

Форум с комментариями

Текст задания

Разработать и реализовать клиент-серверную информационную систему, реализующую механизм CRUD.

Ход работы

- Спроектировать пользовательский интерфейс
- Описать пользовательские сценарии работы
- Описать API сервера и хореографию
- Описать структуру базы данных
- Описать алгоритмы

1. Пользовательский интерфейс

2. Пользовательские сценарии работы

Пользователь попадает на главную страницу index.php. Вводит любое текстовое сообщение, по желанию может добавить картинку, следует нажать при создании поста на кнопку выбрать файл. В случае корректного ввода данных, его сообщение появится на общей стене в обратном хронологическом порядке, сначала новые, затем старые публикации. Пользователи могут ставить лайки на понравившиеся записи или убирать их в противном случае. Есть возможность удалять записи. Для этого пользователь нажимает на кнопку УДАЛИТЬ, и соответствующая запись удаляется. Пользователь может увидеть время каждого поста и добавить к ним комментарии.

3. API сервера и хореография

4. Структура базы данных

Таблица posts

Название	Тип	Длина	По умолчанию	Описание
id	INT		NO	Автоматический идентификатор поста
text	TEXT		NO	Текст поста
dtime	TEXT		NULL	Дата создания поста
likes	INT		0	Количество лайков
image	VARCHAR	255	NO	Путь к файлу

Таблица *likes*

Название	Тип	Длина	NULL	Описание
id	INT		NO	Автоматический идентификатор лайка
userid	INT		NO	ID пользователя
postid	INT		NO	ID поста

Таблица *comments*

Название	Тип	Длина	NULL	Описание
id	INT		NO	Автоматический идентификатор лайка
userid	INT		NO	ID пользователя
postid	INT		NO	ID поста

5. Алгоритмы

1. [Добавление записи](#)
2. [Удаление записи](#)
3. [Реакция](#)
4. [Комментарии](#)

6. Примеры HTTP запросов/ответов

GET /web2.2/index.php HTTP/1.1

Host: localhost

Accept:

text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,;q=0.8,application/signed-exchange;v=b3;q=0.9

sec-ch-ua: "Not?A_Brand";v="8", "Chromium";v="108", "Google Chrome";v="108"

sec-ch-ua-mobile: ?0

sec-ch-ua-platform: "Windows"

Upgrade-Insecure-Requests: 1

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/108.0.0.0

Safari/537.36

HTTP/1.1 200 OK

Connection: Keep-Alive

Content-Type: text/html; charset=UTF-8

Date: Sat, 24 Dec 2022 07:13:25 GMT

Keep-Alive: timeout=5, max=100

Server: Apache/2.4.54 (Win64) PHP/8.1.10

Transfer-Encoding: chunked

X-Powered-By: PHP/8.1.10

POST /web2.2/comments.php HTTP/1.1
Host: localhost
Accept: /
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
sec-ch-ua: "Not?A_Brand";v="8", "Chromium";v="108", "Google Chrome";v="108"
sec-ch-ua-mobile: ?0
sec-ch-ua-platform: "Windows"
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/108.0.0.0 Safari/537.36
X-Requested-With: XMLHttpRequest

HTTP/1.1 200 OK
Connection: Keep-Alive
Content-Length: 1
Content-Type: text/html; charset=UTF-8
Date: Sat, 24 Dec 2022 07:15:05 GMT
Keep-Alive: timeout=5, max=100
Server: Apache/2.4.54 (Win64) PHP/8.1.10
X-Powered-By: PHP/8.1.10

POST /web2.2/add.php HTTP/1.1
Host: localhost
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,/;q=0.8,application/signed-exchange;v=b3;q=0.9
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary0sXWcyVlfQpnRuKk
sec-ch-ua: "Not?A_Brand";v="8", "Chromium";v="108", "Google Chrome";v="108"
sec-ch-ua-mobile: ?0
sec-ch-ua-platform: "Windows"
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/108.0.0.0 Safari/537.36

HTTP/1.1 302 Found
Connection: Keep-Alive
Content-Length: 40
Content-Type: text/html; charset=UTF-8
Date: Sat, 24 Dec 2022 07:16:13 GMT
Keep-Alive: timeout=5, max=100
Location: index.php
Server: Apache/2.4.54 (Win64) PHP/8.1.10
X-Powered-By: PHP/8.1.10

7. Важные части кода

1. Функция добавления комментария с помощью AJAX

```
$(document).ready(function() {  
    $(".btn-danger").click(function(){  
        var postId1 = $(this).data('id');// При нажатии на кнопку  
        var author = $("#author" + postId1).val(); // Получаем имя автора комментария  
        var message = $("#message" + postId1).val();// Получаем само сообщение  
        $.ajax({ // Аякс  
            type: "POST", // Тип отправки "POST"  
            url: "../comments.php", // Куда отправляем(в какой файл)  
            data: {"author": author, "message": message, "postId1":postId1}, // Что передаем и под каким значением  
            cache: false, // Убираем кеширование  
            success: function(response){ // Если все прошло успешно  
                var messageResp = ['Ваше сообщение отправлено','Сообщение не отправлено Ошибка базы данных','Нельзя отправлять пустые сообщения'];  
                var resultStat = messageResp[Number(response)];  
                if(response == 0){  
                    $("#author").val("");  
                    $("#message").val("");  
                    $("#commentBlock").append("<div class='comment'>Автор: <strong>"+author+"</strong><br>+message+</div>");  
                    $("#resp").text(resultStat).show().delay(1500).fadeOut(800);});return false;});});
```

2. Добавление данных

<?php

```
$result = $_POST['list'];

$uploadaddir = './img/';
$uploadfile = $uploadaddir . basename($_FILES['file']['name']);

move_uploaded_file($_FILES['file']['tmp_name'], $uploadfile);

if(!move_uploaded_file($_FILES['file']['tmp_name'], $uploadfile)){
    echo "Ошибка загрузки файла";
}
if (($result == '') or ($_FILES['file']['name'] == '')) {
    header('Location: index.php');
}
require 'db.php';
date_default_timezone_set('Europe/Moscow');
$time = date("Y-d-m H:i:s");

$resultall = mysqli_query($db, "INSERT INTO posts (text,likes,dtime,image) VALUES('$result','0','$time','$uploadfile')");

header('Location: index.php');|
```

3. Функция проставления лайков с помощью AJAX

```
$(document).ready(function(){
    // when the user clicks on like
    $('.like').on('click', function(){
        var postid = $(this).data('id');
        $post = $(this);

        $.ajax({
            url: 'index.php',
            type: 'post',
            data: {
                'liked': 1,
                'postid': postid
            },
            success: function(response){
                $post.parent().find('span.likes_count').text(response + " likes");
                $post.addClass('hide');
                $post.siblings().removeClass('hide');
            }
        });
    });

    // when the user clicks on unlike
    $('.unlike').on('click', function(){
        var postid = $(this).data('id');
        $post = $(this);

        $.ajax({
            url: 'index.php',
            type: 'post',
            data: {
                'unliked': 1,
                'postid': postid
            },
            success: function(response){
                $post.parent().find('span.likes_count').text(response + " likes");
                $post.addClass('hide');
                $post.siblings().removeClass('hide');
            }
        });
    });
});|
```