**РУКОВОДСТВО ПРОГРАММИСТА**

**Форум**

**«Crush test yourself»**

**Листов 13**

2022

**АННОТАЦИЯ**

В данном программном документе приведено руководство программиста по работе с форумом «Crush test yourself».

В разделе «Назначение и условия применения программы» указаны назначение и функции, выполняемые сайтом, условия, необходимые для его работы.

В разделе «Характеристика сайта» приведено описание основных

характеристик и особенностей.

1. Назначение и условия применения программы

Целью продукта являются популяризация спорта и получение финансовой прибыли. Популяризация спорта заключается в основной тематике форума, на нём люди общаются о спорте, спортивном питании, комплексах упражнений и так далее, так что случайно наткнувшийся пользователь может заинтересоваться данной темой. Получение финансовой прибыли осуществляется путём размещения рекламы на форуме.

Сайт является одним из рекламно-информационных инструментов, способных дать первое представление о спорте.

Поскольку именно через Интернет потенциальные клиенты будут искать информацию о том, какой лучше выбрать спортивный центр для занятия спортом, был разработан форум, где люди могут оставлять различные записи о рекомендациях и вопросах, связанных со спортом.

Основной целью сайта будет являться популяризация спорта, за счет размещения историй пользователей о их тренировках, спортивном питании.

1. Характеристики и работа программы

Требования к логике форума, которые были выполнены в ходе разработки:

1. Вход на форум открыт для любого человека, имеющий доступ в интернет.
2. Регистрация, после чего авторизация, если аккаунт уже зарегистрирован, соответственно только авторизация, данный пункт обязателен так как оставить запись на сайте могут только авторизованные люди.
3. Создание записи, с поддержкой языка разметки Markdown.
4. Отправка записи в базу данных, для хранения и последующей модерации.
5. Модерация.
6. Вывод записи на форум.

Для того, чтобы развернуть сайт требуется:

1. Скачать и установить Open server panel по следующей ссылке <https://ospanel.io/>
2. После того, как была установлена данная программная среда для веб-разработки, следует запустить её, открыть настройки и выставить их в соответствии с рисунком 1.

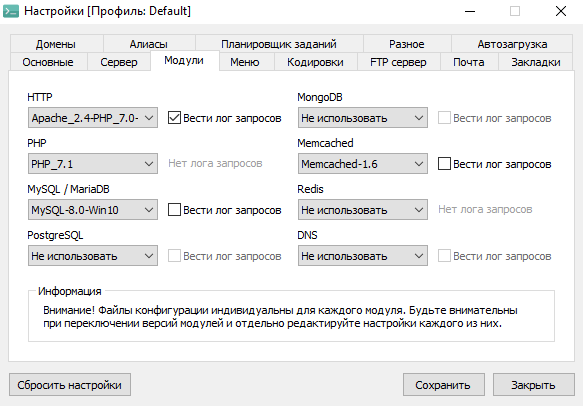


Рисунок 1 – Настройки Open Server Panel

1. После этого, во вкладке дополнительно появится элемент phpMyAdmin, представлено на рисунке 2.

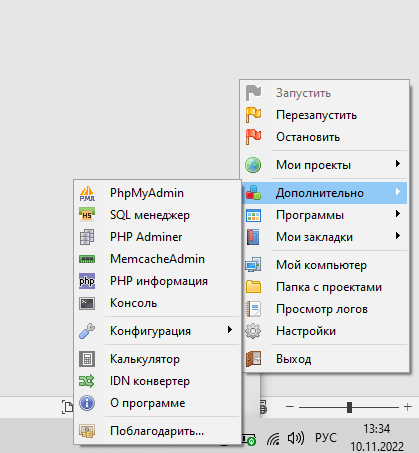


Рисунок 2 – Появление phpMyAdmin

1. Далее следует открыть phpMyAdmin и импортировать в него базу данных, представлено на рисунке 3.

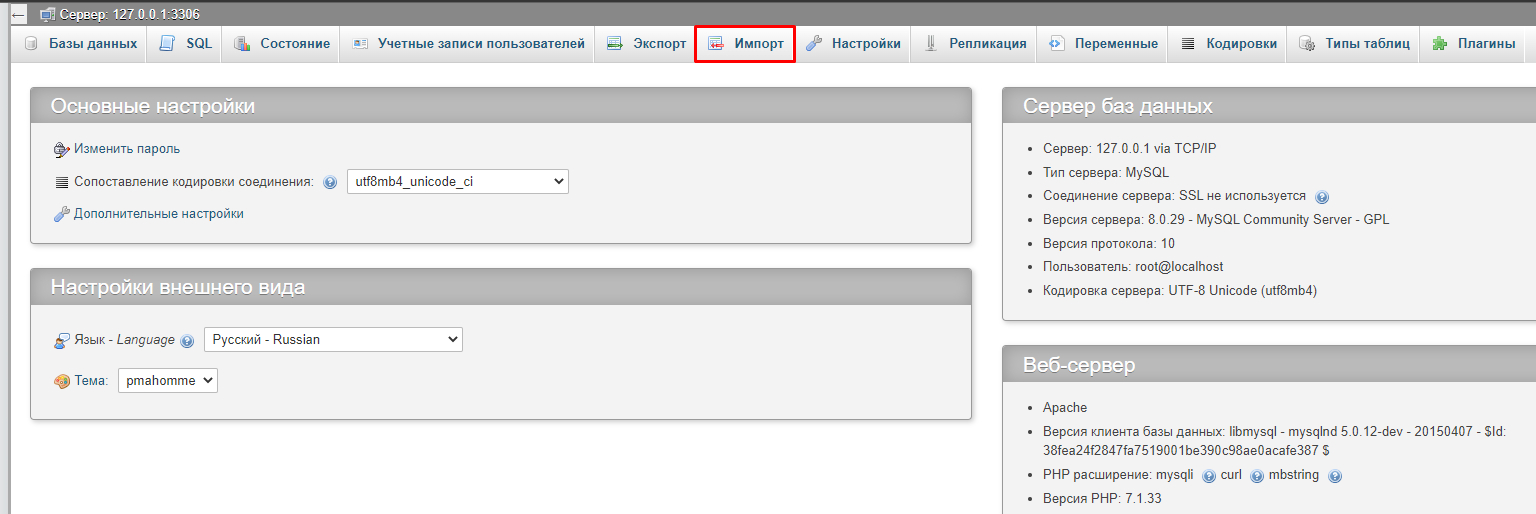


Рисунок 3 – Кнопка импортирования базы данных в phpMyAdmin

1. После импорта базы данных можно импортировать файлы самого форума, их следует разместить в папке, в которую можно попасть, нажав на элемент «Папка с проектами», представлено на рисунке 4.

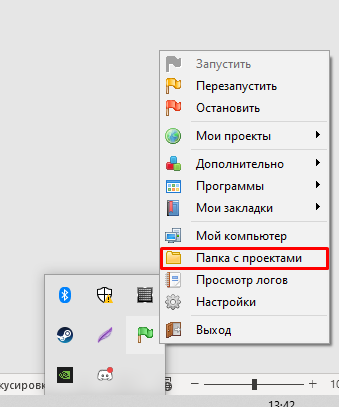


Рисунок 4 – Элемент «Папка с проектами»

1. Теперь, если всё было сделано правильно, после нажатия на элемент, во вкладке «Мои проекты», с названием папки, в которой лежат файлы вашего сайта, должен открыться ваш сайт. Представлено на рисунке 5.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 5 – Список проектов

На рисунке 6 представлена файловая архитектура форума.

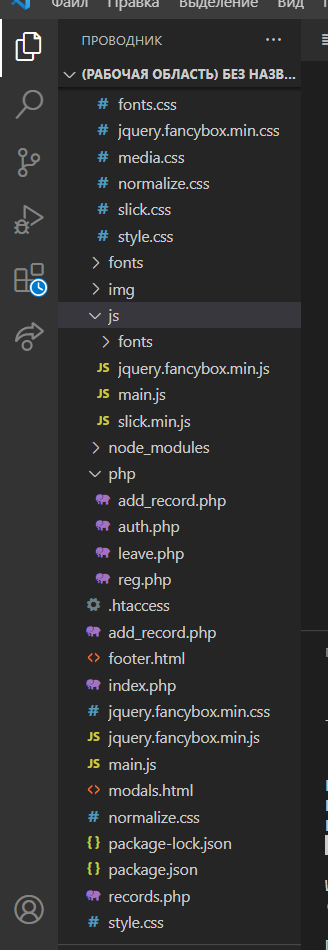


Рисунок 6 – Файловая архитектура форума

Для работы форума необходимы все файлы, представленные на рисунке 1, их можно разделить на следующие группы:

1. Страницы сайта
2. Элементы сайта
3. Стили
4. Коды клиентской части
5. Коды серверной части
6. Картинки
7. Шрифты
8. Плагины
9. Адаптивная вёрстка

При создании форума использовались языки JS для фронт разработки, PHP для бэка, HTML для разметки страницы и CSS для написания стилей элементов.

Сайт состоит из двух страниц index и records, данные файлы используют расширение .php так как он применяется для динамического отображения информации на форуме, остальные PHP файлы являются исполняемым кодом, необходимым для функционирования продукта. Все стили для элементов сайта находятся в файле style.css. Также модальные окна и подвальная часть сайта вынесены в отдельные файлы modals.html и footer.html, так как данные элементы повторяются на обоих страницах сайта и не имеют динамического отображения информации в них.

Также при создании форума использовались плагины: fancybox и slick. С помощью плагина fancybox были созданы модальные окна, а с помощью slick слайдер.

При создании форума учитывались 2 сценария: авторизован пользователь или нет. Если пользователь не авторизован, то у него нет прав на предложение записей, а если авторизан соответственно есть.

На рисунке 7 представлен вид сайта для неавторизованного пользователя.

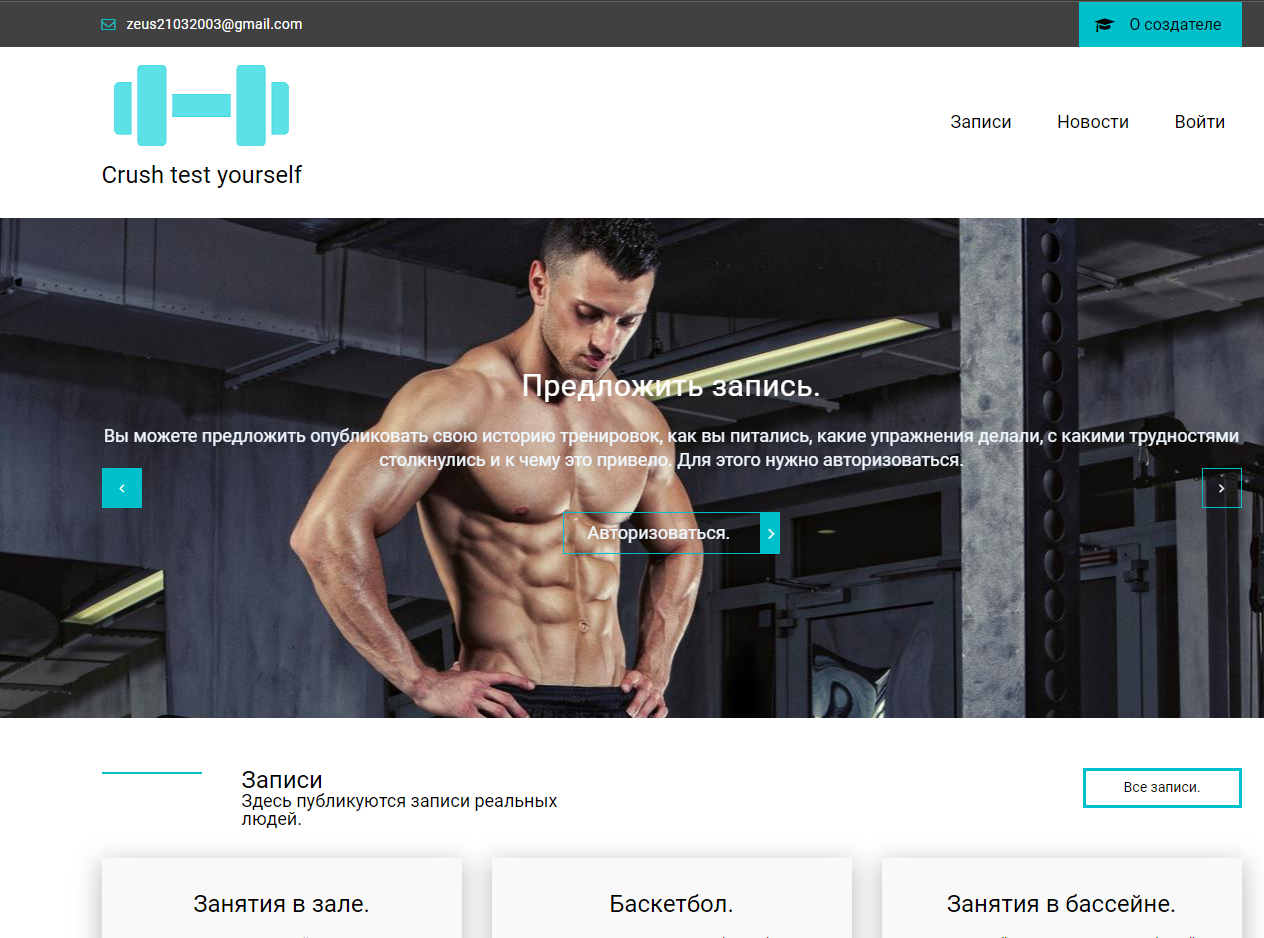


Рисунок 7 – Вид сайта для неавторизованного пользователя

На рисунке 8 представлен вид сайта для авторизированного пользователя.

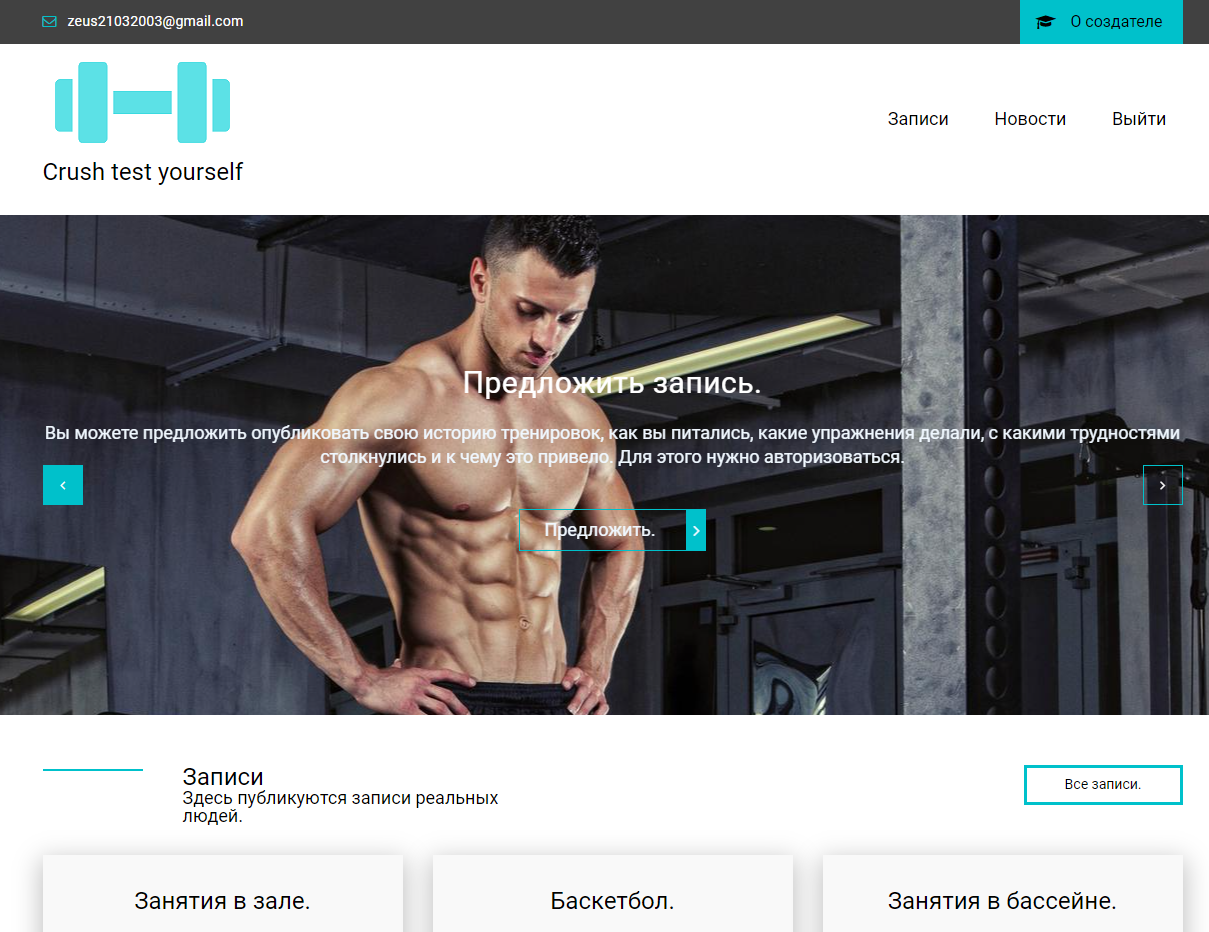


Рисунок 8 – Вид сайта для авторизованного пользователя

Такого поведения сайта удалось добиться благодаря динамическому отображению сайта и кэшированию данных. Когда пользователь авторизуется на сайте его имя заносится в кеш, что видит форум и динамически изменяет элементы на нём, что даёт возможность в главной функциональности сайта – предложению записи.

На рисунке 9 предсталено динамическое отображение информации на главной странице, в зависимости от авторизации.

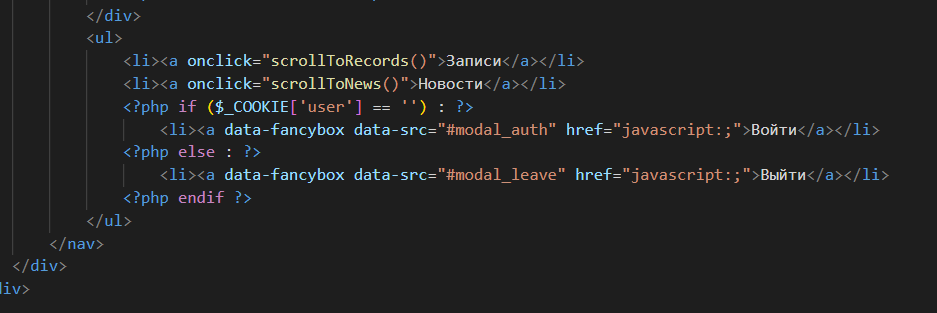


Рисунок 9 – Динамическое отображение кнопки в шапке сайта в зависимости от авторизации

На рисунке видно, что при нажатии на кнопку «Войти» отображается модальное окно авторизации, а при нажатии на кнопку «Выйти» - модальное окно выхода из аккаунта. Также при авторизации изменяется кнопка на слайдере, представлено на рисунке 10.

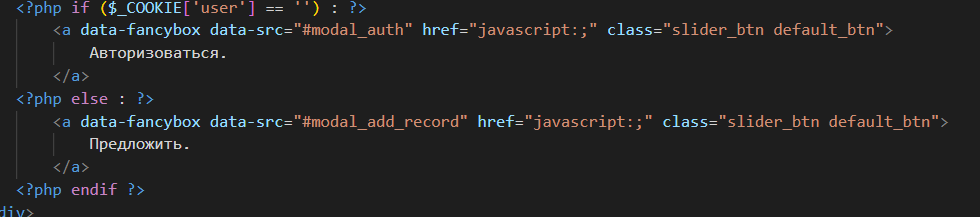


Рисунок 10 – Динамическое отобржаение кнопки на слайдере

На данном рисунке также видно, что при нажатии кнопки «Авторизоваться» отображается модальное окно авторизации, а при нажатии кнопки «Предложить» -модальное окно дабавления записи.

После авторизации становится доступен основной функционал сайта – предложение записей. Для этого после авторизации необходимо нажать на кнопку «Предложить», вследствие чего откроется модальное окно добавления записи, представлено на рисунке 11.

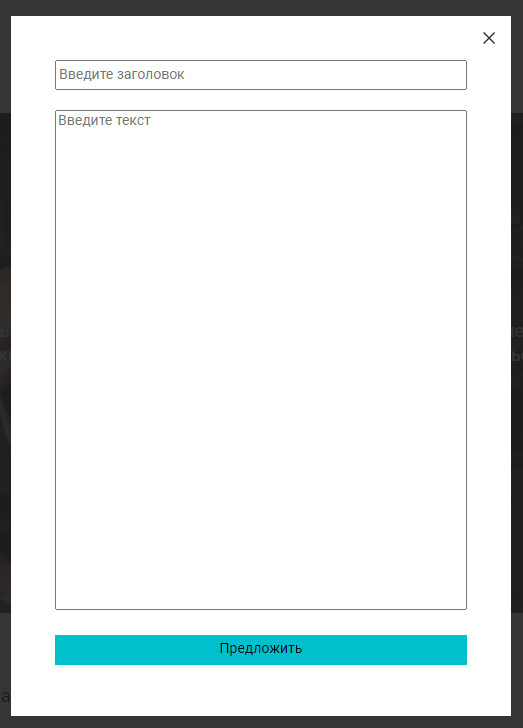


Рисунок 11 – Модальное окно добавления записи

В данном модальном окне необходимо заполнить поля «Заголовок» и «Текст», минимальная длина первого поля равна 2, второго 500, а максимальная первого 50 и 5000 соотвественно.

При нажатии на кнопку «Предложить», введённые пользователем данные отправляются в базу данных для последующей модерации.

На рисунке 12 представлен код добавления данных в базу данных.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 12 – Код добавления данных в базу данных

После того как данные пользователя приходят в базу данных, они модерируются: проверка содержимого на адекватность, проверка синтаксиса Markdown, проверка корректного отображения записи на сайте.

На рисунке 13 представлен вид записи в базе данных.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 13 – Вид записи в базе данных

За местоположение, где отображается запись отвечает свойство importance, если оно равно нулю это значит, что запись ещё находится на стадии модерации, если единице, то она опубликована на странице со всеми записями, а если двум, то на главной странице.

ЛИСТИНГ ПРОГРАММЫ

**«Добавление записи»**

<?php

$dbHost = "localhost";

$user = "root";

$pass = "root";

$dbName = "mydb";

$mysqli = mysqli\_connect($dbHost, $user, $pass, $dbName);

if ($mysqli == false) {

    echo ("Ошибка подключения к БД");

}

$title = $\_POST['title'];

if(mb\_strlen($title) < 2 || mb\_strlen($title)>50){

    echo("Недопустимая длина заголовка, минимум 4, максимум 50 символов");

    exit();

}

$text = $\_POST['text'];

if(mb\_strlen($text) < 500 || mb\_strlen($text)>5000){

    echo("Недопустимая длина текста, минимум 500, максимум 5000 символов");

    exit();

}

mysqli\_query($mysqli, "INSERT INTO `records` (`title`, `text`, `author`) VALUES ('$title', '$text', '$\_COOKIE[user]')");

$mysqli -> close();

echo("Регистрация прошла успешно!");

header('Location: /');

exit();

?>