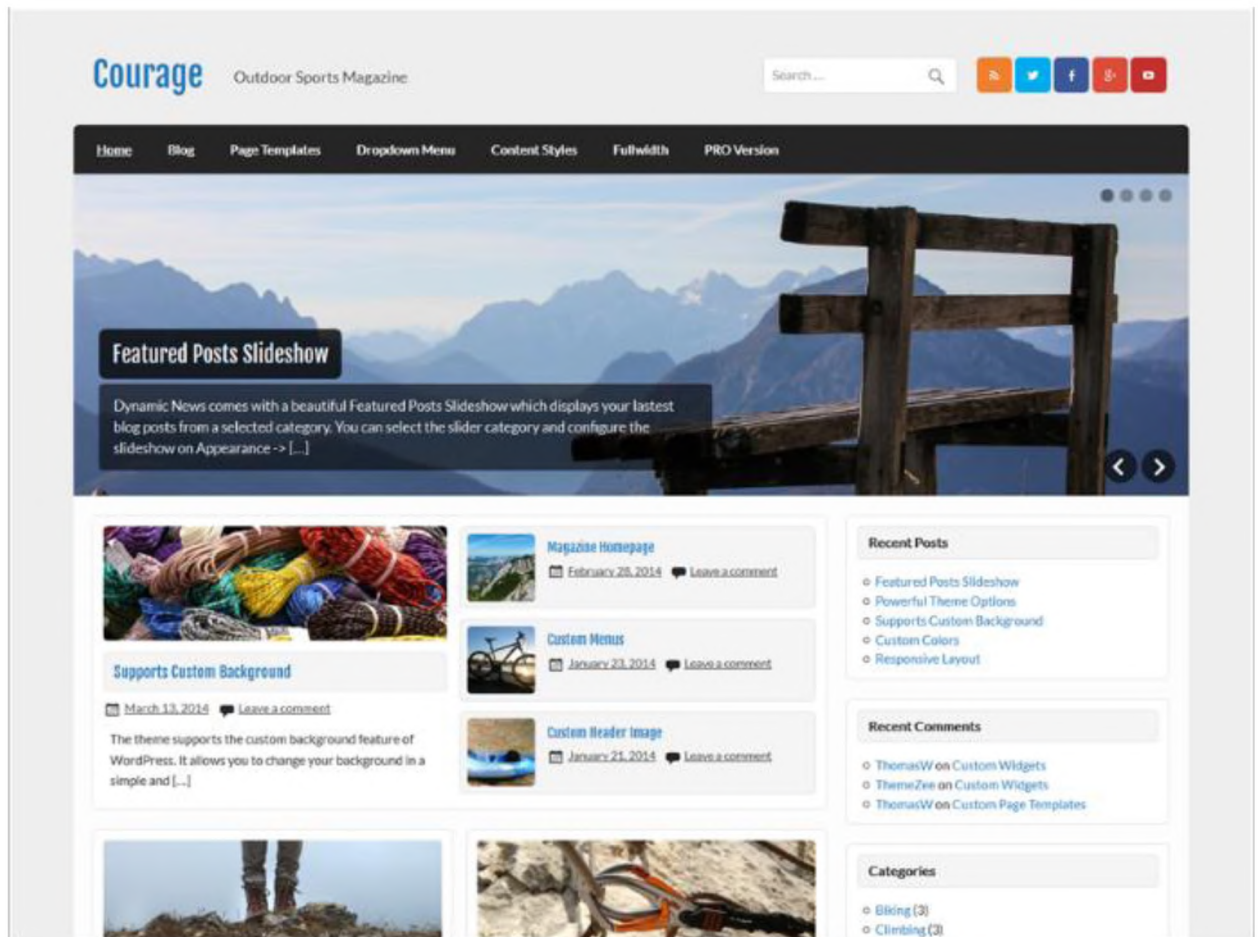


Рисунок 16 – Стандартная тема wp «courage»

### 2.7.3 Создание страниц веб-сервера

В соответствии с общей структурой веб-сервера были созданы соответствующие страницы (рисунок 17). Наполнение страниц было осуществлено с помощью плагина Visual Composer.



Страницы [Добавить новую](#) Настройки экрана ▾

Все (8) | Опубликованные (6) | Личные (2) | Корзина (7)

Действия ▾ Применить Все даты ▾ Фильтр

<input type="checkbox"/> Заголовок	Автор		Дата
<input type="checkbox"/> Главная — Главная страница	the_admin	—	Опубликовано 07.06.2016
<input type="checkbox"/> Контакты	the_admin	—	Опубликовано 23.05.2016
<input type="checkbox"/> Личный кабинет	the_admin	—	Опубликовано 23.05.2016
<input type="checkbox"/> Новости — Страница записей <a href="#">Изменить</a>   <a href="#">Свойства</a>   <a href="#">Удалить</a>   <a href="#">Перейти</a>   <a href="#">Edit with Visual Composer</a>	the_admin	—	Опубликовано 23.05.2016
<input type="checkbox"/> Печать карточки участника — Личное	the_admin	—	Последнее изменение 23.05.2016
<input type="checkbox"/> Расписание соревнований	the_admin	—	Опубликовано 23.05.2016
<input type="checkbox"/> Регистрация	the_admin	—	Опубликовано 23.05.2016
<input type="checkbox"/> Список участников — Личное	the_admin	—	Последнее изменение 23.05.2016

Рисунок 17 – Страницы веб-сервера

## 2.7.4 Создание новостей

Новости создаются и редактируются через раздел «записи» в панели управления wp (рисунок 18).

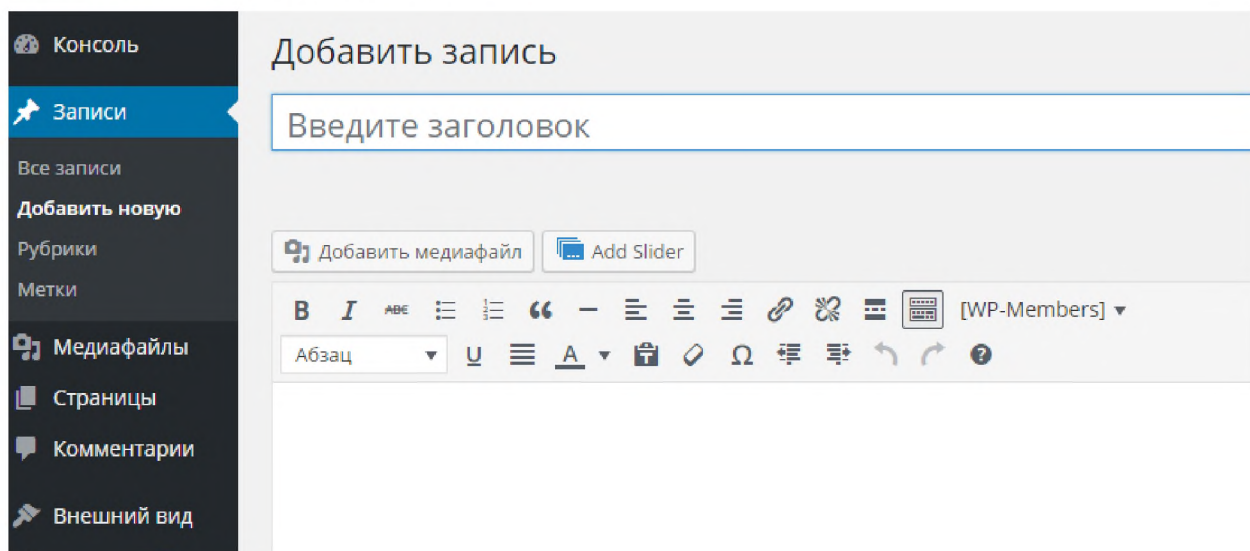


Рисунок 18 – Создание новостей

## 2.7.5 Подключение плагинов

Для удобства работы и выполнения всего необходимого функционала были подключены следующие плагины:

- WP-Members – плагин для создания, ограничения доступа и регистрации пользователей в wp;
- Email as Username for WP-Members – плагин для использования Email в качестве имени пользователя;
- Simple User Listing – плагин для создания шорткодов;
- Visual Composer – конструктор страниц для wp.

## 2.7.6 Регистрация участников

Функция регистрации участников реализована с помощью плагина WP-members. Данный плагин программно реализует весь процесс регистрации, занесения данных в базу данных и подтверждения регистрации через Email. Для корректного функционирования плагина были настроены поля для регистрации. Фрагмент полей регистрации представлен на рисунке 19.

Добавить/удалить	Метка поля	Имя поля	Тип поля
	Email*	user_email	email
<input type="checkbox"/> Удалить	Номер телефона*	phone	text
	Фамилия*	last_name	text
	Имя*	first_name	text
<input type="checkbox"/> Удалить	Отчество*	last_name2	text
<input type="checkbox"/> Удалить	Название учебного заведения*	vuz	text
<input type="checkbox"/> Удалить	Кем вы являетесь?*	st_status	select

Рисунок 19 – Фрагмент полей регистрации



## 2.8 Запуск проекта

Веб-сервер разрабатывался на тестовом хостинге «<https://cp.beget.com/>», панель управления представлена на рисунке 20, на данном хостинге были проведены тесты, и проверка работоспособности веб-сервера. После демонстрации, разработанного веб-сервера дипломному руководителю и исправления, выявленных им недостатков, все данные веб-сервера были перенесены на хостинг СФУ.

**Общая информация:**

Имя пользователя, ID	c9333221 (326660)
Движения по счету	<a href="#">Посмотреть</a>
Тарифный план	<a href="#">Blog</a>
Разрешенная нагрузка	65 CP <a href="#">Посмотреть</a>
Разрешенная нагрузка MySQL	2500 CP
Текущий баланс	24.64 руб. <a href="#">Пополнить</a>
Суммарная стоимость услуг	4.44 руб/день
Дата блокировки	01.06.2016 (6 дней)
Сервер	oscar3.beget.com
IP адрес	87.236.19.32
Сайты	1 / 2
Домены	1 / неогр.
Базы данных MySQL	2 / неогр.
Размер баз данных	1 Мб.
Дисковое пространство	25 Мб / 1000 Мб (3%)
Почтовые ящики	0 / неогр.
Размер ящиков	0 Мб / неогр.
FTP-аккаунты	1 / неогр.

**На Вашем счету осталось мало средств**  
Если у Вас нет возможности пополнить счет в данный момент, то Вы можете воспользоваться "[Обещанным платежом](#)".

- FTP
- CronTab
- Помощь и поддержка
- MySQL
- Поддомены
- Сайты
- Файловый менеджер
- DNS
- Домены
- CMS
- BackUp
- Почта

Рисунок 20 – Панель управления

## **3 Руководство пользователя**

### **3.1 Требования к программному и серверному обеспечению**

Для корректного функционирования веб-сервера программное и серверное обеспечение должны соответствовать минимальным требованиям.

Программное обеспечение:

- CentOS Linux release 7.1.1503.

Серверное обеспечение:

- Apache версии не ниже 1.3.26.

### **3.2 Требования к клиентскому программному обеспечению**

Для корректного отображения веб-сервера, версии браузеров должны быть не ниже следующих:

- Mozilla Firefox 38.0.1;
- Google Chrome 42.0;
- Opera 29;
- Internet Explorer 11;
- Safari 5.1.7.

### **3.3 Перенос веб-сервера на хостинг СФУ**

Перенос веб-сервера на хостинг СФУ можно разделить на два шага.

Первым шагом является перенос базы данных. Для этого необходимо экспортировать существующую базу данных (рисунок 21).

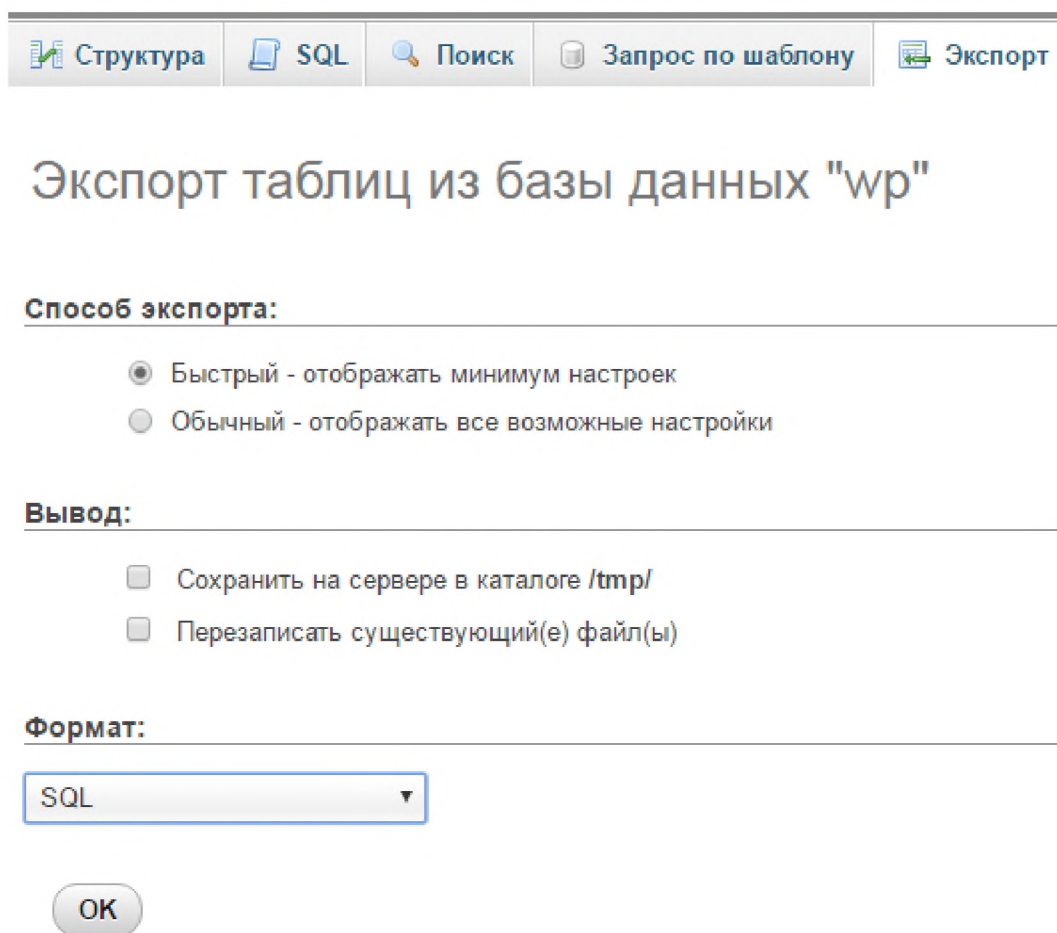


Рисунок 21 – Экспорт базы данных

После этого в файле с базой данных «*wp.sql*» необходимо найти и заменить текущий адрес сайта на адрес, выданный Сибирским федеральным университетом, затем импортировать в СУБД.

Вторым шагом является перенос всех программных файлов веб-сервера. Для реализации данного шага необходимо с помощью FTP- клиента скачать все программные файлы веб-сервера и загрузить их на хостинг СФУ.

## ЗАКЛЮЧЕНИЕ

В ходе выполнения выпускной квалификационной работы был разработан полнофункциональный web-сервер, полностью готовый к применению. Данный веб-сервер ориентирован на студентов и абитуриентов, интересующихся участием в олимпиаде по программированию АСМ, а также на организаторов. С его помощью пользователи смогут получать необходимую информацию по олимпиаде и подавать заявку на участие. А организаторы смогут обрабатывать заявки пользователей, редактировать их, выкладывать всю необходимую и актуальную информацию об олимпиаде в соответствующих разделах веб-сервера, также организаторам будут доступны списки и бейджики всех зарегистрированных участников в удобной для ознакомления форме.

При разработке веб-сервера был проведен анализ современных web-технологий, которые на сегодняшний день позволяют разрабатывать интерактивные веб-страницы.

Разработанный веб-сервер соответствует всем требованиям, приведенным в техническом задании.

В качестве усовершенствования веб-сервера был осуществлен перенос на систему управления содержимым.

Возможна некоторая доработка интерфейса веб-сервера для повышения его информативности, привлекательности и удобства, а также наполнение страниц веб-сервера различным контентом.

## СПИСОК СОКРАЩЕНИЙ

СУБД – система управления базами данных

PHP – Hypertext Preprocessor

НИУ – Национальный Исследовательский Университет

CSS – Каскадная таблица стилей

CGI – Common Gateway Interface

API – Application Program Interface

HTML – HyperText Markup Language

АСМ – Международная студенческая олимпиада

ВКР – Выпускная квалификационная работа

CMS – Система управления содержимым

ФИО – Фамилия, имя, отчество

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Высшая школа экономики [Электронный ресурс]: – Режим доступа: <https://olymp3.hse.ru/stud.html>
2. Developerguru [Электронный ресурс]: Языки веб программирования - какой выбрать? - Режим доступа: <http://developer.guru.net/post/web-programming-languages/>
3. Виталий Кушпель [Электронный ресурс]: Обзор современных web-технологий - Режим доступа: <http://vitalij-kushpel.narod.ru/glava1/3.html>
4. Современные информационные технологии [Электронный ресурс]: Преимущества и недостатки PHP+MySQL - Режим доступа: [http://www.rusnauka.com/30\\_NIEK\\_2011/Informatica/2\\_96228.doc.htm](http://www.rusnauka.com/30_NIEK_2011/Informatica/2_96228.doc.htm)
5. Сайтостроение от А до Я [Электронный ресурс]: Преимущества и недостатки PHP+MySQL - Режим доступа: [http://www.internet-technologies.ru/articles/article\\_1991.html](http://www.internet-technologies.ru/articles/article_1991.html)
6. Открытая Сеть [Электронный ресурс]: Введение в язык Python – Режим доступа: [https://www.opennet.ru/docs/RUS/python/python\\_b.html](https://www.opennet.ru/docs/RUS/python/python_b.html)
7. Лекциопедия [Электронный ресурс]: Виды Web-технологий - Режим доступа: <http://lektsiopedia.org/lek3-779.html>
8. Кто на новенького? [Электронный ресурс]: CSS - что это такое? - Режим доступа: <http://ktonanovenkogo.ru/html/uroki-css/css-tablicy-kaskadnyx-stilej-cto-eto-takoe-style-link-yazyk-css-html.html>
9. Все, что вы хотели знать о сайтостроении и хостинге [Электронный ресурс]: Javascript – что это такое? - Режим доступа: <http://articles-hosting.ru/329/javascript-%E2%80%93-cto-eto-takoe.html>
10. СТО 4.2–07–2014 Система менеджмента качества. Общие требования к построению, изложению и оформлению документов учебной деятельности. – Введ. 22.12.2014. – Красноярск: ИПК СФУ, 2014. – 41 с.