

## Система добавления комментариев на вебсайты

*Кочитов Михаил Евгеньевич*

*Приамурский государственный университет им. Шолом-Алейхема*

*студент*

### Аннотация

В данной статье рассматривается система добавления комментариев на вебсайты. В статье будет приведен собственный пример с реализацией данной системы, используя язык программирования PHP и готовый шаблон стилей CSS от Materialize.

**Ключевые слова:** комментарии, добавление комментариев, система, вебсайт, PHP, CSS, Materialize,

## The system of adding comments on websites

*Kochitov Mikhail Evgenevich*

*Sholom-Aleichem Priamursky State University*

*student*

### Abstract

This article discusses a system for adding comments to websites. The article will provide its own example with the implementation of this system, using the PHP programming language and a ready-made CSS style sheet from Materialize.

**Keywords:** comments, adding comments, system, website, PHP, CSS, Materialize,

В настоящее время большинство сайтов, форумов, блогов, видео блогов, социальных сетей, новостей имеют возможность добавлять комментарии пользователям, чтобы оставить свое мнение или отзыв о выложенном данном материале. Оставлять комментарии в интернете очень полезно, так как автору новости или видео нужно знать, как его аудитория оценит качество материала, опубликованное им самим, чтобы он в дальнейшем выпускал материалы довольно качественными и интересными, опираясь на комментарии его пользователей.

В статье Н.А. Мясниковой и Н.Д. Курина рассматривается проектирование информационной системы для ведения блога [1]. Рассматривая статью А.А. Шайдурова и Д.В. Лучанинова можно увидеть разработку форума для личного блога [2]. С.М. Щербаков в своей статье рассматривает моделирование интернет-приложений электронной коммерции [3]. В статье И.О. Щеголева рассматриваются обязательные и дополнительные элементы информационного сайта [4]. Рассматривая статью Е.С. Левшиной и М.А. Иванько можно заметить информационную систему для социальной сети [5].

Многие сайты пользуются готовыми системами комментариев, самый популярный это виджет комментариев от социальной сети «ВКонтакте» (рисунок 1), после него идет виджет комментариев от известной социальной сети «Facebook» (рисунок 2) и мульти система комментариев «Disqus» (рисунок 3) и другие менее популярные.

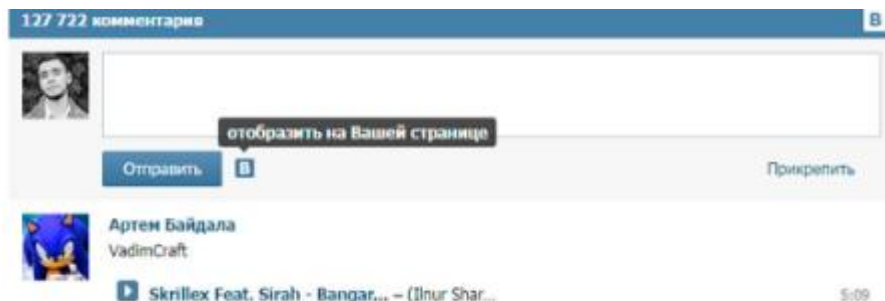


Рисунок 1. Система комментариев от ВКонтакте

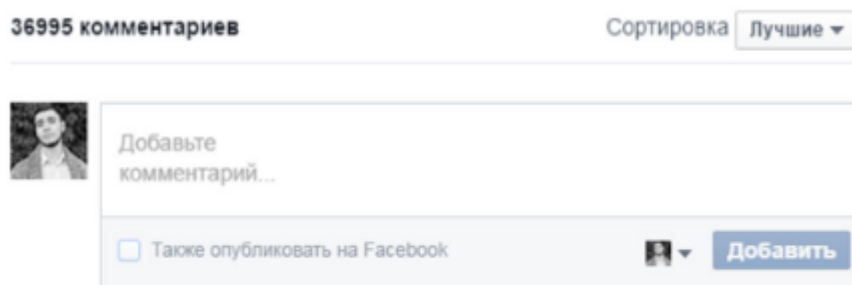


Рисунок 2. Система комментариев от Facebook

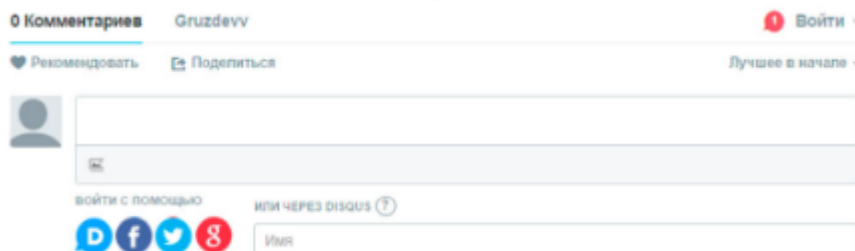


Рисунок 3. Система комментариев от Disqus

Целью данной статьи является использование возможности добавления комментариев на вебсайты. Также будет разработан собственный пример с реализацией данной возможности, используя язык программирования PHP и готовый шаблон стилей CSS от Materialize [6].

Теперь перейдем к разработке собственной системы добавления комментариев. Для начала создадим на локальном сервере базу данных «db\_comments» и в ней добавим новую таблицу под названием «comments».

#	Имя	Тип данных	Длина/Знач...	Беззнаковое	Разрешить NULL	Zerofill	По умолчанию
1	id	INT	11	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	AUTO_INCREMENT
2	name	VARCHAR	50	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	0
3	comment	VARCHAR	255	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	0
4	datetime	VARCHAR	50	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	0

Рисунок 4. Таблица «comments»

На рисунке 4 изображена таблица «comments» с добавленными в нее четырьмя столбцами: id – уникальный идентификатор комментария, name – имя пользователя, оставившего свой комментарий, comment – собственно сам комментарий, datetime – дата и время добавления комментария.

Далее перейдем к созданию страницы, в которой можно будет оставлять комментарии. Для начала создадим на локальном сервере файл «comment.php» и в нем напишем код (рисунок 5)

```

<html>
<head>
<meta charset="UTF-8">
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/materialize/1.0.0/css/materialize.min.css">
<script src="https://cdnjs.cloudflare.com/ajax/libs/materialize/1.0.0/js/materialize.min.js"></script>
<title>Система добавления комментариев</title>
</head>
<body>
<div class="container">
<h4>Система добавления комментариев</h4>
<form action="comment.php" method="POST">
<div class="row">
<div class="input-field col s12">
<input name="name" id="name" type="text" class="validate" required>
<label for="name">Ваше имя</label>
</div>
</div>
<div class="row">
<div class="input-field col s12">
<textarea name="comment" id="comment" class="materialize-textarea" required></textarea>
<label for="comment">Ваш комментарий</label>
</div>
</div>
<button class="btn waves-effect waves-light" type="submit" name="action">Добавить комментарий</button>
</form>

```

Рисунок 5. HTML – код отображения формы добавления комментария

На рисунке 5 изображен HTML – код, который отобразит на странице форму добавления комментария. В форму включены три элемента: первый – поле «Ваше имя», второй – текстовое многострочное поле «Ваш комментарий» и кнопка «Добавить комментарий». Данная форма использует красивый шаблон стиля CSS от Materialize поэтому в начале страницы грузятся их файлы CSS и JS с облачного хранилища «cloudflare». При нажатии на кнопку «Добавить комментарий» произойдет перезагрузка страницы и в таблицу «comments» добавиться запись о новом комментарии, код реализующий эту функцию представлен на рисунке 6.

```
<?php
if ($_POST['name'] != "" && $_POST['comment'] != "") {
    $link = mysqli_connect('localhost', "root", '', 'db_comments');
    if ( !$link ) die("Error");
    $date_time = date('d/m/Y H:i:s');
    $query = "INSERT INTO comments (name, comment, datetime) VALUE ('".$_POST['name']."', '".$_POST['comment']."', '".$_date_time."')";
    $result = mysqli_query( $link, $query );
}
?>
```

Рисунок 6. PHP – код добавления нового комментария в таблицу «comments»

На рисунке 6 представлен PHP – код, который реализует функцию добавления нового комментария в таблицу «comments». Теперь рассмотрим этот код более подробно. Для начала идет проверка полученных переменных от формы `$_POST['name']` (имя пользователя) и `$_POST['comment']` (комментарий пользователя), если эти переменные не пустые, то есть условие выполняется, то идет процесс обработки и добавления комментария. Чтобы подключиться к базе данных потребуется функция «`mysqli_connect`», в которую обязательно необходимо передать параметры: `localhost` (имя хоста), `root` (имя пользователя), пустой пароль и `db_comments` (название базы данных). Если при подключении возникла ошибка, то прерывается весь процесс добавления комментария и выдается ошибка, используя функцию «`die`». При успешном подключении в переменную `$link` заносится информация о текущем подключении к базе данных и она будет главным инструментом по управлению базы данных на сервере с возможностью добавления новой записи в таблицу. Далее функция «`date`» обрабатывает локальное время сервера в формат: (номер дня/номер месяца/год часы:минуты:секунды), это время заносится в переменную `$date_time`. Напоследок остается формирование SQL запроса для добавления новой записи в таблицу «comments», этот запрос сохраняется в переменную `$query`. В запросе указываются названия столбцов и их значения, полученные от формы и локальное время. После этого функция «`mysqli_query`» уже исполняет данный запрос и в базу данных в таблицу «comments» добавляется новый комментарий пользователя.

Теперь осталось рассмотреть PHP – код отображающий блок всех добавленных комментариев пользователей (рисунок 7).

```

<?php
$link = mysqli_connect('localhost', "root", '', 'db_comments');
if ( !$link ) die("Error");
$query = "SELECT COUNT(*) FROM comments";
$result = mysqli_query( $link, $query );
$count = mysqli_fetch_row($result);
if ($count[0] > 0) {
    ?>
    <h5>Все комментарии:</h5>
    <?php
    $query = "SELECT * FROM comments";
    $result = mysqli_query( $link, $query );
    while ( $row = mysqli_fetch_assoc($result) )
    {
        ?>
        <div class="card-panel">
            <b><?= $row['name'] ?></b> <?= $row['datetime'] ?><br>
            <hr>
            <?= $row['comment'] ?>
        </div>
    <?php
    }
}
?>
</div>
</body>
</html>

```

Рисунок 7. PHP – код отображения всех комментариев из таблицы «comments»

На рисунке 7 изображен PHP - код отображения всех добавленных комментариев. Теперь рассмотрим этот код более подробно. Для начала идет подключение к базе данных с помощью функции «`mysqli_connect`» (как это было в PHP - коде на рисунке 6). Далее выполняется запрос о получении текущего количества записей в таблице, то есть сколько на данный момент комментариев в таблице «`comments`», значение о количестве комментариев заносится в переменную `$count[0]`. Далее идет проверка, что если `$count[0]` больше нуля, то есть в таблице «`comments`» есть записи, то идет процесс отображения всех этих записей. Для этого формируется SQL запрос для получения всех записей таблицы и после в цикле «`while`» с помощью функции «`mysqli_fetch_assoc`» в переменную `$row` передаются вся информация о каждой записи: `$row['name']` – имя пользователя, `$row['datetime']` – дата и время добавления комментария, `$row['comment']` – собственно сам комментарий пользователя. Каждый комментарий оформляется в блок «`div`» с использованием класса «`card-panel`», чтобы дать более красивый дизайн с эффектом отбрасывания тени. Тег «`hr`» создает горизонтальную линию для разделения.

Далее осталось протестировать собственную систему добавления комментариев.

## Система добавления комментариев

Ваше имя

Ваш комментарий

ДОБАВИТЬ КОММЕНТАРИЙ

Все комментарии:

Михаил 24/12/2018 06:13:46

Мой познавательный комментарий

Марина 24/12/2018 07:27:53

Этот комментарий сделан в качестве примера

Андрей 24/12/2018 07:28:32

Система добавления комментариев работает успешно

## Рисунок 8. Наглядный вид рабочей системы добавления комментариев

Рассматривая рисунок 8 можно увидеть сверху саму форму добавления нового комментария, где имеются два поля: имя пользователя, комментарий пользователя и кнопка добавления нового комментария. Также под формой идет список всех добавленных комментариев с отображением каждого имени пользователя, даты и времени добавления комментария и самого комментария под горизонтальной линией.

Таким образом, была рассмотрена система добавления комментариев на вебсайты. Также был разработан собственный пример с реализацией данной системы, используя язык программирования PHP и готовый шаблон стиля CSS от Materialize. Данная система комментариев очень востребована тем сайтам, в которых публикуются разного рода материалы.

**Библиографический список**

1. Мясникова Н.А., Курин Н.Д. Проектирование информационной системы для ведения блога // Объектные системы. 2015. № 10 (10). С. 57-63.
2. Шайдулов А.А., Лучанинов Д.В. Разработка форума для личного блога // Постулат. 2018. № 1 (27). С. 35.
3. Щербаков С.М. Моделирование интернет-приложений электронной коммерции // Studium. 2010. № 2 (15). С. 9.
4. Щеголев И.О. Обязательные и дополнительные элементы информационного сайта // Экономика. Бизнес. Информатика. 2016. Т. 2. № 3. С. 160-168.
5. Левшина Е.С., Иванько М.А. Информационная система для социальной сети // Вестник МГУП имени Ивана Федорова. 2013. № 2. С. 106-112.
6. Materialize CSS. URL: <https://materializecss.com> (дата обращения

24.12.2018)

7. Система комментариев для сайта, что выбрать: Disqus, Sackle... // URL: <https://guzdevv.ru/sistema-kommentariev-dlya-sajta/> (дата обращения 24.12.2018)