

# HTTP аутентификация

---

## Цель работы

Спроектировать и разработать систему авторизации пользователей на протоколе HTTP

## Минимальные требования

- функциональность входа/выхода
- хранение паролей в хешированном виде

## Дополнительные возможности

- форма регистрации
- смена пароля
- восстановление пароля
- ограничение времени сессии (на сервере)
- ограничение срока действия пароля (на сервере)
- валидация пароля (на сервере)
- хранение хеша пароля с солью
- возможность использования одновременного использования одним пользователем нескольких клиентов

Вид окон в системе: окно регистрации:

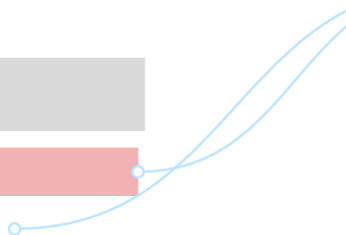
Логин

Пароль

Повторите пароль

Зарегистрироваться

У вас уже есть аккаунт? [Авторизуйтесь](#)



Окно входа:

Логин

Пароль

Войти

У вас нет аккаунта?

Зарегистрируйтесь

Забыли пароль?

Восстановить

Окно восстановления пароля:

Логин

Пароль

Повторите пароль

Новый пароль

отмена

Окно смены пароля:

Старый пароль

выход

Повторите пароль

Сменить пароль

Главная страница:

Профиль

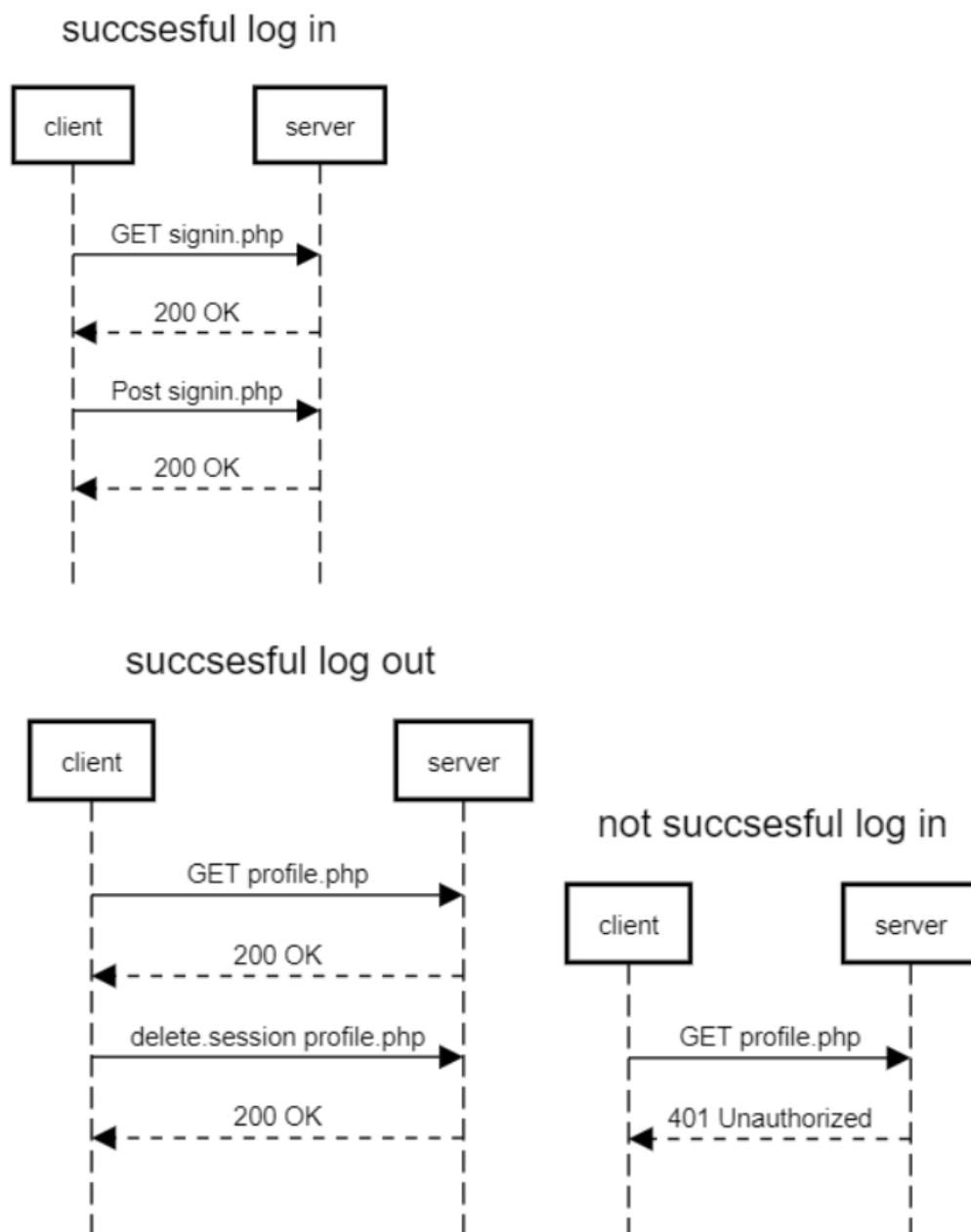
сменить пароль

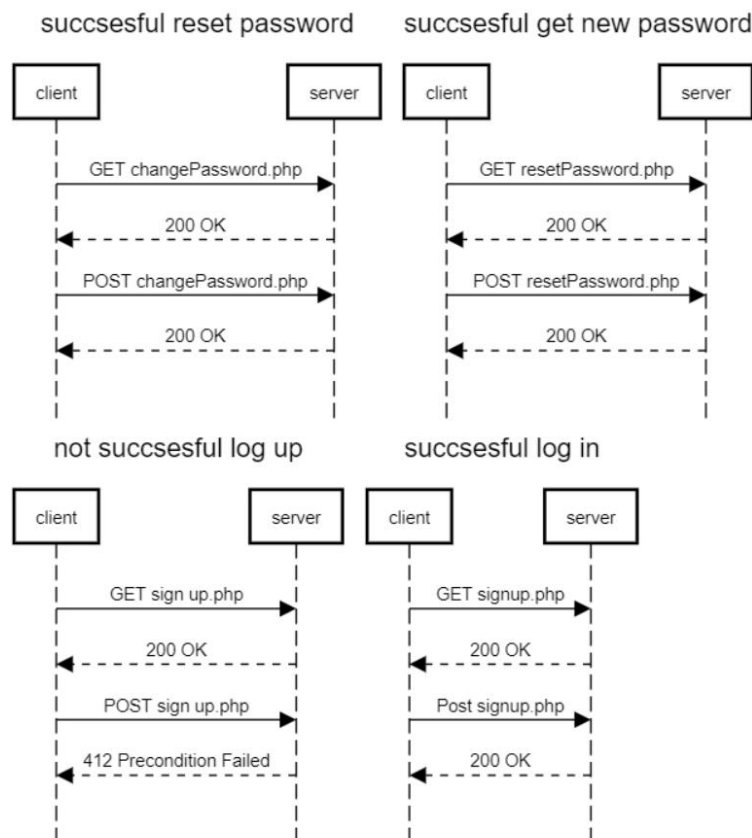
выход

Возможные пользовательские сценарии:

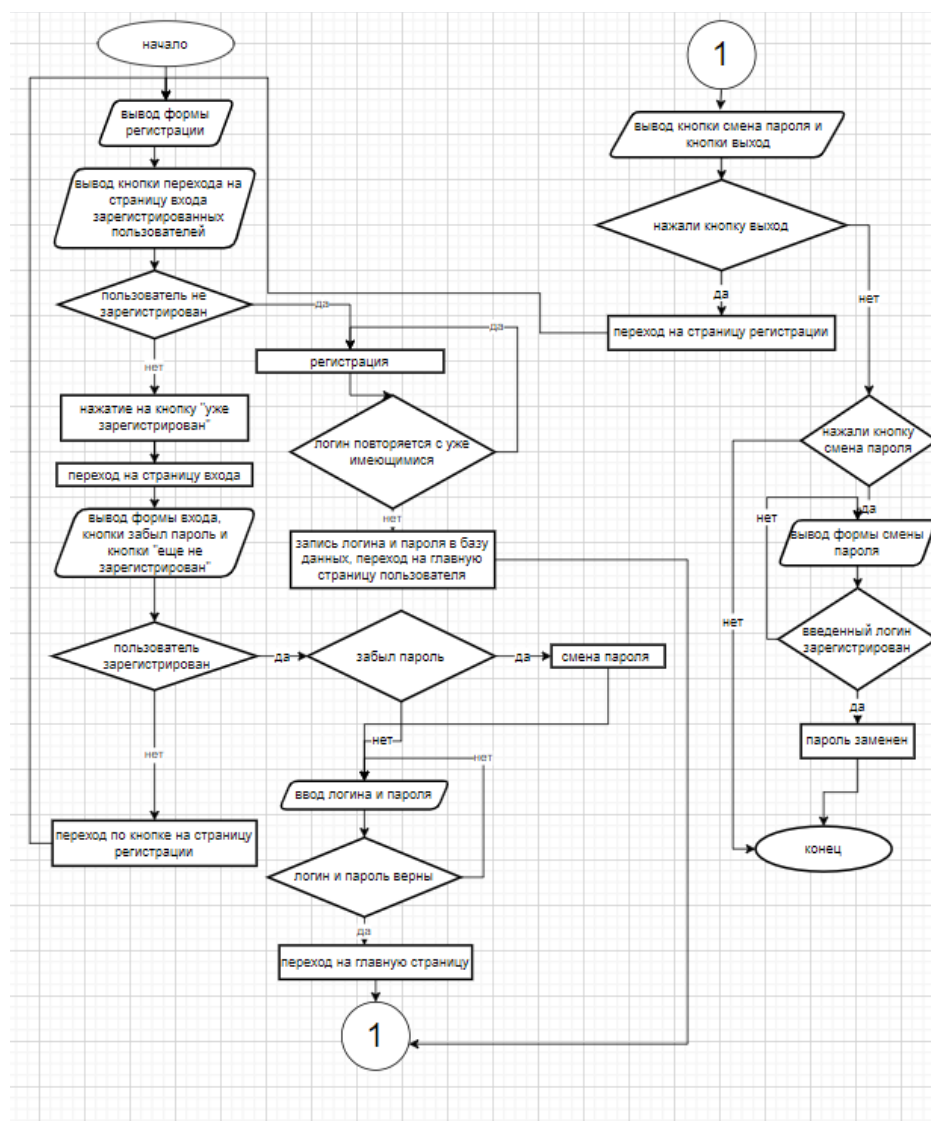
- 1)Регистрация: пользователь заходит на сайт регистрации, вводит логин и пароль с его повторением. После чего пароль и логин записываются в базу данных, а пользователя переносят на страницу входа.
- 2)Вход: пользователь вводит логин и пароль, они проверяются в базе данных, и пользователя переносят на главную страницу.
- 3)Восстановление пароля: пользователь нажимает на кнопку "восстановить", после этого его переносят на страницу восстановления пароля, где он вводит свой логин и новый пароль, после чего у него меняется пароль.
- 4)смена пароля: пользователь на главной странице нажимает на кнопку "сменить пароль", после чего его переносят на страницу смены пароля, где пользователь вводит старый пароль и новый с его повторением, после чего пароль сменяется.

Примеры запросов на сервер и ответы на них:





Работа сайта на примере блок схемы (алгоритмы использующиеся на сайте):



База данных, используемая сайтом — это MySQL. Состоит таблица базы данных из 4 столбцов:

- 1)id
- 2)login
- 3)password
- 4)time.

Основные фрагменты кода:

1)код регистрации:

```
$login = $_POST['login'];
$password = $_POST['password'];
$password_confirm = $_POST['password_confirm'];

    $number = preg_match('@[0-9]@', $password);
    $lowercase = preg_match('@[a-z]@', $password);
    if($login === ""){
        $_SESSION['message'] = 'придумайте логин';
        header('Location: ../register.php');
    }

    if(strlen($password) < 4 || !$number || !$lowercase) {
        $_SESSION['message'] = 'В пароля должна быть цифра и буква, его длина должна быть больше 4';
        header('Location: ../register.php');
    }
    else{

if ($password === $password_confirm) {

        $password = $password . 2**strlen($password);
        $password = md5($password);

        mysqli_query($connect, "INSERT INTO `user` (`id`, `login`, `password`) VALUES (NULL, '$login', '$password')");

        $_SESSION['message'] = 'Регистрация прошла успешно!';
        header('Location: ../index.php');

    } else {
        $_SESSION['message'] = 'Пароли не совпадают';
        header('Location: ../register.php');
    }
}
```

2)код авторизации:

```
$login = $_POST['login'];
$password = $_POST['password'];
    $password = $password . 2**strlen($password);
    $password = md5($password);

    $check_user = mysqli_query($connect, "SELECT * FROM `user` WHERE `login` = '$login' AND `password` = '$password'");
    if (mysqli_num_rows($check_user) > 0) {

        $user = mysqli_fetch_assoc($check_user);

        $_SESSION['user'] = [
            "id" => $user['id']
        ];

        header('Location: ../profile.php');

    } else {
        $_SESSION['message'] = 'Не верный логин или пароль';
        header('Location: ../index.php');
    }
}
```

### 3)код смены пароля:

---

```
$id = $_SESSION['user']['id']; // id юзера из сессии
$query = "SELECT * FROM user WHERE id='$id'";

$result = mysqli_query($connect, $query);
$user = mysqli_fetch_assoc($result);

$hash = $user['password']; // соленный пароль из БД
$password = $_POST['password'];
$password = $password . 2**strlen($password);
$password = md5($password);
$newpassword = $_POST['new_password'];
$newpassword = $newpassword . 2**strlen($newpassword);
$newpassword = md5($newpassword);

// Проверяем соответствие хеша из базы введенному старому паролю
if ($password === $hash) {

    $query = "UPDATE user SET password='$newpassword' WHERE id='$id'";
    mysqli_query($connect, $query);
    $_SESSION['message'] = 'Пароль сменен';
    header('Location: ../changePassword.php');
} else {
    $_SESSION['message'] = 'Старый пароль не верен';
    header('Location: ../changePassword.php');
    // старый пароль введен неверно, выведем сообщение
}
```