**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»**

**Факультет безопасности информационных технологий**

**Дисциплина:**

«Разработка систем аутентификации и криптографии»

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №2**

**Методы аутентификации**

**Выполнили:**

Студентка гр. N42514c

Холоденина А.В.

Изображение выглядит как вычерчивание линий

Автоматически созданное описание

**Проверил:**

Фёдоров И.Р.

Санкт-Петербург

2020 г.

**Оглавление**

[**Цель работы** 3](#_Toc57928569)

[**Описание выбранных средств реализации и обоснование выбора** 4](#_Toc57928570)

[**Описание алгоритма** 5](#_Toc57928571)

[**Исходный код** 6](#_Toc57928572)

[**Демонстрация работы программы** 9](#_Toc57928573)

[**Выводы** 9](#_Toc57928574)

# **Цель работы**

Требования к реализации программы:

* необходимо реализовать метод аутентификации в клиент-серверном приложении согласно варианту
* клиент должен представлять собой веб-страницу с формой авторизации пользователя
* сервер должен включать в себя две части:

1) таблица идентификаторов (данные о пользователях для аутентификации: логин/пароль/токен/и т. д. в зависимости от метода аутентификации);

2) процесс с реализованной логикой метода аутентификации.

* реализовать аутентификацию по паролю с хешированием MD5 (можно использовать функцию из библиотеки).
* таблице идентификаторов должны храниться: логин, хеш пароля (md5), срок действия пароля.
* таблица идентификаторов должна представлять собой таблицу в реляционной БД, данные должны передаваться через SQL-запросы.
* при истечении срока действия пароля аутентификация не должна проходить.
* при аутентификации на сервере сравниваются не пароли, а результаты MD5.

# **Описание выбранных средств реализации и обоснование выбора**

Для реализации алгоритма был выбран язык php, который отличается своей простотой и широтой применения.

PHP (рекурсивный акроним словосочетания PHP: Hypertext Preprocessor) - это распространенный язык программирования общего назначения с открытым исходным кодом. PHP специально сконструирован для веб-разработок и его код может внедряться непосредственно в HTML.

Так же для соединения с базой данных была выбрана библиотека RedBeanPHP – это мощная ORM для PHP, которая значительно упрощает работу с базами данных.

ORM или Object-relational mapping (Объектно-реляционное отображение) — это технология программирования, которая позволяет преобразовывать несовместимые типы моделей в ООП, в частности, между хранилищем данных и объектами программирования.

# **Описание алгоритма**

В таблице идентификаторов хранятся: логин, хеш пароля (md5), срок действия пароля. Реализована аутентификация по паролю с хешированием. Таблица идентификаторов так же представляет собой таблицу в реляционной БД, данные должны передаваться через SQL-запросы. При аутентификации на сервере сравниваются не пароли, а результаты MD5, и email. При истечении срока действия пароля аутентификация не проходит. При совпадении email и хешей паролей аутентификация считается успешной и происходит переход на страницу с приветствием пользователя.

# **Исходный код**

Создание базы данных:

*CREATE TABLE `users`(*

*id int (10) AUTO\_INCREMENT,*

*name varchar(20) NOT NULL,*

*login varchar(50) NOT NULL,*

*password varchar(256) NOT NULL,*

*data varchar(15) NOT NULL,*

*PRIMARY KEY (id)*

*);*

Конкретные данные:

*INSERT INTO users*

*('id', 'name', 'login', 'password', 'data')*

*VALUES (1 , "user", "myuser", "6ad14ba9986e3615423dfca256d04e3f", "2021-12-03")*

Создаем подключение к базе данных (созданной ранее в phpAdmin):

|  |
| --- |
| <?php |
|  | // библиотека RedBeanPHP |
|  | require "lib/rb-mysql.php"; |
|  |  |
|  | // подключаемся к БД |
|  | R::setup( 'mysql:host=localhost;dbname=login\_users', |
|  | 'root', '' ); |
|  |  |
|  |  |
|  | if(!R::testConnection()) die('No DB connection!'); |
|  |  |
|  | session\_start(); // сессия для авторизации |
|  | ?> |

Вход и проверка введенных данных:

|  |
| --- |
| <?php |
|  | $title="Парольная аутентификация"; |
|  | require \_\_DIR\_\_ . '/header.php'; |
|  | require "db.php"; |
|  | ?> |
|  | <!DOCTYPE html> |
|  | <html lang="ru"> |
|  | <head> |
|  | <title><?php echo $title; ?></title> |
|  | <link rel="stylesheet" type="text/css" href="css/bootstrap.min.css"> |
|  | <meta content="text/html; charset=utf-8"> |
|  | </head> |
|  | <body> |
|  |  |
|  |  |
|  | <!--Форма--> |
|  | <div class="container-fluid"> |
|  | <div class="row"> |
|  | <div class="col-md-4"> |
|  | <h2>Вход</h2> |
|  | <form action="login.php" method="post"> |
|  | <input type="text" class="form-control" name="login" id="login" placeholder="Введите логин" required><br> |
|  | <input type="password" class="form-control" name="password" id="pass" placeholder="Введите пароль" required><br> |
|  | <button class="btn btn-success" name="action\_login" type="submit">Войти</button> |
|  | </form> |
|  | <br> |
|  | </div> |
|  | </div> |
|  | </div> |
|  | </body> |
|  | </html> |
|  | <!--Проверка--> |
|  | <?php |
|  | $data = $\_POST; |
|  |  |
|  | if(isset($data['action\_login'])) { |
|  |  |
|  | $errors = array(); |
|  | $user = R::findOne('users', 'login = ?', array($data['login'])); |
|  |  |
|  | if($user) { |
|  | if(md5($data['password']) == $user->password) { |
|  | if (strtotime((date("Y-m-d"))) < strtotime($user->data) || strtotime((date("Y-m-d"))) == strtotime($user->data)) { |
|  | $\_SESSION['logged\_user'] = $user; |
|  | header('Location: index.php'); |
|  | } |
|  | else { |
|  | $errors[] = 'Извините, но срок действия пароля подошел к концу'; |
|  | } |
|  | } |
|  | else { |
|  | Ф |
|  | $errors[] = 'Введенный пароль не верный, попробйте еще раз'; |
|  | } |
|  |  |
|  | } |
|  | else { |
|  | $errors[] = 'Пользователь с таким логином не найден'; |
|  | } |
|  | //проверка на ошибки |
|  | if(!empty($errors)) { |
|  | echo '<div align = "center" style="color: red; ">' . array\_shift($errors). '</div><hr>'; |
|  | }} |
|  | ?> |

Форма выхода:

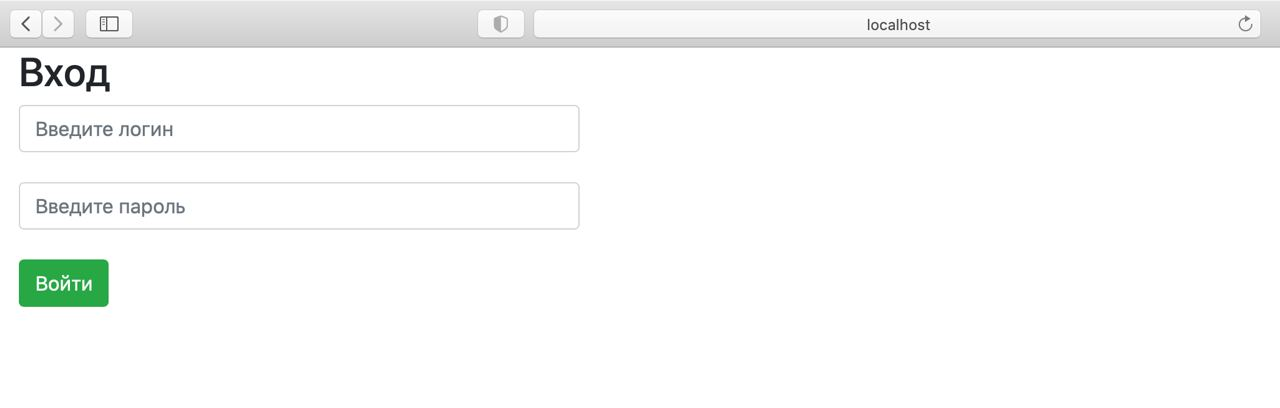
|  |
| --- |
| <!DOCTYPE html> |
|  | <html lang="ru"> |
|  | <head> |
|  | <title><?php echo $title; ?></title> |
|  | <link rel="stylesheet" type="text/css" href="css/bootstrap.min.css"> |
|  | <meta content="text/html; charset=utf-8"> |
|  | </head> |
|  | <body> |
|  | <?php |
|  | require "db.php"; |
|  | unset($\_SESSION['logged\_user']); |
|  | header('Location: index.php'); |
|  | ?> |
|  | </body> |
|  | </html> |

Страница заглушки:

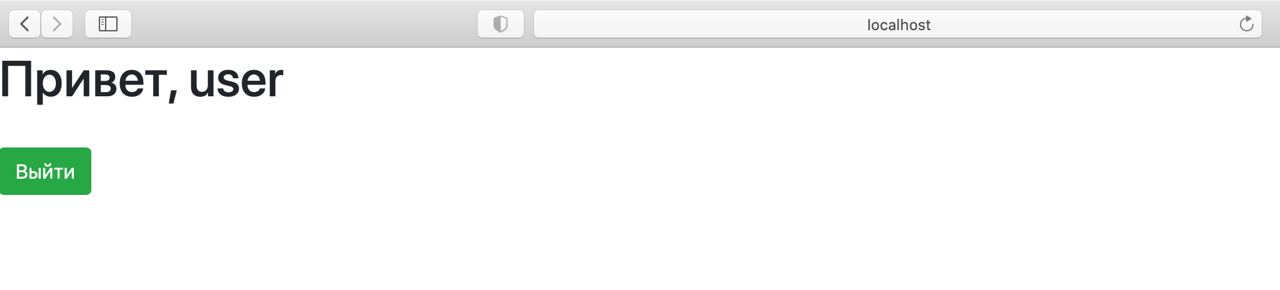
|  |
| --- |
| <?php |
|  | $title="Парольная аутентификация"; |
|  | require "db.php"; |
|  | ?> |
|  |  |
|  | <!DOCTYPE html> |
|  | <html lang="ru"> |
|  | <head> |
|  | <title><?php echo $title; ?></title> |
|  | <link rel="stylesheet" type="text/css" href="css/bootstrap.min.css"> |
|  | <meta content="text/html; charset=utf-8"> |
|  | </head> |
|  | <body> |
|  |  |
|  | <?php if(isset($\_SESSION['logged\_user'])) : ?> |
|  | <h1>Добро пожаловать, <?php echo $\_SESSION['logged\_user']->name; |
|  | ?></br></h1> |
|  | <br> |
|  | <a href="logout.php" class="btn btn-success">Выйти</a> |
|  | <?php else : ?> |
|  | <div class="container-fluid"> |
|  | <div class="row"> |
|  | <div class="col-md-4"> |
|  | <h2>Вход</h2> |
|  | <form action="login.php" method="post"> |
|  | <input type="text" class="form-control" name="login" id="login" placeholder="Введите логин" required><br> |
|  | <input type="password" class="form-control" name="password" id="pass" placeholder="Введите пароль" required><br> |
|  | <button class="btn btn-success" name="action\_login" type="submit">Войти</button> |
|  | </form> |
|  | <br> |
|  | </div> |
|  | </div> |
|  | </div> |
|  | </body> |
|  | </html> |
|  | <?php endif; ?> |

# **Демонстрация работы программы**

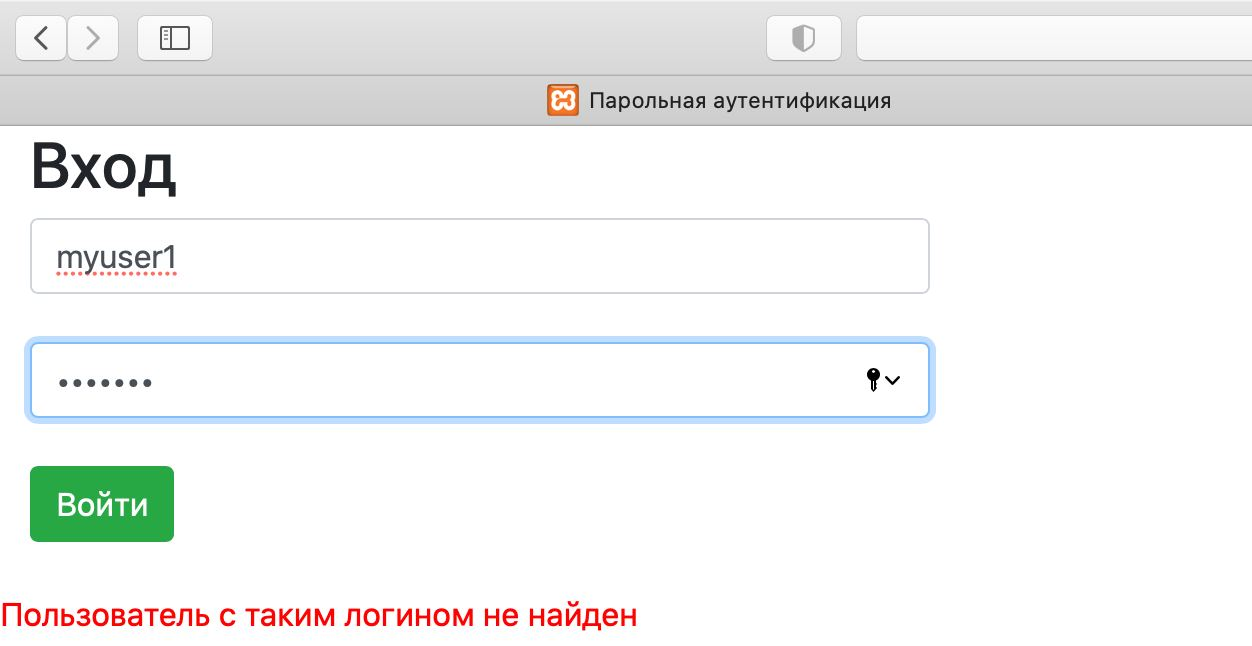
Открываем сайт, появляется окно входа:



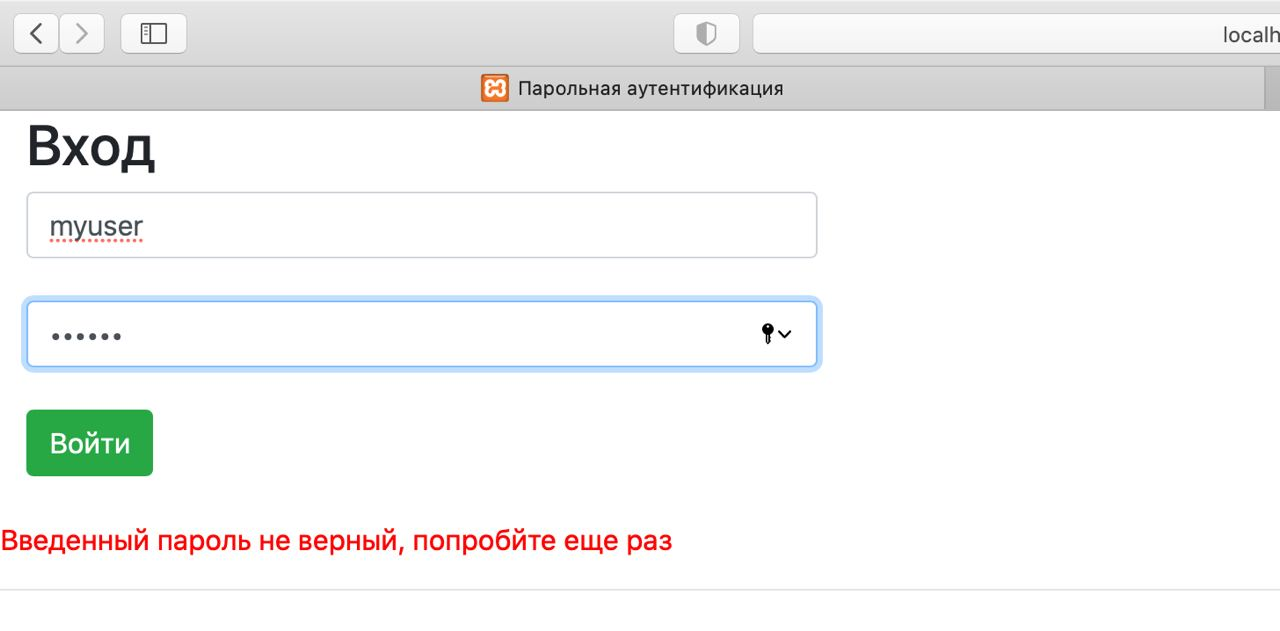
Вводим логин myuser и пароль user 123 и нажимаем войти:



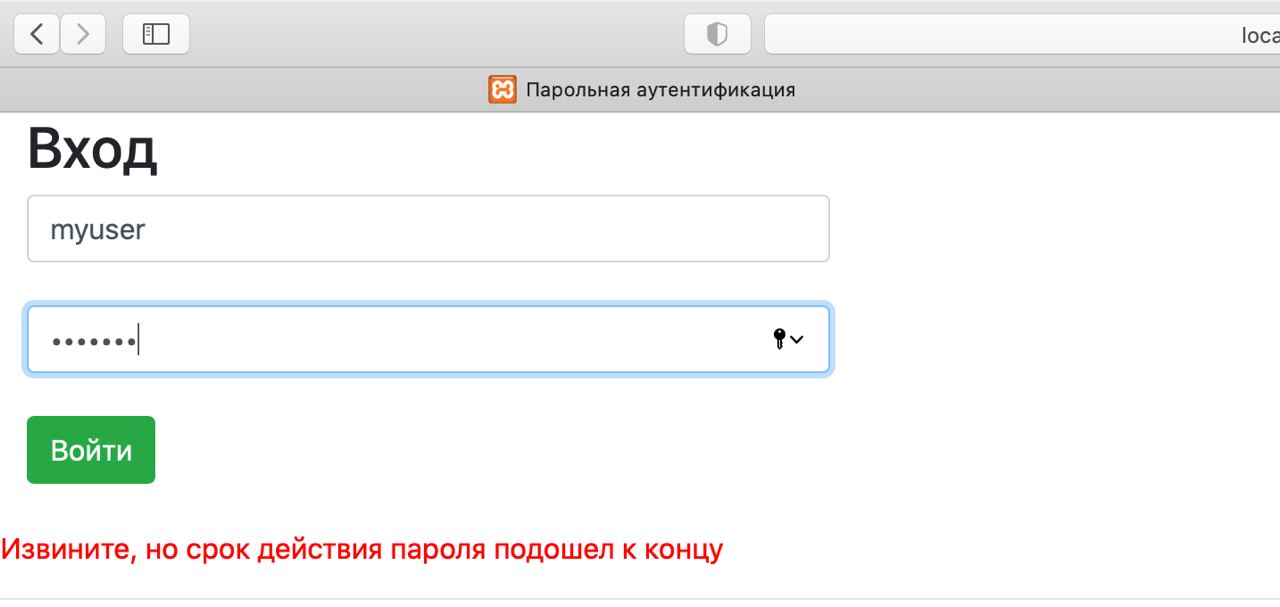
При вводе неправильного логина отобразится информационное окно:



При вводе неправильного пароля аналогично:



Так же предупреждение появится после истечения срока годности пароля (месяц):



# **Выводы**

В ходе лабораторной работы разработана система парольной аутентификации с хэшированием паролей в клиент-сервером приложении.