МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ

ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТЕЛЕКОММУНИКАЦИЙ ИМ. ПРОФ. М. А. БОНЧ-БРУЕВИЧА» (СПбГУТ)

Факультет инфокоммуникационных сетей и систем Кафедра программной инженерии и вычислительной техники

Курсовая работа

«Разработка web-приложения с системой авторизации и монетизированной системой доступа»

по дисциплине «Web-технологии»

Выполнил:

студент 3-го курса

группы ИКПИ-14

Хохлов Т. В.

Подпись\_\_\_\_\_\_\_\_

Принял:

Краева Е. В.

Подпись\_\_\_\_\_\_\_\_

«\_\_\_» \_\_\_\_\_\_\_2023

Санкт-Петербург 2023

Оглавление

[Введение 2](#_Toc151916692)

[Постановка задачи 3](#_Toc151916693)

[Реализация 5](#_Toc151916694)

[Заключение 18](#_Toc151916695)

[Механизм регистрации: 18](#_Toc151916696)

[Процесс входа на сайт: 18](#_Toc151916697)

[Безопасность: 18](#_Toc151916698)

[Управление сессиями и доступом: 18](#_Toc151916699)

[Интеграция с MySQL: 19](#_Toc151916700)

[Тестирование и отладка: 19](#_Toc151916701)

[Список использованных источников 20](#_Toc151916702)

[Приложения 21](#_Toc151916707)

[Код программы: 21](#_Toc151916708)

[1. login.php 21](#_Toc151916709)

[2. register.php 23](#_Toc151916710)

[3. logout.php 25](#_Toc151916711)

[4. session.php 25](#_Toc151916712)

[5. config.php 25](#_Toc151916713)

[6. yourAccount.php 25](#_Toc151916714)

[7. checkFunction.php 30](#_Toc151916715)

[8. labsoverview.php 31](#_Toc151916716)

[9. sberIntegration.php 35](#_Toc151916717)

[10. helpFunctions.php 36](#_Toc151916718)

[Файлы стилей (.css): 37](#_Toc151916719)

[1. fileMenu.css 37](#_Toc151916720)

[2. register.css 37](#_Toc151916721)

[3. styles.css 39](#_Toc151916722)

[База данных зарегистрированных пользователей: 47](#_Toc151916723)

Введение

В контексте постоянного роста важности виртуального пространства для обмена информацией и услуг, реализация эффективной системы управления доступом и монетизации является неотъемлемой частью разработки веб-приложений. Данная курсовая работа ориентирована на анализ и реализацию системы авторизации с дополнительным функционалом, написанного с использованием HTML и PHP, а также использованием базы данных SQL для хранения и управления данными. Цель работы заключается в создании и усовершенствовании системы авторизации с интегрированными механизмами управления доступом и монетизации. Основные задачи включают в себя:

1. Разработка и внедрение системы авторизации с использованием HTML и PHP: Создание безопасной системы для аутентификации пользователей и их регистрации, обеспечивая удобство использования.
2. Управление доступом через базу данных SQL: Определение уровней доступа для различных категорий пользователей, внедрение механизмов контроля доступа к различным разделам приложения.
3. Реализация механизма загрузки файлов: Интеграция функционала для администратора по загрузке файлов, с последующим управлением доступа для зарегистрированных пользователей.
4. Внедрение функционала "Pay access" с использованием PHP: Реализация механизма монетизации, включая переадресацию на платежный эквайринг и обновление статуса пользователя после успешной оплаты.
5. Интеграция сбербанк эквайринга с использованием PHP и SQL: Внедрение проверки оплаты через Сбербанк эквайринг, обеспечивая безопасность и надежность процесса монетизации.

С использованием HTML, PHP и базы данных SQL, данная курсовая работа представляет собой пример реализации современных технологий для разработки системы авторизации с дополнительным функционалом управления доступом и монетизации. Этот проект предоставляет практический опыт внедрения основных принципов безопасности и функциональности веб-приложения, а также демонстрирует эффективное использование языков программирования и базы данных для достижения заданных целей.

Постановка задачи

**Цель:**

Разработать безопасную и эффективную систему авторизации с использованием PHP и MySQL для обеспечения аутентификации пользователей и управления доступом к веб-ресурсу.

**Задачи:**

Реализация механизма регистрации пользователей:

* Создать страницу регистрации с формой для ввода данных (логин, email, пароль, верификация пароля).
* Провести проверку на уникальность логина и email в базе данных перед регистрацией.
* Хранить данные о зарегистрированных пользователях в таблице MySQL.

Разработка процесса входа на сайт:

* Разработать страницу для входа с формой для ввода email и пароля.
* Проверять соответствие введенных данных с данными в базе данных для аутентификации.
* Создать сессию для авторизованных пользователей для последующего доступа к странице.

Обеспечение безопасности:

* Применить безопасные методы хранения паролей, такие как хэширование с использованием bcrypt.
* Использовать подготовленные запросы (Prepared Statements) для предотвращения SQL-инъекций.

Управление сессиями и доступом:

* Создать механизм управления сессиями для аутентифицированных пользователей.
* Ограничить доступ к защищенным страницам только для авторизованных пользователей.

Интеграция с MySQL:

* Установить и поддерживать соединение с базой данных MySQL для хранения информации о пользователях.
* Реализовать механизмы запросов к базе данных для проверки учетных данных пользователя при входе и регистрации.

Тестирование и отладка:

* Провести тестирование функциональности системы авторизации, включая регистрацию и вход под различными условиями.
* Отладить возможные ошибки и обеспечить корректное функционирование системы.

Эти задачи представляют основные направления работы для разработки системы авторизации с использованием PHP и MySQL и помогут обеспечить эффективное и безопасное функционирование данного веб-приложения.

Реализация

Данное web-приложение состоит из следующих страниц:

* welcome.php – главная страница.
* register.php – страница с формой регистрации.
* login.php – страница с формой авторизации.
* logout.php – вспомогательный файл, который заканчивает сессию и переадресует пользователя на страницу login.php.
* session.php – вспомогательный файл, который проверяет была ли авторизация осуществлена.
* config.php – вспомогательный файл, который производит подключение базы данных зарегистрированных пользователей.
* yourAccount.php – страница редактирования аккаунтов пользователей.
* checkFunction.php – вспомогательный файл для страницы yourAccount.php, который проверяет существует ли пользователь с данным логином или электронным адресом при попытке их смены.
* labsoverview.php – страница электронного хранилища, с монетизированной системой доступа.
* helpFunctions.php – вспомогательный файл для страницы labsoverview.php.
* sberIntegration.php – вспомогательный файл для реализации интеграции «Сбербанк Эквайринг».

Пользователю предоставляется ссылка на главную страницу сайта. При входе на главную страницу производится проверка пользователя на авторизацию:

<?php

// start the session

session\_start();

// Check if the user is not logged in, then redirect the user to login page

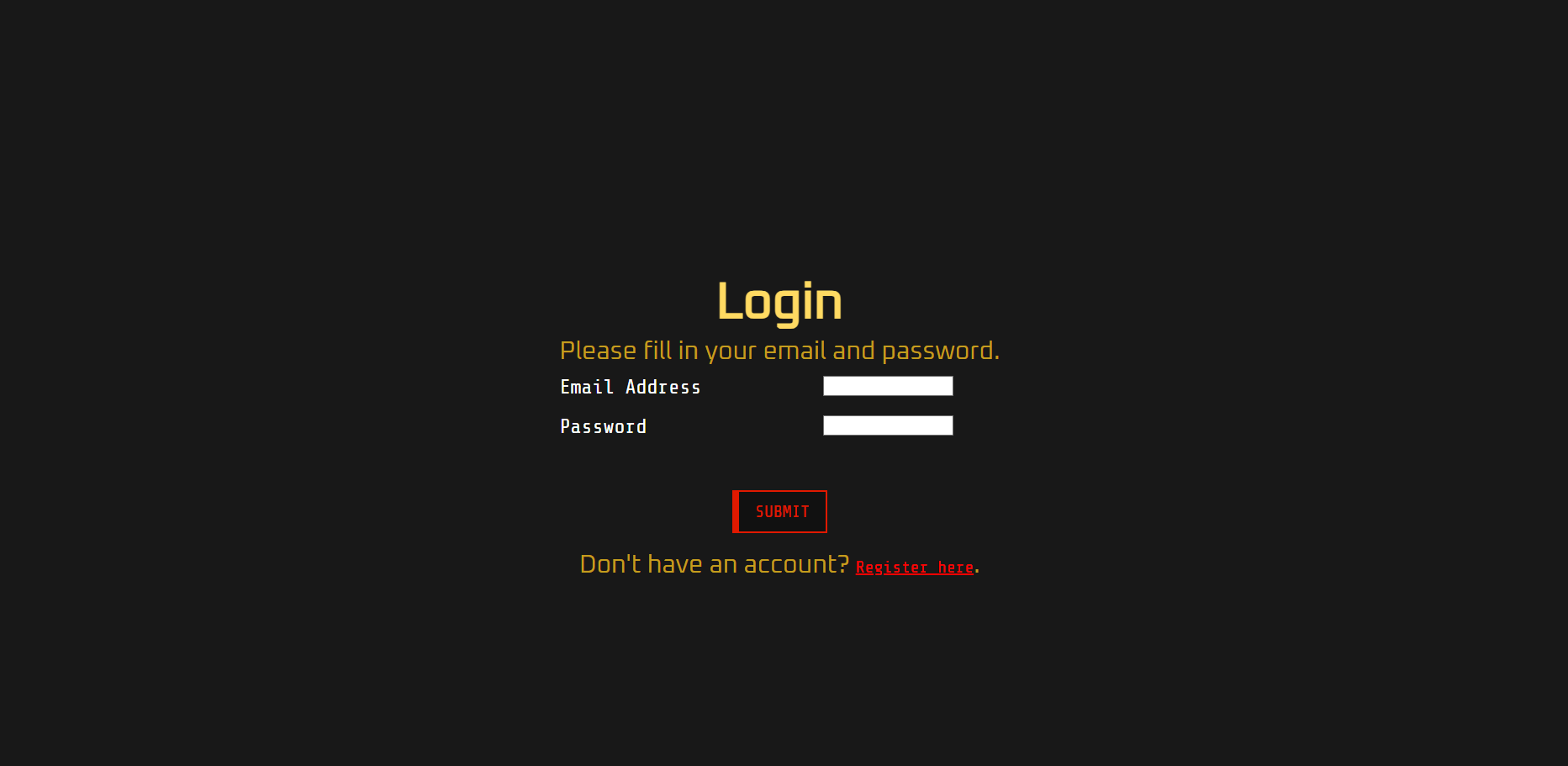
if ($\_SESSION["userLoggedIn"] !== true) {

    header("location: userstatus/login.php");

}

?>

Если пользователь не был авторизован, то его перенаправят на страницу авторизации – login.php, в противном случае ему отобразится главная страница сайта.



*Рисунок 1 – Попытка входа без авторизации.*

Для доступа к главной странице сайта, пользователю необходим зарегистрированный в базе данных Mysql аккаунт, необходимую информацию которого нужно ввести в форме на странице – login.php, а именно адрес электронной почты и пароль.

Если пользователь попытается обойти систему авторизации, то форма дополнится окном с описанием соответствующей ошибки.

*Приложение кода.*

*Часть программы, отвечающая за проверку правильности ввода login.php.*

if ($\_SERVER["REQUEST\_METHOD"] == "POST" && isset($\_POST['submit'])) {

    $email = trim($\_POST['email']);

    $entered\_password = trim($\_POST['password']);

    // validate if email is empty

    if (empty($email)) {

        $error .= '<div class="hintBar">Please enter email.</div>';

    }

    // validate if password is empty

    if (empty($entered\_password)) {

        $error .= '<div class="hintBar">Please enter your password.</div>';

    }

    if (empty($error)) {

        $query = "SELECT id, name, password, email, fullname, description, status FROM users WHERE email = (?)";

        if($statement = $db->prepare($query)) {

            $statement->bind\_param('s', $email);

            $statement->execute();

            $statement->bind\_result($idDB, $nameDB, $passwordDB, $emailDB, $fullnameDB, $descriptionDB, $status);

            $statement->fetch();

            // echo $nameDB;

            if ($emailDB==$email) {

                if (password\_verify($entered\_password, $passwordDB)) { ...

                } else {

                    $error .= '<div class="hintBar">The password is not valid.</div>';

                }

            } else {

                $error .= '<div class="hintBar">No user exist with that email address.</div>';

            }

        }

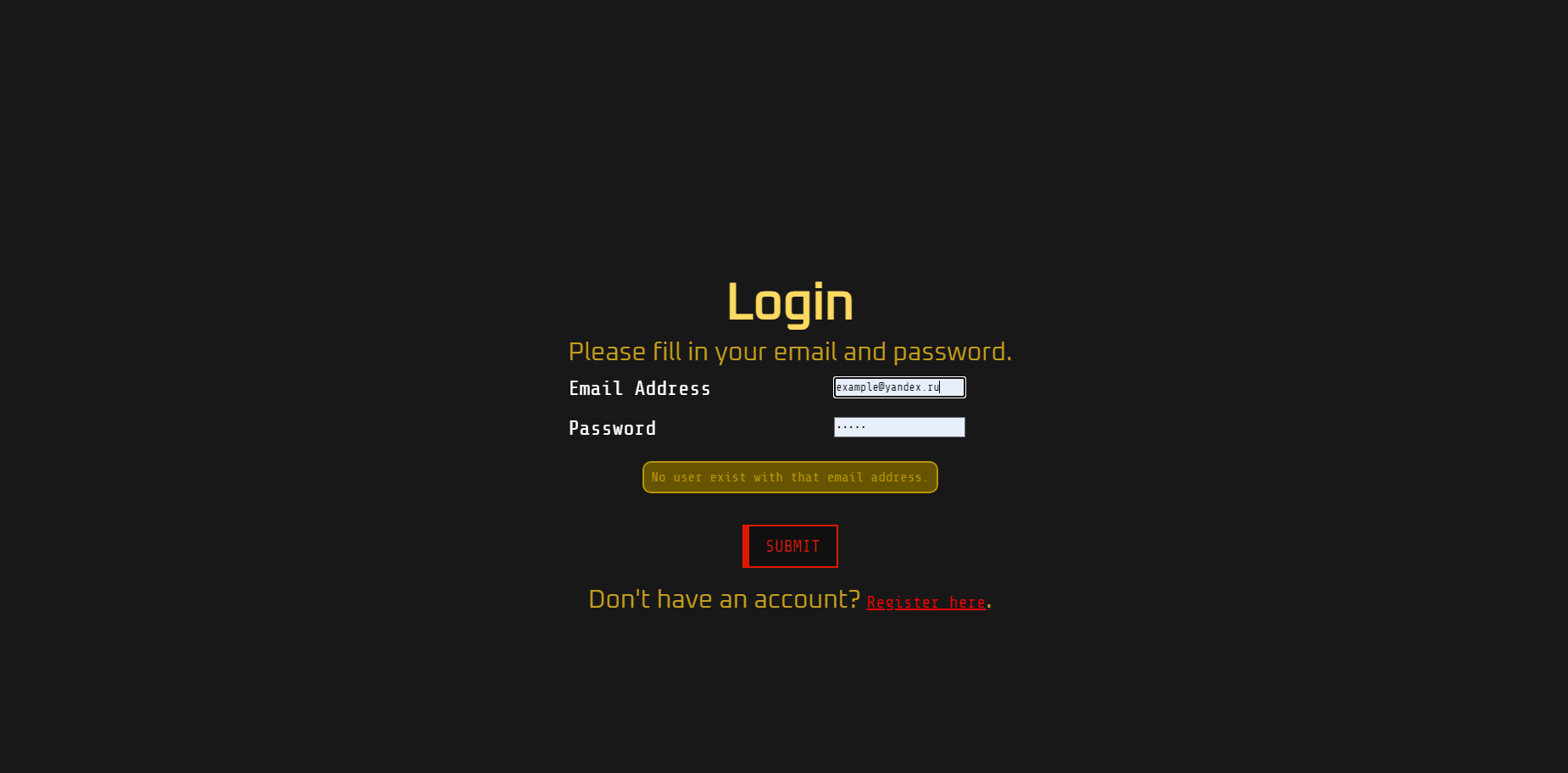
        $statement->close();

    }

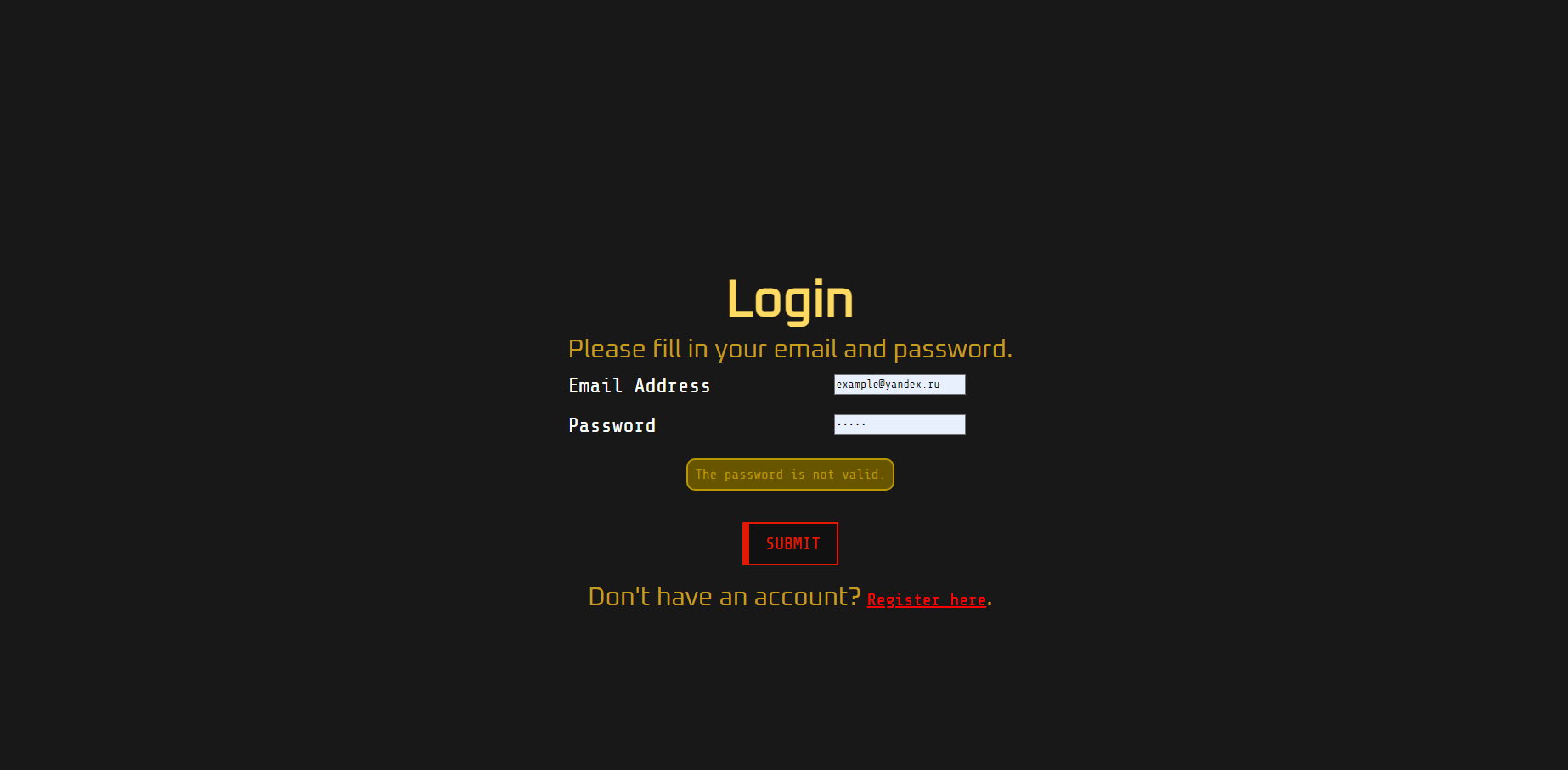
    // Close connection

    mysqli\_close($db);

}

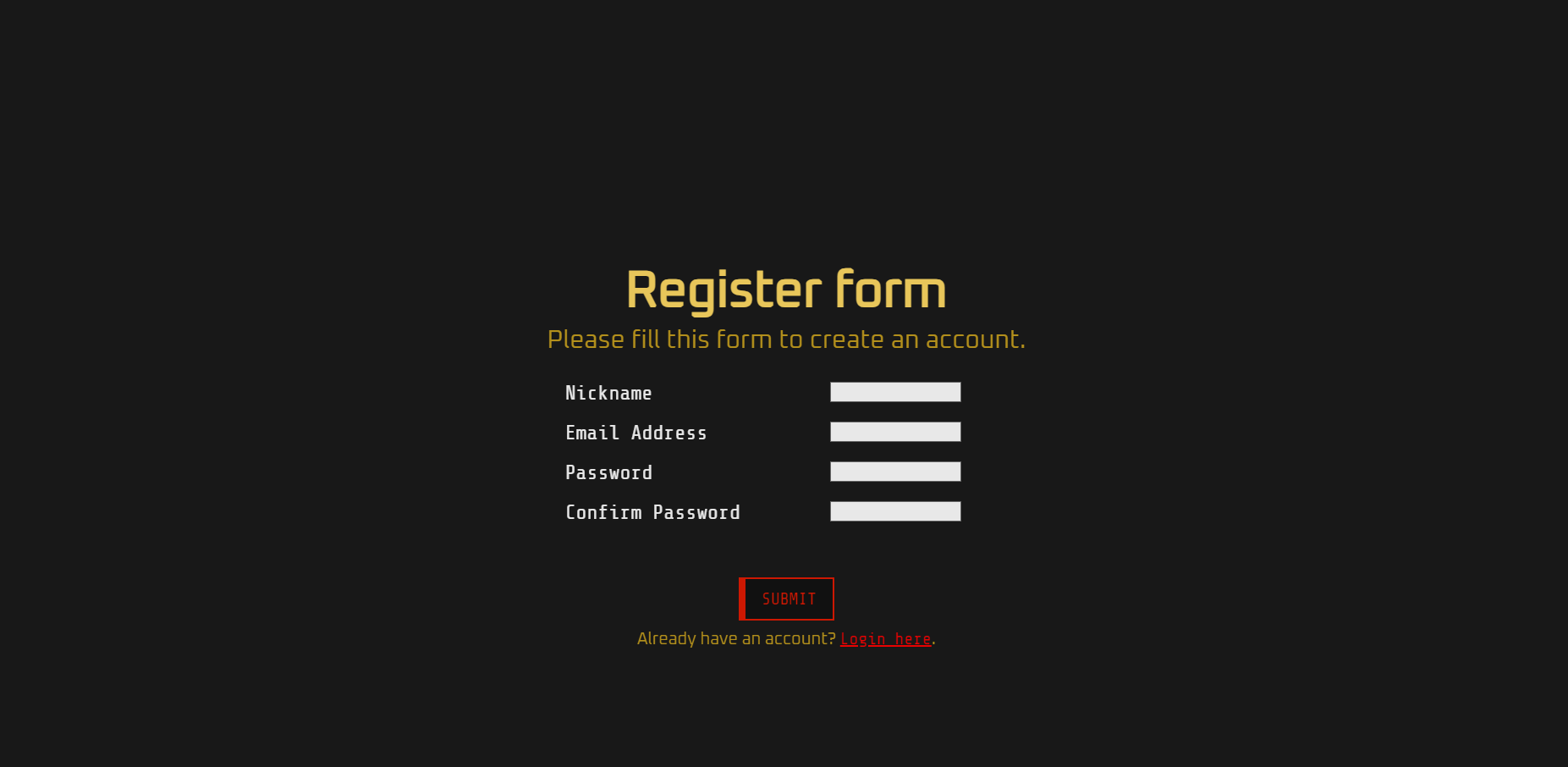


*Рисунок 2 – Попытка входа в незарегистрированный аккаунт.*



*Рисунок 3 – Попытка входа с вводом неверного пароля.*

Для регистрации нового аккаунта, пользователь должен нажать на ссылку «Register here.», после чего он будет перенаправлен на страницу с формой регистрации.



*Рисунок 4 – Форма регистрации.*

После перехода на страницу регистрации, пользователь должен указать необходимую информацию для создания аккаунта – Nickname (логин пользователя), Email address (Адрес электронной почты), Password (Пароль для последующей авторизации на сайте), Confirm password (Подтверждение пароля для авторизации). Если пользователь не укажет какую-либо указанную информацию или укажет её неверно, то форма дополнится окном с описанием соответствующей ошибки.

После успешной регистрации, данные пользователя будут занесены в базу данных и через пять секунд пользователь будет перенаправлен на страницу авторизации – login.php.

*Приложение кода.*

*Часть программы, отвечающая за проверку правильности ввода register.php.*

if ($\_SERVER["REQUEST\_METHOD"] == "POST" && isset($\_POST['submit'])) {

    $nickname = trim($\_POST['name']);

    $email = trim($\_POST['email']);

    $entered\_password = trim($\_POST['password']);

    $confirm\_password = trim($\_POST["confirm\_password"]);

    $password\_hash = password\_hash($entered\_password, PASSWORD\_BCRYPT);

    if($statement = $db->prepare("SELECT \* FROM users WHERE email = ? OR name = ?")) {

        $error = '';

        $statement->bind\_param('ss', $email, $nickname);

        $statement->execute();

        // Store the result so we can check if the account exists in the database.

        $statement->store\_result();

        if ($statement->num\_rows > 0) {

            $error .= '<div class="errorBar">This user is already exists!</div>';

        } else {

            // Validate password

            if (strlen($entered\_password ) < 6) {

                $error .= '<div class="hintBar">Password must have atleast 6 characters.</div>';

            }

            // Validate confirm password

            if (empty($confirm\_password)) {

                $error .= '<div class="hintBar">Please confirm a password.</div>';

            } else {

                if (empty($error) && ($entered\_password != $confirm\_password)) {

                    $error .= '<div class="errorBar">Password did not match.</div>';

                }

            }

            if (empty($error)) {

                } else {

                    $error .= '<div class="errorBar">Something went wrong!</div>';

                }

            }

        }

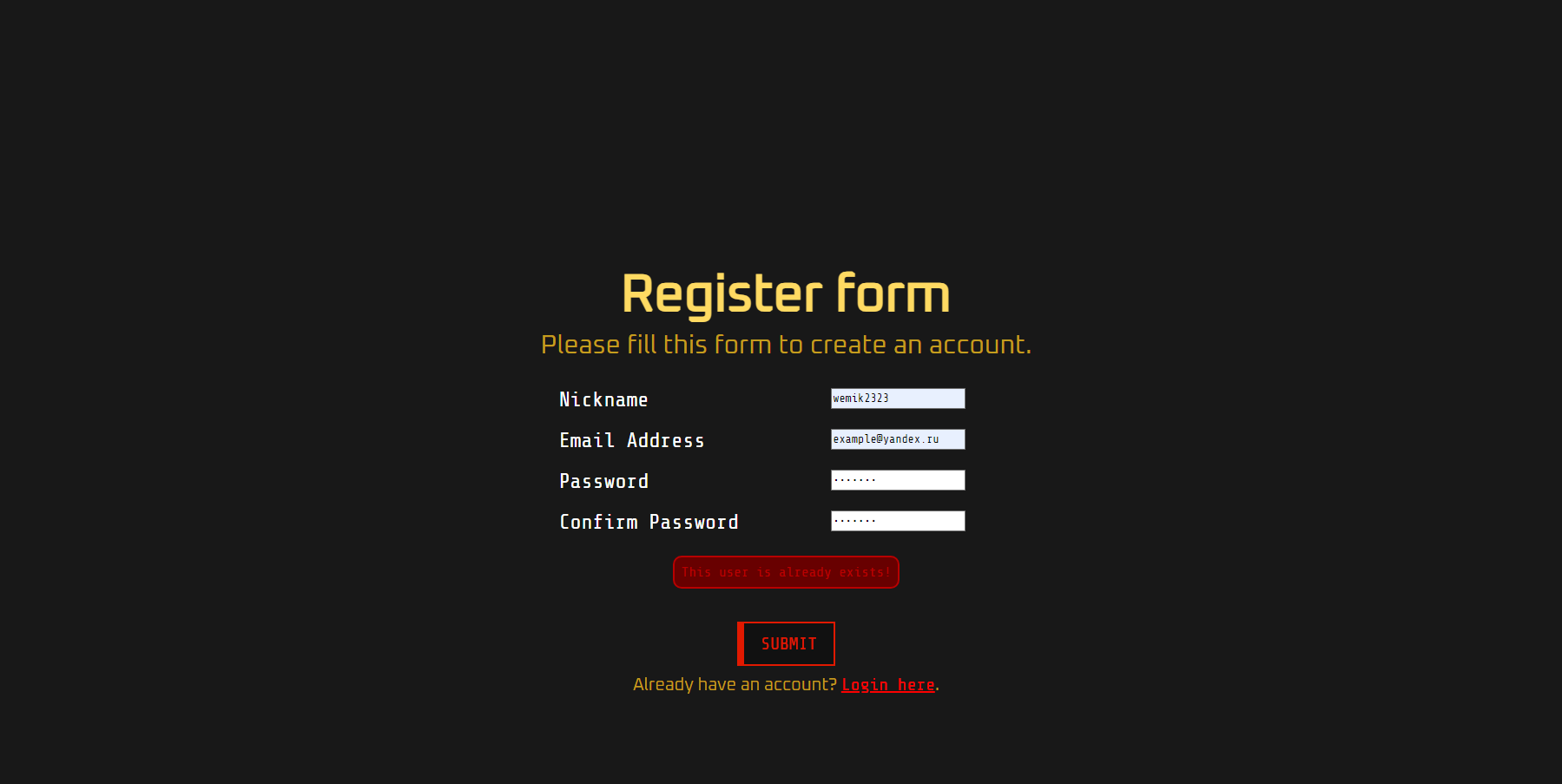
        $statement->close();

    }

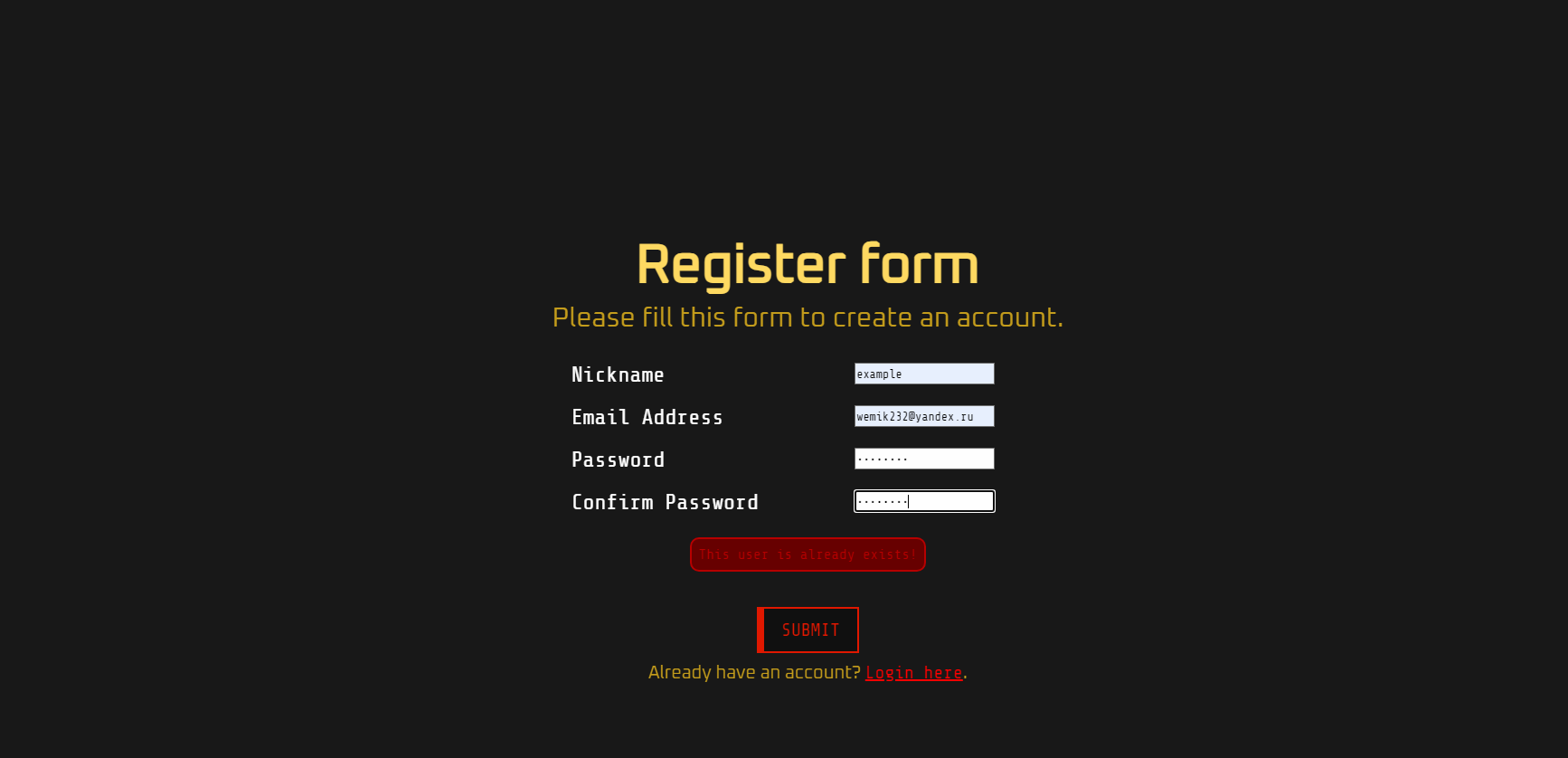
    // Close DB connection

    mysqli\_close($db);

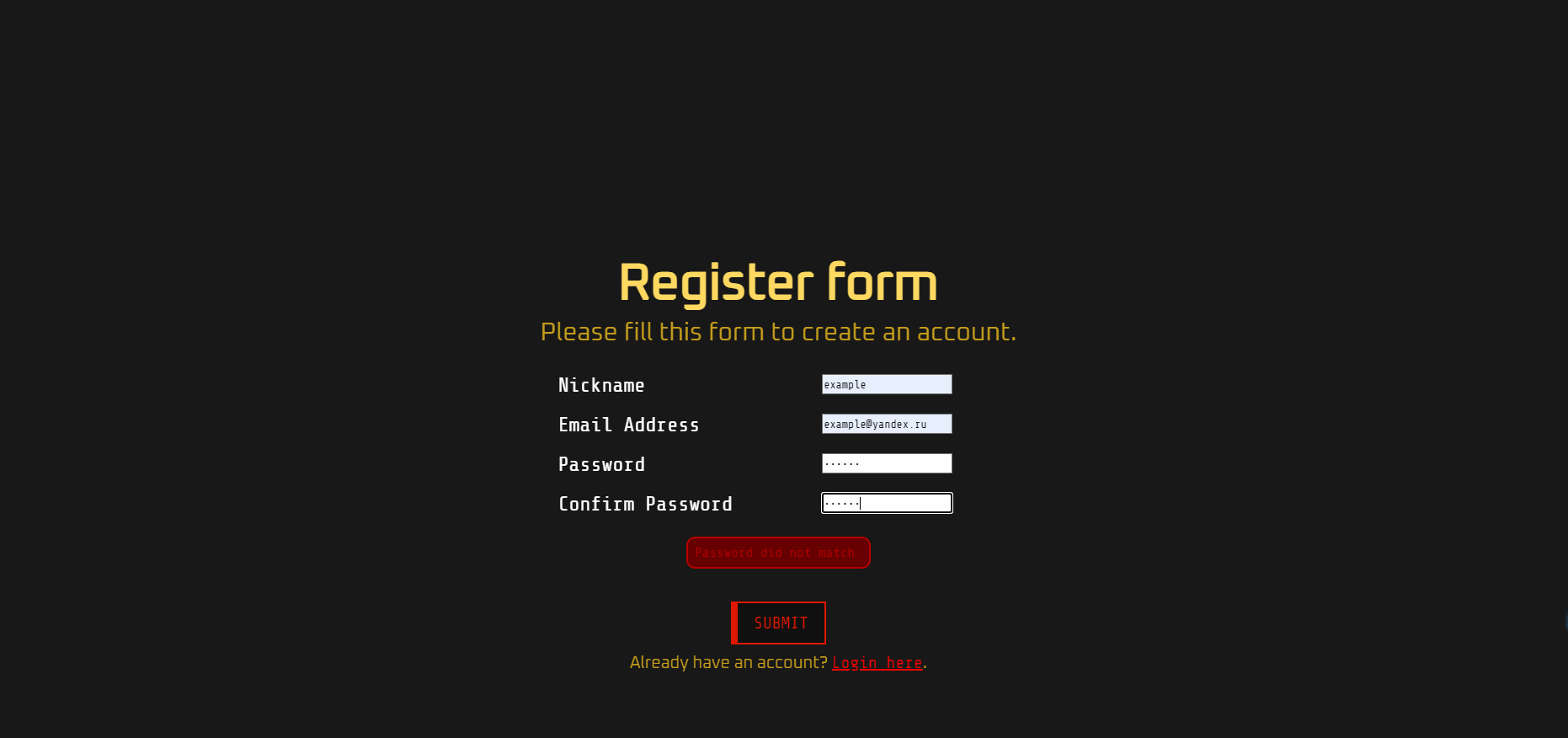
}



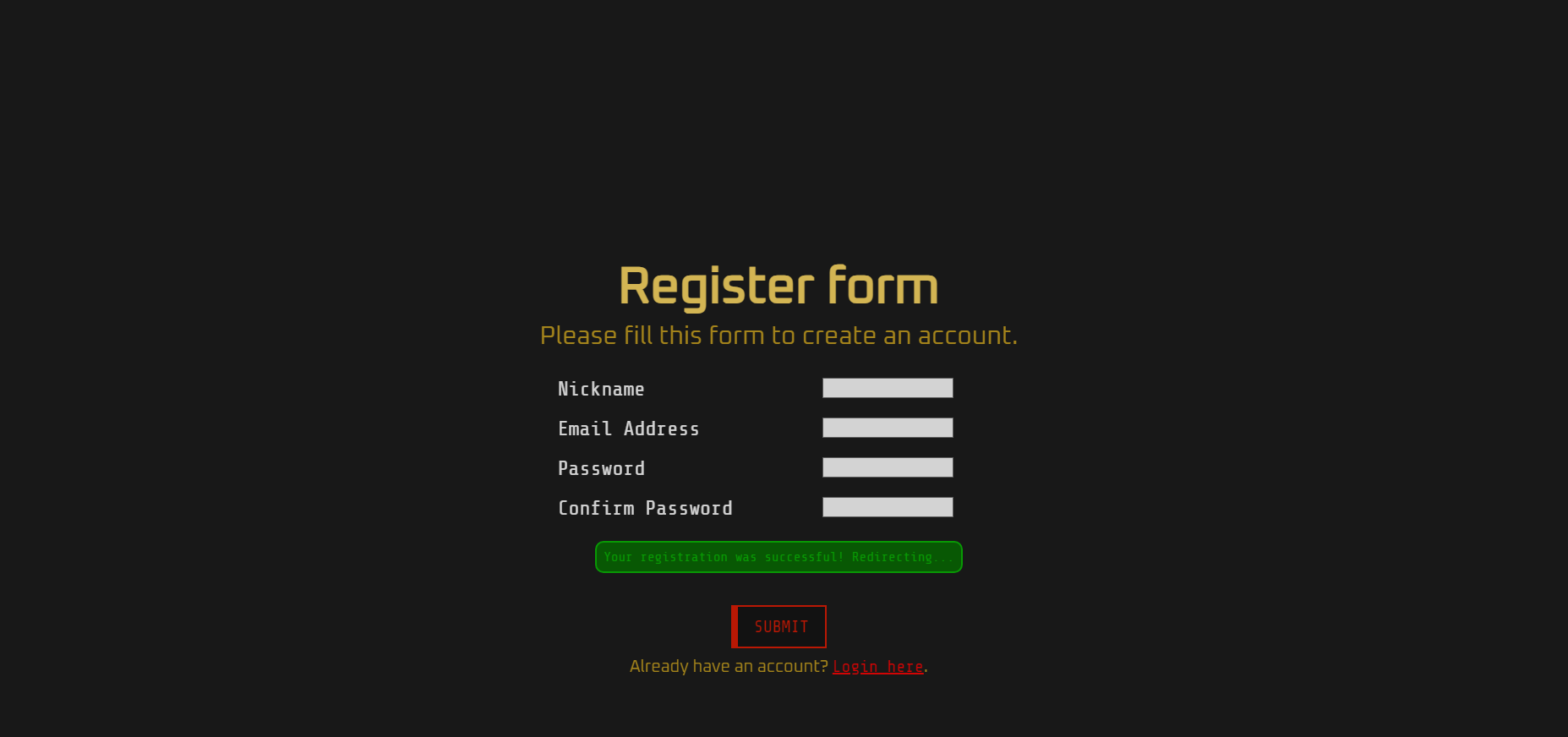
*Рисунок 5 – Попытка создания нового пользователя с занятым логином.*



*Рисунок 5 – Попытка создания нового пользователя с занятой почтой.*



*Рисунок 6 – Несовпадение паролей при регистрации.*



*Рисунок 7 – Пример успешной регистрации.*

После успешной авторизации на сайт, пользователь будет перенаправлен на главную страницу, на которой будет изображен его логин и текущее время:



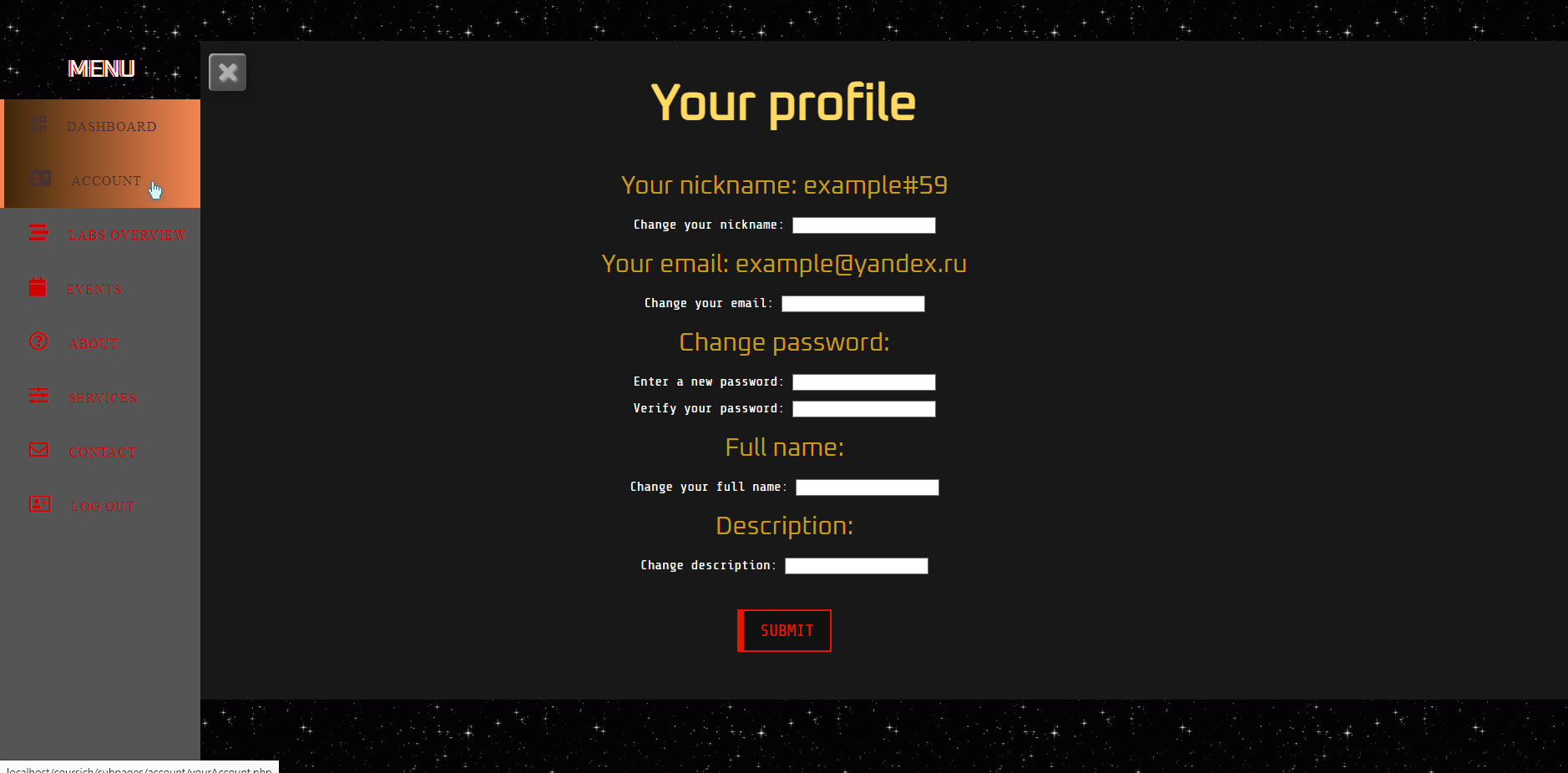
*Рисунок 8 – Главная страница.*

Так же главная страница имеет свою боковую панель (sidebar), который выдвигается после нажатия соответствующей кнопки. С помощью этой панели и производится переход на соответствующие страницы:



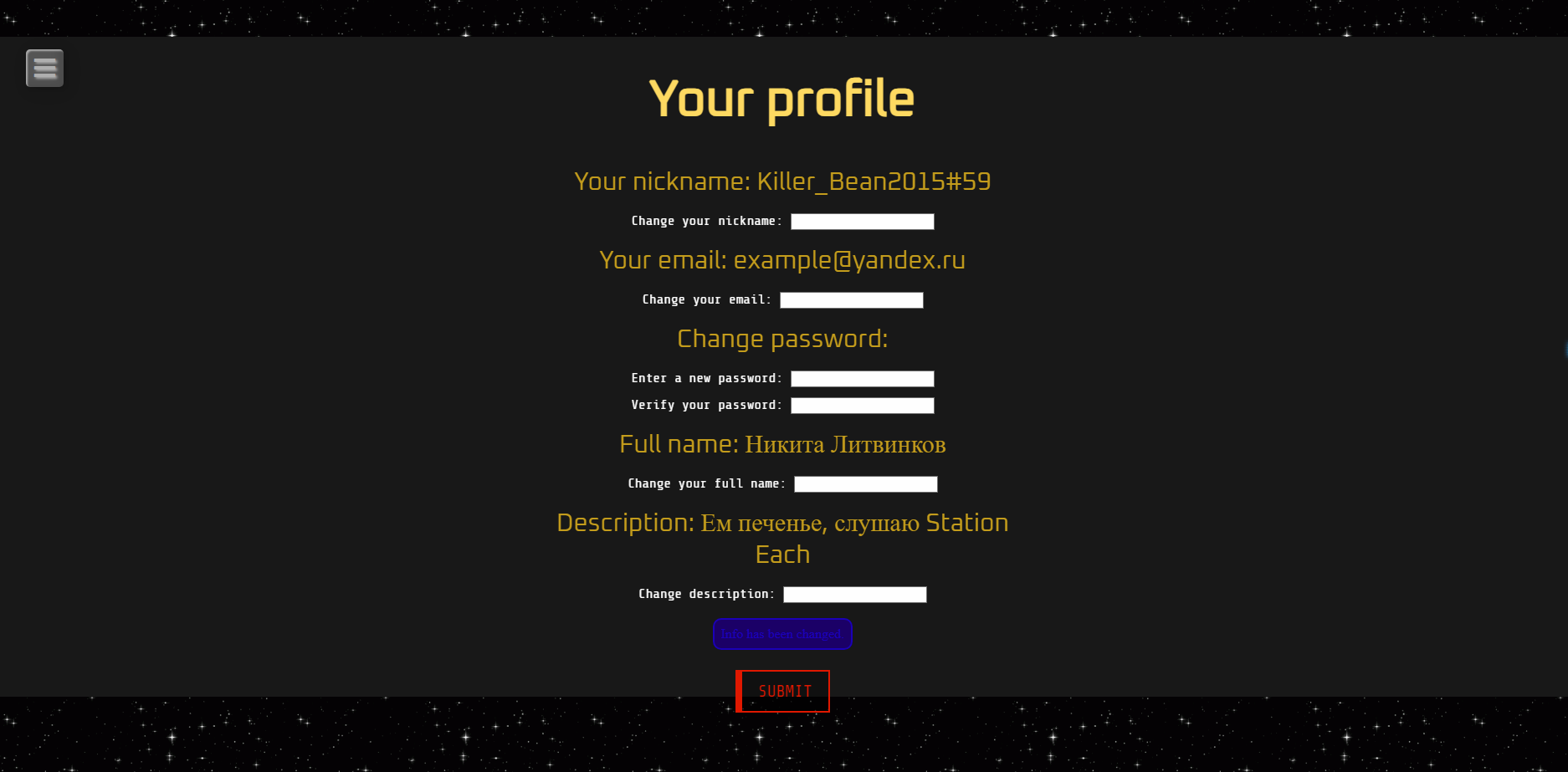
*Рисунок 9 – Боковая панель (sidebar).*

С помощью кнопки «Account» бокового меню, можно перейти на страницу редактирования своего аккаунта на сайте – yourAccount.php:



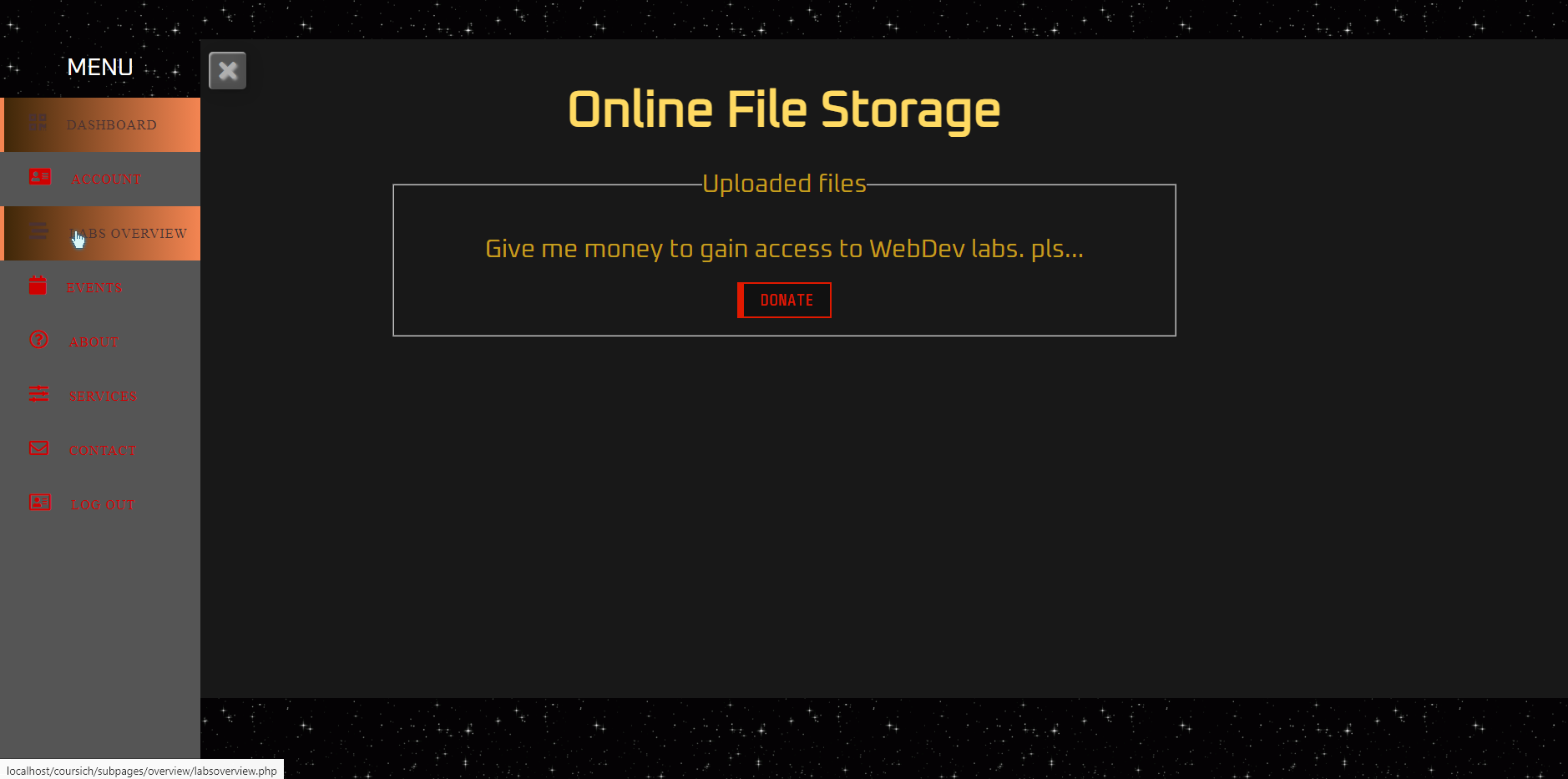
*Рисунок 10 – Страница редактирования аккаунта yourAccount.php.*

На данной странице пользователь может дополнить информацию о себе указав своё полное имя (Full name) и добавив описание о себе (Description). Так же пользователь может изменить уже существующую информацию о себе, а именно свой логин (nickname), адрес электронной почты (email) и пароль (password).

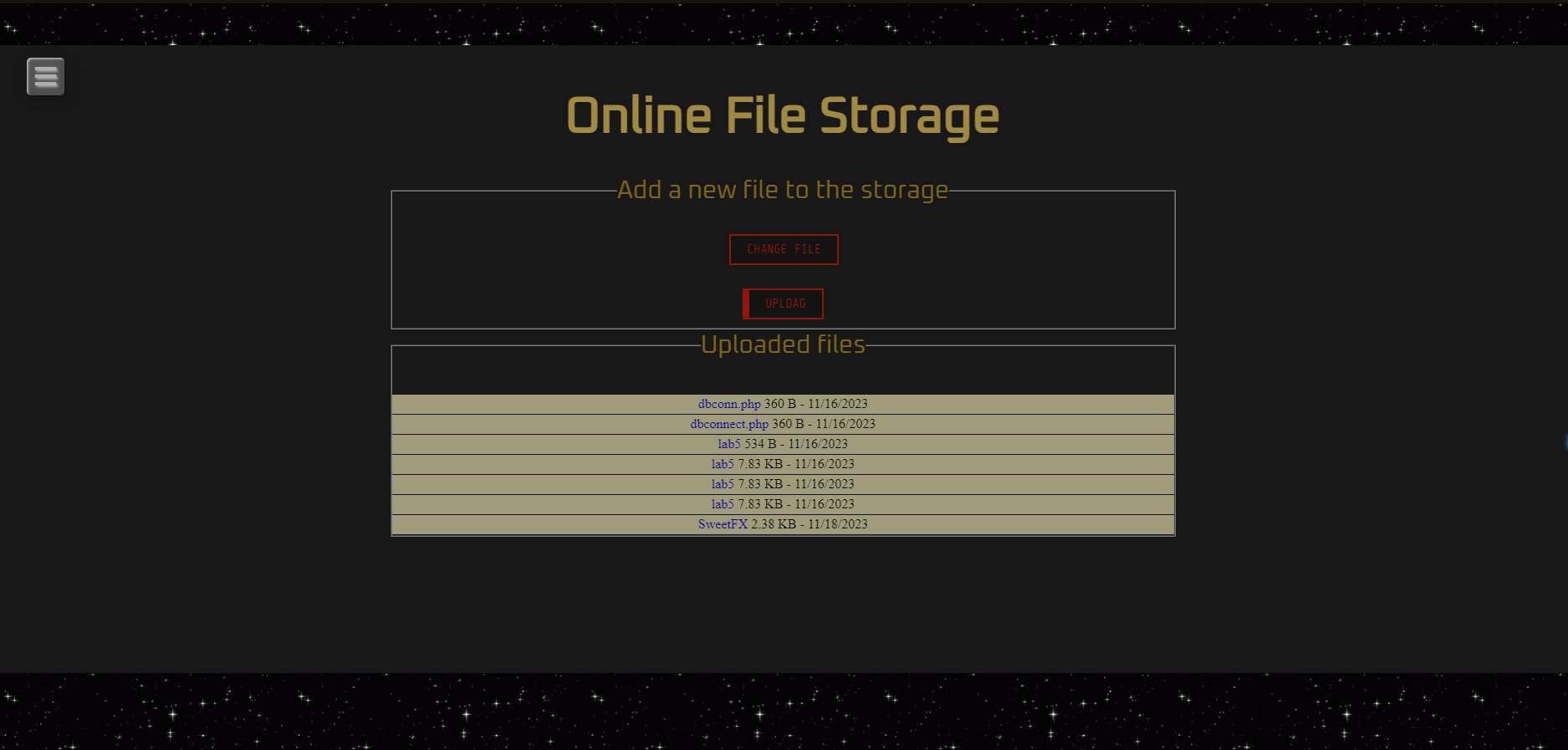


*Рисунок 11 – Пример редактирования аккаунта yourAccount.php.*

С помощью кнопки «LABS OVERVIEW» бокового меню, можно перейти на страницу монетизированного онлайн хранилища – labsoverview.php. На данной странице обычные пользователи могут получить доступ к определенным файлам, загруженным администратором, заплатив определённую сумму.

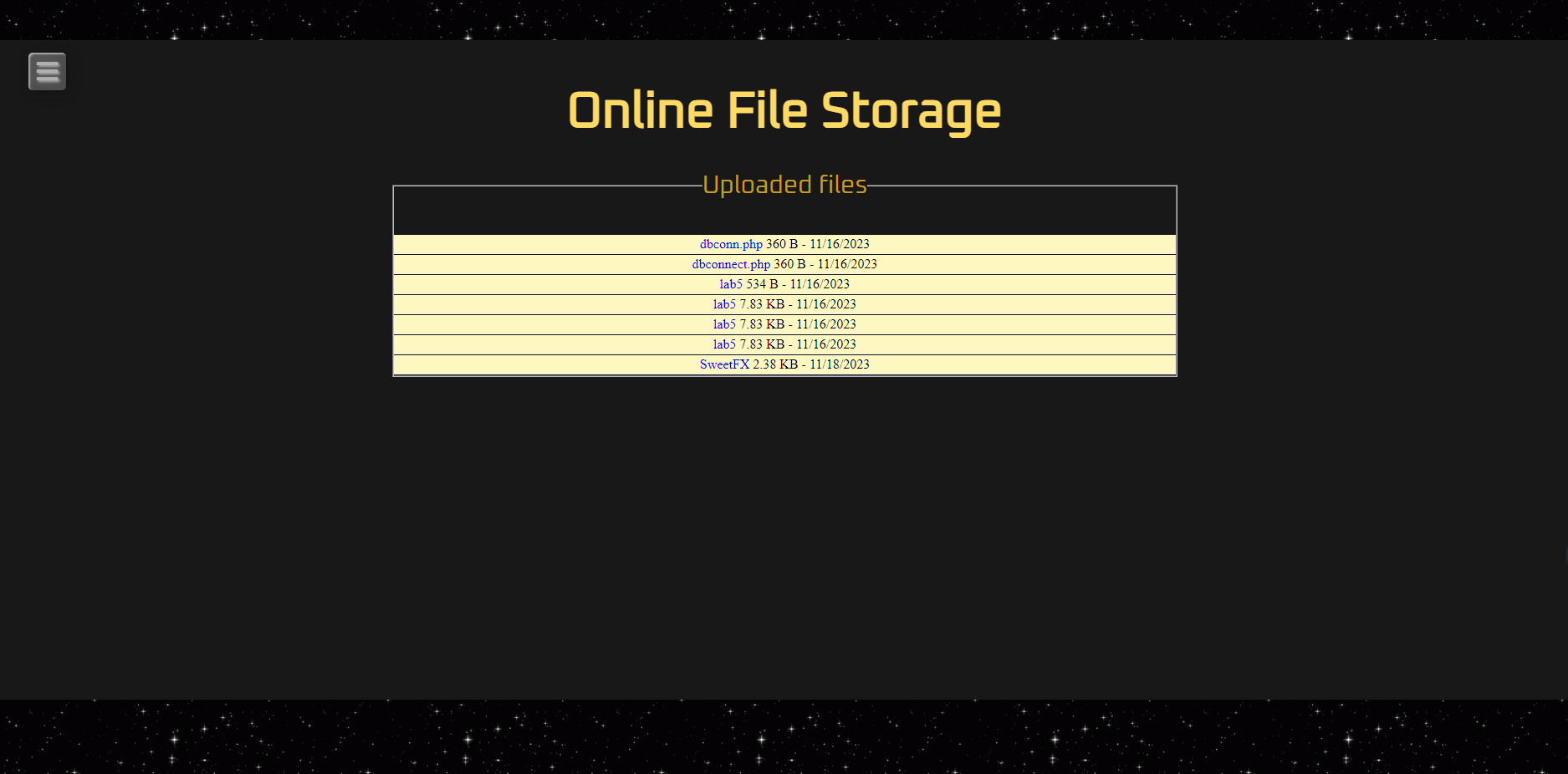


*Рисунок 12 – Страница labsoverview.php для пользователя статуса regular.*



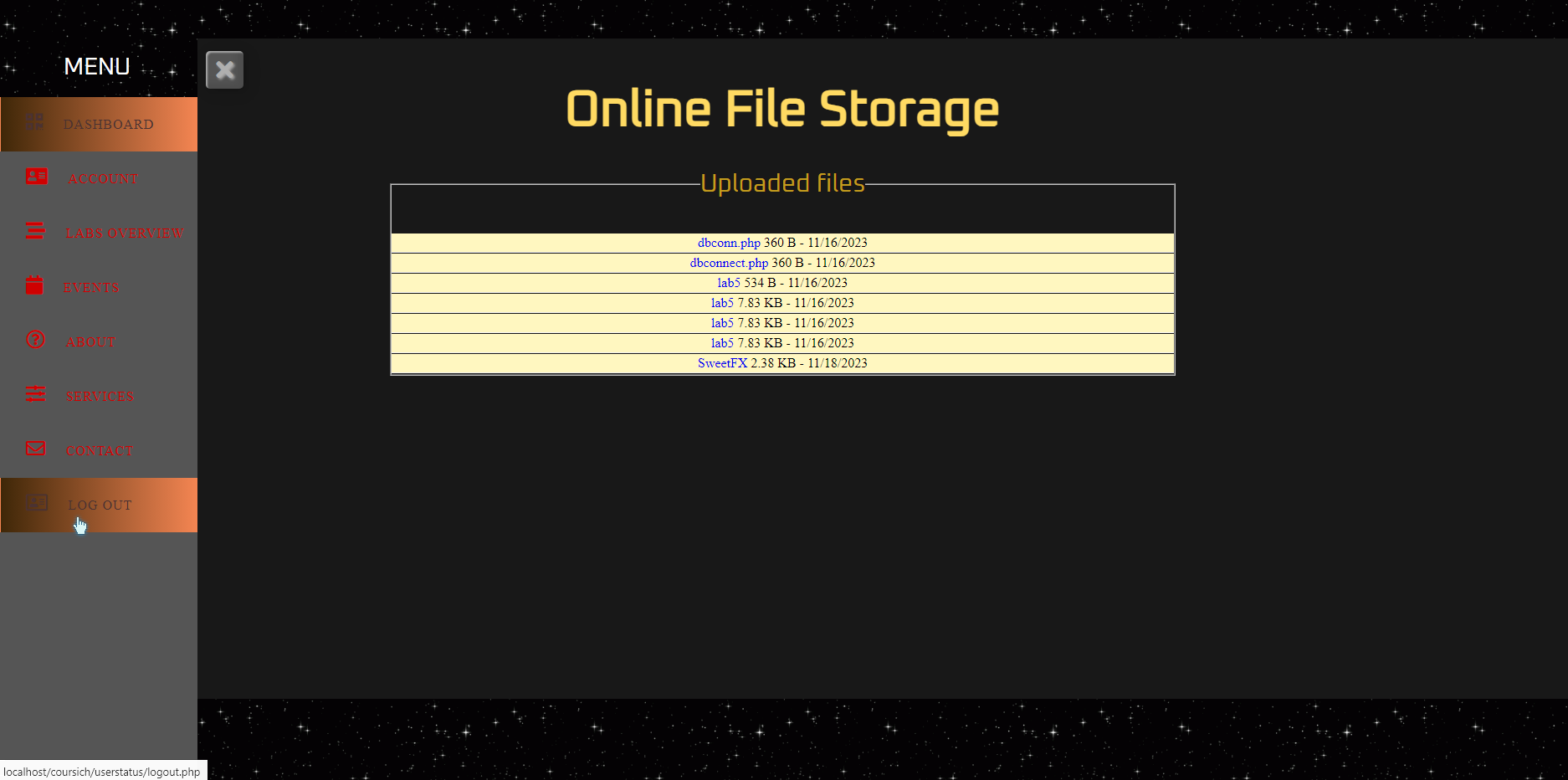
*Рисунок 13 – Страница labsoverview.php для пользователя статуса admin.*

После нажатия на кнопку «Donate», пользователя переадресует на страницу оплаты «Сбербанк Онлайн». Если оплата пройдёт успешно, то статус пользователя будет изменён на donated и его вернёт обратно, на страницу labsoverview.php, с доступом к электронному хранилищу и возможностью скачивания файлов с него.

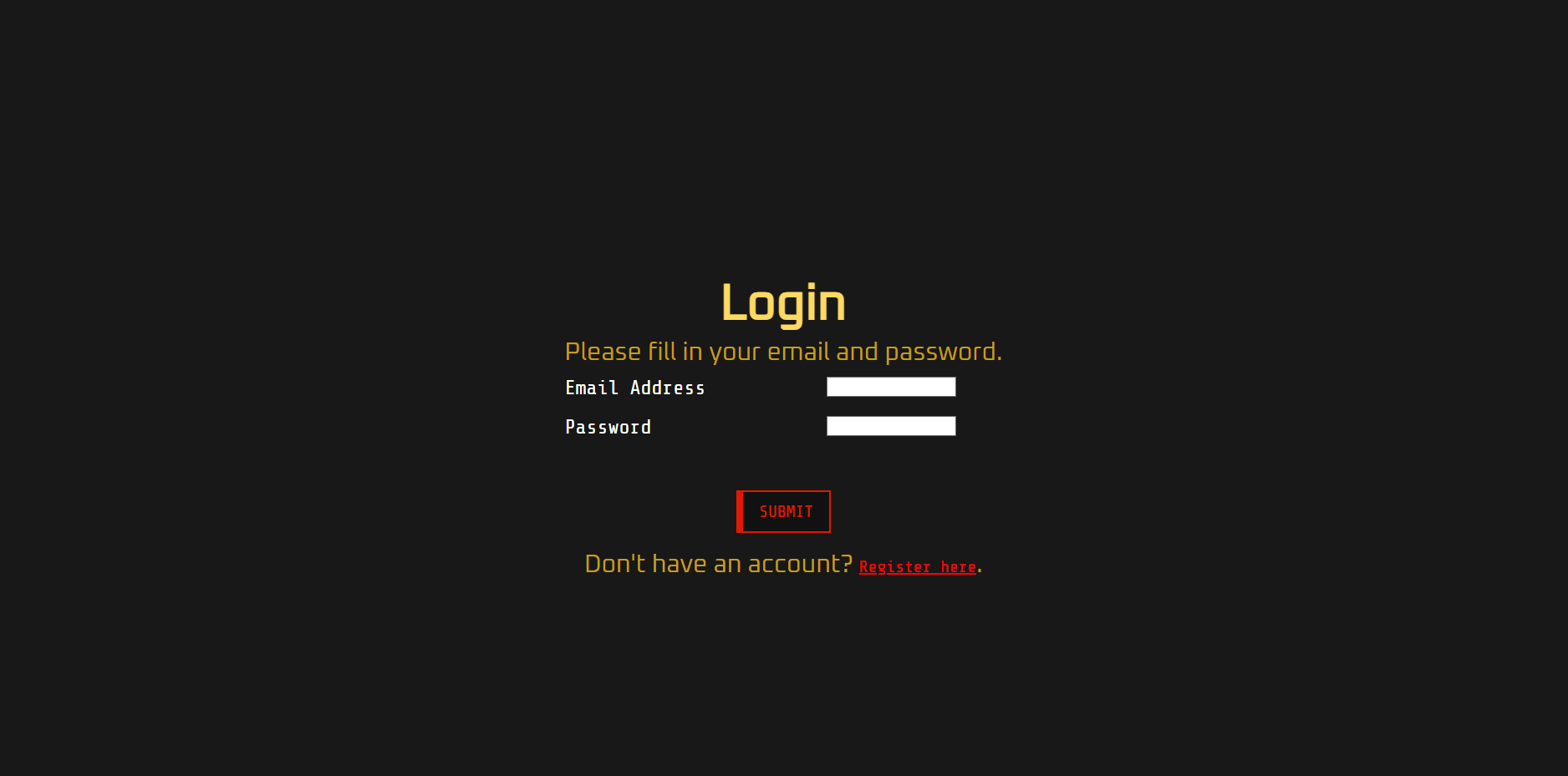


*Рисунок 14 – Страница labsoverview.php для пользователя статуса donated.*

После работы, в боковом меню пользователь может закончить текущую сессию и выйти со своего аккаунта на сайте нажав на кнопку «LOG OUT», после чего он будет перенаправлен на страницу авторизации – login.php:



*Рисунок 15 – Боковое меню кнопка «LOG OUT».*



*Рисунок 16 – Последствия нажатия на кнопку «LOG OUT».*

Заключение

В ходе разработки проекта были успешно достигнуты поставленные цели, а именно создание безопасного и эффективного механизма аутентификации пользователей для управления доступом к веб-ресурсу, реализован функционал онлайн хранилища файлов с механизмом платного доступа. Реализация включала в себя ключевые этапы, каждый из которых внес вклад в общую функциональность системы.

## **Механизм регистрации:**

* Создана интуитивно-понятная форма регистрации с учетом требований безопасности.
* Реализована проверка уникальности логина и email перед сохранением данных в базу данных MySQL.
* Использовано хэширование с bcrypt для безопасного хранения паролей.

## **Процесс входа на сайт:**

* Разработана страница входа с соответствующей формой для ввода email и пароля.
* Внедрена проверка аутентификации, обеспечивающая безопасный вход пользователей.
* Создан механизм управления сессиями для обеспечения безопасного и эффективного взаимодействия.

## **Безопасность:**

* Применены передовые методы безопасности, такие как хэширование с использованием bcrypt, для защиты конфиденциальной информации пользователей.
* Использованы подготовленные запросы для предотвращения возможных SQL-инъекций.

## **Управление сессиями и доступом:**

* Реализован механизм управления сессиями, обеспечивающий безопасное хранение данных авторизованных пользователей.
* Ограничен доступ к защищенным страницам только для авторизованных пользователей.

## **Интеграция с MySQL:**

* Установлено и поддерживается соединение с базой данных MySQL для хранения информации о пользователях.
* Реализованы безопасные механизмы запросов для проверки учетных данных при входе и регистрации.

## **Тестирование и отладка:**

* Проведено комплексное тестирование функциональности системы, включая сценарии успешной регистрации, входа и негативные сценарии.
* Выявленные ошибки были тщательно отлажены, обеспечивая корректное функционирование системы.

В итоге, реализованная система авторизации представляет собой надежное средство контроля доступа пользователей к веб-ресурсу. Результаты проекта могут быть использованы в качестве основы для дальнейшего развития и расширения функционала.

Список использованных источников

1. *PHP Manual*: [сайт]. - The PHP Group, (2001-2023). – Текст: электронный. – URL: <https://www.php.net/manual>.
2. *StackOverflow:* [сайт]. *-* Stack Exchange Inc., (2023)*. –* Текст: электронный. – URL: <https://stackoverflow.com>.
3. *Introduction to MySQL/PHP.* – Текст: электронный // PHP.org,(2023). – URL:[*https://php.org/introduction-to-mysql-php/#:~:text=MySQL%20is%20one%20of%20the,a%20robust%20web%20development%20platform*](https://php.org/introduction-to-mysql-php/#:~:text=MySQL%20is%20one%20of%20the,a%20robust%20web%20development%20platform)
4. *MySql:* [сайт]. - Oracle MYSQL, (2023). – Текст: электронный. – URL: <https://www.mysql.com>.

Приложения

Код программы:

1. login.php

<?php

require\_once "../subpages/config.php";

require\_once "../subpages/session.php";

global $error, $nameBD;

$error = '';

if ($\_SERVER["REQUEST\_METHOD"] == "POST" && isset($\_POST['submit'])) {

    $email = trim($\_POST['email']);

    $entered\_password = trim($\_POST['password']);

    // validate if email is empty

    if (empty($email)) {

        $error .= '<div class="hintBar">Please enter email.</div>';

    }

    // validate if password is empty

    if (empty($entered\_password)) {

        $error .= '<div class="hintBar">Please enter your password.</div>';

    }

    if (empty($error)) {

        $query = "SELECT id, name, password, email, fullname, description, status FROM users WHERE email = (?)";

        if($statement = $db->prepare($query)) {

            $statement->bind\_param('s', $email);

            $statement->execute();

            $statement->bind\_result($idDB, $nameDB, $passwordDB, $emailDB, $fullnameDB, $descriptionDB, $status);

            $statement->fetch();

            // echo $nameDB;

            if ($emailDB==$email) {

                if (password\_verify($entered\_password, $passwordDB)) {

                    $\_SESSION["userLoggedIn"] = true;

                    $\_SESSION["userId"] = $idDB;

                    $\_SESSION["userLogin"] = $nameDB;

                    $\_SESSION["userEmail"] = $emailDB;

                    $\_SESSION["userPassword"] = $passwordDB;

                    $\_SESSION["userFullname"] = $fullnameDB;

                    $\_SESSION["userDescription"] = $descriptionDB;

                    $\_SESSION["userStatus"] = $status;

                    // Redirect the user to welcome page

                    header("location: ../welcome.php");

                    exit;

                } else {

                    $error .= '<div class="hintBar">The password is not valid.</div>';

                }

            } else {

                $error .= '<div class="hintBar">No user exist with that email address.</div>';

            }

        }

        $statement->close();

    }

    // Close connection

    mysqli\_close($db);

}

?>

<!DOCTYPE html>

<html lang="en">

    <head>

        <meta charset="UTF-8">

        <title>Login</title>

        <link rel="preconnect" href="https://fonts.googleapis.com">

        <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

        <link href="https://fonts.googleapis.com/css2?family=Oxanium&family=Share+Tech+Mono&display=swap" rel="stylesheet">

        <link href="../styles/register.css" rel="stylesheet">

    </head>

    <body>

        <div class="container">

            <div class="row">

                <div class="col-md-12">

                    <h2>Login</h2>

                    <p>Please fill in your email and password.</p>

                    <form action="" method="post">

                        <div class="form-group">

                            <label>Email Address</label>

                            <input type="email" name="email" class="form-control" required />

                        </div>

                        <div class="form-group">

                            <label>Password</label>

                            <input type="password" name="password" class="form-control" required>

                        </div><br>

                        <?=$error;?>

                        <div class="button-to-verify">

                            <input type="submit" name="submit" class="GTFObutton" value="Submit" style="margin-top: 4%; margin-right: 0%; margin-bottom: 2%;">

                        </div>

                        <p>Don't have an account? <a href="register.php" style="color:red; font-size: 20px;">Register here</a>.</p>

                    </form>

                </div>

            </div>

        </div>

    </body>

</html>

1. register.php

<?php

require\_once "../subpages/config.php";

require\_once "../subpages/session.php";

global $error, $success;

$success = false;

if ($\_SERVER["REQUEST\_METHOD"] == "POST" && isset($\_POST['submit'])) {

    $nickname = trim($\_POST['name']);

    $email = trim($\_POST['email']);

    $entered\_password = trim($\_POST['password']);

    $confirm\_password = trim($\_POST["confirm\_password"]);

    $password\_hash = password\_hash($entered\_password, PASSWORD\_BCRYPT);

    if($statement = $db->prepare("SELECT \* FROM users WHERE email = ? OR name = ?")) {

        $error = '';

        $statement->bind\_param('ss', $email, $nickname);

        $statement->execute();

        // Store the result so we can check if the account exists in the database.

        $statement->store\_result();

        if ($statement->num\_rows > 0) {

            $error .= '<div class="errorBar">This user is already exists!</div>';

        } else {

            // Validate password

            if (strlen($entered\_password ) < 6) {

                $error .= '<div class="hintBar">Password must have atleast 6 characters.</div>';

            }

            // Validate confirm password

            if (empty($confirm\_password)) {

                $error .= '<div class="hintBar">Please confirm a password.</div>';

            } else {

                if (empty($error) && ($entered\_password != $confirm\_password)) {

                    $error .= '<div class="errorBar">Password did not match.</div>';

                }

            }

            if (empty($error) ) {

                $InserQuery = $db->prepare("INSERT INTO users (name, email, password, status) VALUES (?, ?, ?, 'regular');");

                $InserQuery->bind\_param("sss", $nickname, $email, $password\_hash);

                $result = $InserQuery->execute();

                if ($result) {

                    $error .= '<div class="successBar">Your registration was successful! Redirecting...</div>';

                    header("refresh:5; url=login.php");

                } else {

                    $error .= '<div class="errorBar">Something went wrong!</div>';

                }

            }

        }

        $statement->close();

    }

    // Close DB connection

    mysqli\_close($db);

}

?>

<!DOCTYPE html>

<html lang="en">

    <head>

        <meta charset="UTF-8">

        <title>Sign Up</title>

        <link rel="preconnect" href="https://fonts.googleapis.com">

        <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

        <link href="https://fonts.googleapis.com/css2?family=Oxanium&family=Share+Tech+Mono&display=swap" rel="stylesheet">

        <link href="../styles/register.css" rel="stylesheet">

    </head>

    <body>

        <div class="container">

            <div class="registration\_form">

                <h2>Register form</h2>

                <p style="margin-bottom: 20px;">Please fill this form to create an account.</p>

                <form action="" method="post">

                    <div class="form-group">

                        <label>Nickname</label>

                        <input type="text" name="name" class="form-control" required>

                    </div>

                    <div class="form-group">

                        <label>Email Address</label>

                        <input type="email" name="email" class="form-control" required />

                    </div>

                    <div class="form-group">

                        <label>Password</label>

                        <input type="password" name="password" class="form-control" required>

                    </div>

                    <div class="form-group">

                        <label>Confirm Password</label>

                        <input type="password" name="confirm\_password" class="form-control" required>

                    </div><br>

                    <?=$error;?>

                    <div class="button-to-verify" style="margin-top: 20px;">

                        <input type="submit" name="submit" class="GTFObutton" value="Submit">

                        <p style="font-size: 20px; margin-top: 10px;">Already have an account? <a href="login.php" style="color: red;">Login here</a>.</p>

                    </div>

                </form>

            </div>

        </div>

    </body>

</html>

1. logout.php

<?php

// Start the session

session\_start();

// Destroy the session.

if (session\_destroy()) {

    // redirect to the login page

    header("Location: login.php");

    exit;

}

?>

1. session.php

<?php

// Start the session

session\_start();

// if the user is already logged in then redirect user to welcome page

if (isset($\_SESSION["userid"]) && $\_SESSION["userid"] === true) {

    header("location: welcome.php");

    exit;

}

?>

1. config.php

<?php

define('DBSERVER', 'localhost'); // Database server

define('DBUSERNAME', 'root'); // Database username

define('DBPASSWORD', 'mysql'); // Database password

define('DBNAME', 'WebDevCurse'); // Database name

/\* connect to MySQL database \*/

$db = mysqli\_connect(DBSERVER, DBUSERNAME, DBPASSWORD, DBNAME);

// Check db connection

if($db === false){

    die("Error: connection error. " . mysqli\_connect\_error());

}

?>

1. yourAccount.php

<?php

require\_once "../config.php";

require\_once "checkFunctions.php";

session\_start();

global $error;

$error = '';

if ($\_SERVER["REQUEST\_METHOD"] == "POST" && isset($\_POST['submit'])) {

    $name = trim($\_POST['name']);

    $email = trim($\_POST['email']);

    $entered\_password = trim($\_POST['password']);

    $real\_passwordVerify = trim($\_POST['passwordVerify']);

    $nickname = $\_POST['fullname'];

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

    $userID = $\_SESSION['userId'];

    $user = checkUserExists($db, $name, $email);

    if ($user) {

        if ($user['name'] == $name) {

            $error = '<div style="background: rgb(104, 0, 0);

                        border: 0.125rem solid;

                        padding: 8px;

                        border-radius: 10px;

                        display:inline-block;

                        color: rgb(184, 0, 0);">This username is already taken</div>';

        } else if ($user['email'] == $email) {

            $error = '<div style="background: rgb(104, 0, 0);

                        border: 0.125rem solid;

                        padding: 8px;

                        border-radius: 10px;

                        display:inline-block;

                        color: rgb(184, 0, 0);">This email is already taken</div>';

        }

    }

    if(!empty($name) && empty($error)) {

        $query = "UPDATE users SET name=(?) WHERE id=$userID";

        $statement = $db->prepare($query);

        $statement->bind\_param('s', $name);

        $statement->execute();

        $\_SESSION['userLogin'] = $name;

        $error = '<div style="background: rgb(28, 0, 104);

                            border: 0.125rem solid;

                            padding: 8px;

                            border-radius: 10px;

                            display:inline-block;

                            color: rgb(28, 0, 184);"> Info has been changed. </div>';

    }

    if(!empty($email) && empty($error)) {

        $query = "UPDATE users SET email=(?) WHERE id=$userID";

        $statement = $db->prepare($query);

        $statement->bind\_param('s', $email);

        $statement->execute();

        $\_SESSION['userEmail'] = $email;

        $error = '<div style="background: rgb(28, 0, 104);

                            border: 0.125rem solid;

                            padding: 8px;

                            border-radius: 10px;

                            display:inline-block;

                            color: rgb(28, 0, 184);"> Info has been changed. </div>';

    }

    if(!empty($nickname)) {

        $query = "UPDATE users SET fullname=(?) WHERE id=$userID";

        $statement = $db->prepare($query);

        $statement->bind\_param('s', $nickname);

        $statement->execute();

        $\_SESSION['userFullname'] = $nickname;

        $error = '<div style="background: rgb(28, 0, 104);

                            border: 0.125rem solid;

                            padding: 8px;

                            border-radius: 10px;

                            display:inline-block;

                            color: rgb(28, 0, 184);"> Info has been changed. </div>';

    }

    if(!empty($description)) {

        $query = "UPDATE users SET description=(?) WHERE id=$userID";

        $statement = $db->prepare($query);

        $statement->bind\_param('s', $description);

        $statement->execute();

        $\_SESSION['userDescription'] = $description;

        $error = '<div style="background: rgb(28, 0, 104);

                            border: 0.125rem solid;

                            padding: 8px;

                            border-radius: 10px;

                            display:inline-block;

                            color: rgb(28, 0, 184);"> Info has been changed. </div>';

    }

    if (!empty($entered\_password)) {

        if ($entered\_password!=$real\_passwordVerify) {

            $error = '<div style="background: rgb(104, 85, 0);

                                border: 0.125rem solid;

                                padding: 8px;

                                border-radius: 10px;

                                display:inline-block;

                                color: rgb(184, 150, 0);">Password did not match.</div>';

        } else if (password\_verify($entered\_password, $\_SESSION['userPassword'])) {

            $error = '<div style="background: rgb(104, 85, 0);

                                border: 0.125rem solid;

                                padding: 8px;

                                border-radius: 10px;

                                display:inline-block;

                                color: rgb(184, 150, 0);">This password is already in use</div>';

        } else {

            $hashedPass = password\_hash($entered\_password, PASSWORD\_BCRYPT);

            $query = "UPDATE users SET password=(?) WHERE id=$userID";

            $statement = $db->prepare($query);

            $statement->bind\_param('s', $hashedPass);

            $statement->execute();

            $\_SESSION['userPassword'] = $hashedPass;

            $\_SESSION['password\_changed'] = true;

            header("location: ../../userstatus/login.php");

        }

    }

}

?>

<!DOCTYPE html>

<html lang="en">

<head>

    <link rel="preconnect" href="https://fonts.googleapis.com">

    <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

    <link href="https://fonts.googleapis.com/css2?family=Oxanium&family=Share+Tech+Mono&display=swap" rel="stylesheet">

    <link href="https://use.fontawesome.com/releases/v5.6.1/css/all.css" rel="stylesheet">

    <link href="../../styles/styles.css" rel="stylesheet">

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title><?=$\_SESSION['userLogin']?></title>

</head>

<body>

    <div style="position:-webkit-sticky;

                position:sticky;

                top:0;

                color:#cf0101;

                background-color: #000000;

                background-image: url(../../styles/accbg.gif);

                height: 50px;

                width: 100%">

    </div>

    <div>

        <input type="checkbox" id="check">

        <label for="check">

            <i class="fas fa-bars" id="btn"></i>

            <i class="fas fa-times" id="cancel"></i>

        </label>

        <div class="sidebar">

            <header class="glitch" style="background-image: url(../../styles/accbg.gif);">Menu</header>

            <a href="../../welcome.php" class="active">

                <i class="fas fa-qrcode"></i>

                <span>Dashboard</span>

            </a>

            <a href="yourAccount.php">

                <i class="fas fa-address-card"></i>

                <span>Account</span>

            </a>

            <a href="../overview/labsoverview.php">

                <i class="fas fa-stream"></i>

                <span>Labs overview</span>

            </a>

            <!-- IN PROGGRESS -->

            <a href="#">

                <i class="fas fa-calendar"></i>

                <span>Events</span>

            </a>

            <a href="#">

                <i class="far fa-question-circle"></i>

                <span>About</span>

            </a>

            <a href="#">

                <i class="fas fa-sliders-h"></i>

                <span>Services</span>

            </a>

            <!-- IN PROGGRESS -->

            <a target="\_blank" rel="noopener noreferrer" href="https://vk.com/wemik2323">

                <i class="far fa-envelope"></i>

                <span>Contact</span>

            </a>

            <a href="../../userstatus/logout.php">

                <i class="far fa-address-card"></i>

                <span>Log Out</span>

            </a>

        </div>

    </div>

</body>

<div class="Main\_Window" style="margin-top:0;">

    <h2 style="margin-top:40px; margin-bottom: 40px;">Your profile</h2>

    <center><form action="" method="post" style="width:600px; height: 100%;">

        <p style="margin:3%;">Your nickname: <?=$\_SESSION["userLogin"]?>#<?=$\_SESSION['userId']?></p>

        <div class="form-example">

            <label style="color: #ffffff; font-family:Share Tech Mono;" for="name">Change your nickname: </label>

            <input type="text" name="name" id="name" value=""/>

        </div>

        <p style="margin:3%;">Your email: <?=$\_SESSION['userEmail']?></p>

        <div class="form-example">

            <label style="color: #ffffff;font-family:Share Tech Mono;" for="email">Change your email: </label>

            <input type="email" name="email" id="email" value=""/>

        </div>

        <p style="margin:3%;">Change password:</p>

        <div>

            <label style="color: #ffffff;font-family:Share Tech Mono;" for="password">Enter a new password: </label>

            <input type="password" name="password" id="password" value=""/>

        </div>

        <div style="margin:2%;">

            <label style="color: #ffffff;font-family:Share Tech Mono;" for="password">Verify your password: </label>

            <input type="password" name="passwordVerify" id="passwordVerify" value=""/>

        </div>

        <p style="margin:3%;">Full name: <?=$\_SESSION['userFullname']?></p>

        <div>

            <label style="color: #ffffff;font-family:Share Tech Mono;" for="password">Change your full name: </label>

            <input type="text" name="fullname" id="fullname" value=""/>

        </div>

        <p style="margin:3%;">Description: <?=$\_SESSION['userDescription']?></p>

        <div>

            <label style="color: #ffffff;font-family:Share Tech Mono;" for="password">Change description: </label>

            <input type="text" name="description" id="description" value=""/>

        </div><br>

        <?=$error;?>

        <div>

            <input type="submit" name="submit" style="font-family: Share Tech Mono;

            font-style: normal;

            font-weight: 400;

            font-size: 20px;

            text-transform: uppercase;

            position: relative;

            display: inline-block;

            padding: 0.75rem 1.1875rem;

            cursor: pointer;

            color: #e11900;

            background: rgba(0,0,0,.3);

            border: 0.125rem solid;

            border-left: 0.5rem solid;

            margin-top: 4%;

            outline: none;"

            value="Submit" style="margin-top: 4%; margin-right: 0%; margin-bottom: 2%;">

        </div>

    </form></center>

</div>

<div style="bottom:0;height:100px; width:100%; margin-top:0; background-color:#000000; background-image:url(../../styles/accbg.gif); color:#ffffff;"></div>

</html>

1. checkFunction.php

<?php

require\_once('../config.php');

function checkUserExists($mysqli, $loginCheck, $emailCheck)

{

    $stmt = $mysqli->prepare("SELECT \* FROM users WHERE name=? or email=?");

    $stmt->bind\_param("ss", $loginCheck, $emailCheck);

    $stmt->execute();

    $result = $stmt->get\_result();

    return $result->fetch\_assoc();

}

?>

1. labsoverview.php

<?php

require\_once "../config.php";

// start the session

session\_start();

// Check if the user is not logged in, then redirect the user to login page

if ($\_SESSION["userLoggedIn"] !== true) {

    header("location: userstatus/login.php");

}

global $uploaded\_files;

$uploaded\_files = "";

include "helpFunctions.php";

$error = "";

if (isset($\_FILES['file'])) {

    $target\_path = 'uploads/';

    $target\_path = $target\_path.time().'\_'.basename($\_FILES['file']['name']);

    if(move\_uploaded\_file($\_FILES['file']['tmp\_name'], $target\_path)) {

        $error = "The file ".basename( $\_FILES['file']['name'])." has been uploaded";

    } else{

        $error = "There was an error uploading the file, please try again!";

    }

    if(strlen($error) > 0)

    {

        $error = '<p>'.$error.'</p>';

    }

}

/\*\* List Loaded Files DATE\_NAME\_TYPE \*\*/

//Read directory

if ($\_SESSION['userStatus'] == 'donated' || $\_SESSION['userStatus'] == 'admin') {

    $dh = opendir('uploads/');

    while (($file = readdir($dh)) !== false) {

        if($file != '.' && $file != '..') {

            $filename = 'uploads/'.$file;

            $parts = explode("\_", $file);

            $size = formatBytes(filesize($filename));

            $added = date("m/d/Y", $parts[0]);

            $origName = $parts[1];

            $uploaded\_files .= "<li><a href=\"$filename\" download>$origName</a> $size - $added</li>\n";

        }

    }

    closedir($dh);

    if(strlen($uploaded\_files) == 0) {

        $uploaded\_files = "<li><em>No files found</em></li>";

    }

} else {

    $uploaded\_files .= '<form action="" method="post">

                        <p>Give me money to gain access to WebDev labs. pls...</p>

                        <p><input type="submit" name="donate" value="Donate" style="font-family: Share Tech Mono;

                                                                                    font-style: normal;

                                                                                    font-weight: 400;

                                                                                    font-size: 20px;

                                                                                    text-transform: uppercase;

                                                                                    position: relative;

                                                                                    display: inline-block;

                                                                                    padding: 0.5rem 1.1875rem;

                                                                                    cursor: pointer;

                                                                                    color: #e11900;

                                                                                    background: rgba(0,0,0,.3);

                                                                                    border: 0.125rem solid;

                                                                                    border-left: 0.5rem solid;

                                                                                    margin-top: 20px;

                                                                                    margin-bottom: 20px;

                                                                                    outline: none;"></p>

                        </form>';

}

$adminFileLoad = '';

if ($\_SESSION['userStatus'] == 'admin') {

    $adminFileLoad = '<fieldset>

                            <legend><p>Add a new file to the storage</p></legend>

                            <form method="post" action="labsoverview.php" enctype="multipart/form-data">

                            <input type="hidden" name="MAX\_FILE\_SIZE" value="100000" />

                            <img src="chngfile.png" alt="" style="margin:30px;" />

                            <input type="file" name="file" style="display:block;

                                                            postition:absolute;

                                                            border: 1px solid;

                                                            margin-top: -70px;

                                                            margin-left: 35%;

                                                            height:50px;

                                                            width:250px;

                                                            opacity: 0;

                                                            cursor: pointer;

                                                            "/></p>

                            <p><input type="submit" name="submit" value="upload" style="font-family: Share Tech Mono;

                                                                                        font-style: normal;

                                                                                        font-weight: 400;

                                                                                        font-size: 15px;

                                                                                        text-transform: uppercase;

                                                                                        position: relative;

                                                                                        display: inline-block;

                                                                                        padding: 0.5rem 1.1875rem;

                                                                                        cursor: pointer;

                                                                                        color: #e11900;

                                                                                        background: rgba(0,0,0,.3);

                                                                                        border: 0.125rem solid;

                                                                                        border-left: 0.5rem solid;

                                                                                        margin-top: 10px;

                                                                                        margin-bottom: 10px;

                                                                                        outline: none;" /></p>

                            </form>

                        </fieldset>';

}

?>

<!DOCTYPE html>

<html lang="en">

<title>Labs</title>

<head>

    <link rel="preconnect" href="https://fonts.googleapis.com">

    <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

    <link href="https://fonts.googleapis.com/css2?family=Oxanium&family=Share+Tech+Mono&display=swap" rel="stylesheet">

    <link href="https://use.fontawesome.com/releases/v5.6.1/css/all.css" rel="stylesheet">

    <link href="../../styles/styles.css" rel="stylesheet">

    <link href="../../styles/fileMenu.css" rel="stylesheet">

</head>

<body>

    <div style="position:-webkit-sticky;

                position:sticky;

                top:0;

                color:#cf0101;

                background-color: #000000;

                background-image: url(../../styles/accbg.gif);

                height: 50px;

                width: 100%">

    </div>

    <div>

        <input type="checkbox" id="check">

        <label for="check">

            <i class="fas fa-bars" id="btn"></i>

            <i class="fas fa-times" id="cancel"></i>

        </label>

        <div class="sidebar">

            <header class="glitch" style="background-image: url(../../styles/accbg.gif);">Menu</header>

            <a href="../../welcome.php" class="active">

                <i class="fas fa-qrcode"></i>

                <span>Dashboard</span>

            </a>

            <a href="../account/yourAccount.php">

                <i class="fas fa-address-card"></i>

                <span>Account</span>

            </a>

            <a href="labsoverview.php">

                <i class="fas fa-stream"></i>

                <span>Labs overview</span>

            </a>

            <!-- IN PROGGRESS -->

            <a href="#">

                <i class="fas fa-calendar"></i>

                <span>Events</span>

            </a>

            <a href="#">

                <i class="far fa-question-circle"></i>

                <span>About</span>

            </a>

            <a href="#">

                <i class="fas fa-sliders-h"></i>

                <span>Services</span>

            </a>

            <!-- IN PROGGRESS -->

            <a target="\_blank" rel="noopener noreferrer" href="https://vk.com/wemik2323">

                <i class="far fa-envelope"></i>

                <span>Contact</span>

            </a>

            <a href="../../userstatus/logout.php">

                <i class="far fa-address-card"></i>

                <span>Log Out</span>

            </a>

        </div>

    </div>

</body>

<div class="Main\_Window" style="margin-top:50px;">

    <h2 style="margin-bottom:30px;">Online File Storage</h2>

    <?=$adminFileLoad?>

    <fieldset>

        <legend><p>Uploaded files</p></legend>

        <ul id="files">

            <?php

                if (isset($\_POST['donate'])) {

                    ?> <script>alert("Я как будто бы получил деньги... Но у меня нет ИНН и организации с сотрудниками и налогами >:')");</script><?php

                    //Это произойдет после выполнения запроса в SperEquaring и получения чека

                    //.... $response = curl\_exec($myCurl); curl\_close($myCurl); ....

                    // + запись в бд с услугами donates с историей платы и ссылками на чек и т.д....

                    $userId = $\_SESSION['userId'];

                    $query = "UPDATE users SET status='donated' WHERE id=$userId";

                    $statement = $db->prepare($query);

                    $statement->execute();

                    $\_SESSION['userStatus'] = 'donated';

                }

                echo $uploaded\_files;

            ?>

        </ul>

    </fieldset>

</div>

    <?=$error?>

<div style="height:100px; width:100%; margin-top:0; background-color:#000000; background-image:url(../../styles/accbg.gif); color:#ffffff;"></div>

</html>

1. sberIntegration.php

<?php

$timestamp = date('dmYHis');  // это у нас будет номер заказа

// $pre\_order - это сумма, которую надо заплатить. Берётся из формы.

$pre\_order = str\_replace(' ', '', $pre\_order); // 1 000 -> 1000

$pre\_order = $pre\_order\*100; // и умножить на сто, т.к. в запросе в банк надо указывать сумму в копейках.

// Отправляем запрос в банк

$myCurl = curl\_init();

curl\_setopt\_array($myCurl, array(

    CURLOPT\_URL => 'https://securepayments.sberbank.ru/payment/rest/register.do?userName=XXXXXXXXXXXXX-api&password=XXXXXXXXXXXXX&amount='.$pre\_order.'&language=ru&orderNumber='.$timestamp.'&returnUrl=https://XXXXXXXXXXXXXXXX/send',

// Вот эта строка и есть запрос в банк. Тут нужно указать только свой userName, password, сумму и номер заказа, а так же returnUrl= адрес страницы куда мы вернёмся после оплаты.

    CURLOPT\_RETURNTRANSFER => true,

    CURLOPT\_POST => true,

    CURLOPT\_POSTFIELDS => http\_build\_query(array(''))

));

$response = curl\_exec($myCurl);

curl\_close($myCurl);

// Всё. Запрос отправлен. В ответ мы получаем строку с ID заказа и ссылкой на оплату.

// Обрезаем ответ, чтобы получить url

$url = strstr($response, 'https:');

$url = strstr($url, '"', true);

// Обрезаем ответ, чтобы получить id заказа

$orderid = strstr($response, '","formUrl"', true);

$orderid = mb\_substr($orderid, 12);

// Создаём файл с именем = id заказа. Именно так, потому что потом мы по этому id этот файл и будем находить.

$myfile='../orders/'.$orderid.'.txt';

// Заполняем его данными из формы

$text .= $familiya.";";

$text .= $imya.";";

$text .= $otchestvo.";";

$text .= $tel.";";

$text .= $email.";";

$text .= $city.";";

$text .= $course\_info.";";

$text .= $price.";";

$text .= $pre\_order."";

$file=fopen($myfile,'x+');

fputs($file,$text);

fclose($file);

// Переходим на страницу оплаты. $url мы получили из ответа от банка

header('Location: '.$url);

?>

1. helpFunctions.php

<?php

function formatBytes($bytes, $precision = 2) {

    $units = array('B', 'KB', 'MB', 'GB', 'TB');

    $bytes = max($bytes, 0);

    $pow = floor(($bytes ? log($bytes) : 0) / log(1024));

    $pow = min($pow, count($units) - 1);

    $bytes /= pow(1024, $pow);

    return round($bytes, $precision) . ' ' . $units[$pow];

}

?>

Файлы стилей (.css):

1. fileMenu.css

ul#files

{

    list-style-type: none;

    margin: 40px 0px 0px 0px;

    padding: 0px;

}

ul#files li

{

    background-color: #fff7c0;

    border-bottom: 1px solid #efefef;

    padding: 2px;

    margin-bottom: 1px;

}

1. register.css

\* {

    color: #000000;

    margin: 0;

    padding: 0;

    font-family: "Share Tech Mono";

}

body{

    display: flex;

    flex-direction: column;

    justify-content: center;

    align-items: center;

    width: 100vw;

    height: 100vh;

    background-color: #181818;

}

h2 {

    position: relative;

    text-align: center;

    font-size: 60px;

    font-family: 'Oxanium';

    color:#ffda62;

    margin: 0;

}

p {

    font-family: 'Oxanium';

    font-weight: 300;

    text-align: center;

    font-size: 30px;

    color:#c49d1d;

    margin: 0;

}

.hintBar {

    background: rgb(104, 85, 0);

    border: 0.125rem solid;

    padding: 8px;

    border-radius: 10px;

    display:inline-block;

    color: rgb(184, 150, 0);

}

.errorBar {

    background: rgb(104, 0, 0);

    border: 0.125rem solid;

    padding: 8px;

    border-radius: 10px;

    display:inline-block;

    color: rgb(184, 0, 0);

}

.successBar {

    background: rgb(6, 104, 0);

    border: 0.125rem solid;

    padding: 8px;

    border-radius: 10px;

    display:inline-block;

    color: rgb(6, 184, 0);

}

.container .form-group {

    clear:both;

    text-align:right;

}

.container .form-group label {

    display: inline;

    float: left;

    padding-left: 20%;

    margin-left: 20px;

    margin-top: 10px;

    margin-bottom: 10px;

    font-size: 24px;

    color:white;

}

.container .form-group input {

    display: inline;

    margin-right: 28%;

    margin-left: auto;

    margin-top: 10px;

    margin-bottom: 10px;

    /\* width: 30px; \*/

    height: 20px;

}

.GTFObutton {

    font-family: Share Tech Mono;

    font-style: normal;

    font-weight: 400;

    font-size: 20px;

    text-transform: uppercase;

    position: relative;

    display: inline-block;

    padding: 0.75rem 1.1875rem;

    cursor: pointer;

    color: #e11900;

    background: rgba(0,0,0,.3);

    border: 0.125rem solid;

    border-left: 0.5rem solid;

    margin-top: 2%;

    outline: none;

}

/\* Register animations \*/

.container {

    width: 50%;

    height: 30%;

    margin: auto;

    text-align: center;

    -webkit-animation-duration: 5s;

    animation-duration: 5s;

    -webkit-animation-fill-mode: both;

    animation-fill-mode: both;

    -webkit-animation-name: samefadeIn;

    animation-name: samefadeIn;

}

@-webkit-keyframes samefadeIn {

    0% {opacity: 0;}

    100% {opacity: 1;}

}

@keyframes samefadeIn {

    0% {opacity: 0;}

    100% {opacity: 1;}

}

1. styles.css

\* {

    font-family: 'Share Tech Mobo';

    margin: 0;

    padding: 0;

    text-decoration: none;

}

body{

    display: flex;

    flex-direction: column;

    justify-content: center;

    align-items: center;

    width: 100vw;

    height: 100vh;

    background-color: #181818;

}

h2 {

  position: relative;

  text-align: center;

  font-size: 60px;

  font-family: 'Oxanium';

  color:#ffda62;

  margin: 0;

}

p {

  font-family: 'Oxanium';

  font-weight: 300;

  text-align: center;

  font-size: 30px;

  color:#c49d1d;

  margin: 0;

}

.hintBar {

  background: rgb(104, 85, 0);

  border: 0.125rem solid;

  padding: 8px;

  border-radius: 10px;

  display:inline-block;

  color: rgb(184, 150, 0);

}

.errorBar {

  background: rgb(104, 0, 0);

  border: 0.125rem solid;

  padding: 8px;

  border-radius: 10px;

  display:inline-block;

  color: rgb(184, 0, 0);

}

.successBar {

  background: rgb(28, 0, 104);

  border: 0.125rem solid;

  padding: 8px;

  border-radius: 10px;

  display:inline-block;

  color: rgb(28, 0, 184);

}

#clock {

  font-family: 'Share Tech Mono';

  color: #ffffff;

}

label #btn,label #cancel{

  position: absolute;

  left: 5px;

  cursor: pointer;

  color: rgb(175, 175, 175);

  border-radius: 5px;

  margin: 15px 30px;

  font-size: 29px;

  background-color: #555555;

  box-shadow:inset 2px 2px 2px 0px rgba(255,255,255,.5),

      inset -7px -7px 10px 0px rgba(0,0,0,.1),

      3.5px 3.5px 20px 0px rgba(0,0,0,.1),

      2px 2px 5px 0px rgba(0,0,0,.1);

  height: 45px;

  width: 45px;

  text-align: center;

  text-shadow: 2px 2px 3px rgba(255,255,255,0.5);

  line-height: 45px;

  transition: all .5s ease;

}

label #cancel{

  opacity: 0;

  visibility: hidden;

}

#check:checked ~ .sidebar{

  left: 0;

}

#check:checked ~ label #btn{

  margin-left: 245px;

  opacity: 0;

  visibility: hidden;

}

#check:checked ~ label #cancel{

  margin-left: 245px;

  opacity: 1;

  visibility: visible;

}

.sidebar{

    position: fixed;

    width: 240px;

    left: -240px;

    height: 100%;

    background-color:  #555555;

    transition: all .5s ease;

}

.sidebar header{

    font-size: 28px;

    line-height: 70px;

    background-color: #000000;

    user-select: none;

    text-transform: uppercase;

    font-family: 'Oxanium';

    color: #ffffff;

}

.sidebar a{

    display: block;

    height: 65px;

    width: 100%;

    line-height: 65px;

    padding-left: 30px;

    box-sizing: border-box;

    border-left: 5px solid transparent;

    font-family: 'Share Tech Mono';

    color: #cf0101;

    transition: all .5s ease;

}

a.active,a:hover{

    border-left: 5px solid var(--accent-color);

    color: #4b3232;

    background: linear-gradient(to left, var(--accent-color), var(--gradient-color));

}

.sidebar a i{

    font-size: 23px;

    margin-right: 16px;

}

.sidebar a span{

    letter-spacing: 1px;

    text-transform: uppercase;

}

#check{

    display: none;

}

@media(max-width : 860px) {

    .sidebar{

        height: auto;

        width: 70px;

        left: 0;

        margin: 100px 0;

    }

    header,#btn,#cancel{

        display: none;

    }

    span{

        position: absolute;

        margin-left: 23px;

        opacity: 0;

        visibility: hidden;

    }

    .sidebar a{

        height: 60px;

    }

    .sidebar a i{

        margin-left: -10px;

    }

    a:hover {

        width: 200px;

        background: inherit;

    }

    .sidebar a:hover span{

        opacity: 1;

        visibility: visible;

    }

}

/\* ------------- Animations & CHNG colors ------------- \*/

.sidebar > a.active,.sidebar > a:hover:nth-child(even) {

  --accent-color: #f48552;

  --gradient-color: #3f2708;

}

.sidebar a.active,.sidebar > a:hover:nth-child(odd) {

  --accent-color: #f48552;

  --gradient-color: #3f2708;

}

/\* Main page animations \*/

.Main\_Window {

    width: 50%;

    height: 30%;

    margin: auto;

    text-align: center;

    -webkit-animation-duration: 5s;

    animation-duration: 5s;

    -webkit-animation-fill-mode: both;

    animation-fill-mode: both;

    -webkit-animation-name: samefadeIn;

    animation-name: samefadeIn;

}

@-webkit-keyframes samefadeIn {

    0% {opacity: 0;}

    100% {opacity: 1;}

}

@keyframes samefadeIn {

    0% {opacity: 0;}

    100% {opacity: 1;}

}

.sidebar {

  .glitch {

    position: relative;

    padding-left: 80px;

    &:before, &:after {

        content: 'Menu';

        position: absolute;

        top: 0;

        overflow: hidden;

        padding-left: 80px;

    }

    &:before {

        left:3px;

        text-shadow: -2px 0 red;

        animation: glitch-before 2s linear 0s infinite alternate-reverse;

    }

    &:after {

        left:-3px;

        text-shadow: -2px 0 blue;

        animation: glitch-after 2s linear 0s infinite alternate-reverse;

    }

  }

}

@keyframes glitch-before {

    0% {

      clip: rect(20px, 350px, 124px, 30px);

    }

    5% {

      clip: rect(107px, 350px, 146px, 30px);

    }

    10% {

      clip: rect(67px, 350px, 140px, 30px);

    }

    15% {

      clip: rect(141px, 350px, 126px, 30px);

    }

    20% {

      clip: rect(59px, 350px, 47px, 30px);

    }

    25% {

      clip: rect(142px, 350px, 88px, 30px);

    }

    30% {

      clip: rect(138px, 350px, 73px, 30px);

    }

    35% {

      clip: rect(26px, 350px, 5px, 30px);

    }

    40% {

      clip: rect(94px, 350px, 64px, 30px);

    }

    45% {

      clip: rect(83px, 350px, 28px, 30px);

    }

    50% {

      clip: rect(27px, 350px, 29px, 30px);

    }

    55% {

      clip: rect(5px, 350px, 142px, 30px);

    }

    60% {

      clip: rect(118px, 350px, 13px, 30px);

    }

    65% {

      clip: rect(14px, 350px, 80px, 30px);

    }

    70% {

      clip: rect(60px, 350px, 5px, 30px);

    }

    75% {

      clip: rect(120px, 350px, 132px, 30px);

    }

    80% {

      clip: rect(55px, 350px, 131px, 30px);

    }

    85% {

      clip: rect(70px, 350px, 34px, 30px);

    }

    90% {

      clip: rect(46px, 350px, 50px, 30px);

    }

    95% {

      clip: rect(15px, 350px, 101px, 30px);

    }

    100% {

      clip: rect(137px, 350px, 93px, 30px);

    }

  }

  @keyframes glitch-after {

    0% {

      clip: rect(105px, 350px, 142px, 30px);

    }

    5% {

      clip: rect(59px, 350px, 101px, 30px);

    }

    10% {

      clip: rect(72px, 350px, 96px, 30px);

    }

    15% {

      clip: rect(142px, 350px, 125px, 30px);

    }

    20% {

      clip: rect(2px, 350px, 78px, 30px);

    }

    25% {

      clip: rect(135px, 350px, 75px, 30px);

    }

    30% {

      clip: rect(49px, 350px, 33px, 30px);

    }

    35% {

      clip: rect(100px, 350px, 43px, 30px);

    }

    40% {

      clip: rect(49px, 350px, 40px, 30px);

    }

    45% {

      clip: rect(56px, 350px, 16px, 30px);

    }

    50% {

      clip: rect(83px, 350px, 14px, 30px);

    }

    55% {

      clip: rect(64px, 350px, 86px, 30px);

    }

    60% {

      clip: rect(127px, 350px, 127px, 30px);

    }

    65% {

      clip: rect(32px, 350px, 120px, 30px);

    }

    70% {

      clip: rect(114px, 350px, 118px, 30px);

    }

    75% {

      clip: rect(142px, 350px, 113px, 30px);

    }

    80% {

      clip: rect(44px, 350px, 139px, 30px);

    }

    85% {

      clip: rect(93px, 350px, 34px, 30px);

    }

    90% {

      clip: rect(58px, 350px, 40px, 30px);

    }

    95% {

      clip: rect(46px, 350px, 3px, 30px);

    }

    100% {

      clip: rect(129px, 350px, 104px, 30px);

    }

}

База данных зарегистрированных пользователей:

