

ДОКУМЕНТАЦИЈА ЗА ЛАБОРАТОРИСКА ВЕЖБА 1 ПО ПРЕДМЕТОТ ИНТЕРНЕТ ТЕХНОЛОГИИ

1. Вовед

1.1 Цел на лабораториската вежба

Цел на оваа лабораториска вежба е да се имплементира комплетен систем за корисничка автентификација. Системот обезбедува безбедна регистрација, најава и управување со корисничките податоци.

Оваа лабораториска вежба е изработена од МАртина Петковска со индекс број 223313.

Целиот документ може да се најде на GitHub: https://github.com/mpetkovska27/IB_2025_lab1

1.2 Користени технологии

За изработка на овој проект е користен програмскиот јазик PHP, за пишување на backend логиката, HTML5 CSS за frontend и SQLite за едноставана база на податоци.

СТРУКТУРА НА ПРОЕКТОТ:

```
ib_lab1_223313/
- assets/
- css/
    o style.css      # Стилови за апликацијата
- config/
    o db_connection.php  # Конекција со SQLite база
    o session.php       # Управување со сесии и cookies
- includes/
    o auth.php        # Функции за автентификација
- database/
    o db.sqlite        # SQLite база на податоци
- index.php          # Главна страна (за најавени корисници)
- register.php       # Регистрација
- login.php          # Најава
- logout.php         # Одјава
- edit_profile.php   # Менаџирање со корисничките податоци
```

2. Функционалности

2.1 Корисничка регистрација (register.php)

- Внес на кориснички податоци (Username, Email Address и Password)
- Валидација
 - Username: 3-50 карактери

- Email: валиден формат
- Password: повеќе од 8 карактери, барем една голема буква, една мала, бројка и специјален знак.
- Хеширање на лозинка со password_hash()
- Чување податоци во SQLite база

Register

Username:

Email Address:

Password:

Password must be longer than 8 characters and contain at least one uppercase letter, one lowercase letter, one number, and one special character.

Confirm Password:

Register

Already have an account? [Log in](#)

2.2 Корисничка најава (login.php)

- Најава со username или email
 - Верификација на лозинка со password_verify() - повеќе од 8 карактери, барем една голема буква, една мала, бројка и специјален знак.
- Генерирање на session token и поставување на истиот како HttpOnly и Secure cookie
- Сесија за најавен корисник

Login

Username or Email:

Password:

Log In

Don't have an account? [Register](#)

2.3 Главна страна (index.php)

- Приказ на информации за најавениот корисник
- Листа на сите регистрирани корисници
- Навигација до Edit Profile и Logout
- Само за најавени корисници

Welcome, mpet! [Edit Profile](#) [Logout](#)

User Authentication System

Lab exercise, made by Martina Petkovska - 223313, that implements a complete authentication solution for secure user management

About This Lab Exercise

This application demonstrates a complete user authentication system implemented as part of a laboratory exercise. The system provides:

- secure user registration,
- login functionality
- and user data management.

Your Account Information

Username: mpet
Email: mpet@test.com
Registered on: November 10, 2025 at 5:38 PM

Registered Users

ID	Username	Email	Registered
8	mpet	mpet@test.com	Nov 10, 2025
7	asto	asto@test.com	Nov 9, 2025
4	vbla	vbla@test.com	Nov 9, 2025
2	ilat	ilat@test.com	Nov 9, 2025

Total users: 4

© 2025 User Authentication System - Laboratory Exercise 1

2.4 Менаџирање со кориснички податоци (edit_profile.php)

- Ажурирање на веќе постоечки информации (име, презиме, username и email)
- Промена на лозинка
 - Верификација на тековната лозинка
 - Верификација за новата лозинка - повеќе од 8 карактери, барем една голема буква, една мала, бројка и специјален знак.
- Бришење на профил

- Предупредување дека оваа акција е неповратна
- Бришење на сесија
- Потврда со лозинка

Auth System

Welcome, mpet! [Home](#) [Logout](#)

Edit Profile

Personal Information

First Name:

Last Name:

Username:

Email Address:

[Update Personal Information](#)

Change Password

Current Password:

New Password:

>Password must be longer than 8 characters and contain at least one uppercase letter, one lowercase letter, one number, and one special character.

Confirm New Password:

[Change Password](#)

Delete Account

Warning: This action cannot be undone. All your data will be permanently deleted.

Enter your password to confirm deletion:

[Delete My Account](#)

[Back to Home](#)

2.5 Одјава (logout.com)

- Бришење на сесија
- Пренасочување на login.php

Auth функции (includes/auth.php)

```
validatePassword($password)
```

- Проверува: должина >8, голема буква, мала буква, бројка, специјален знак.
- Враќа: ['valid' => false, 'message' => 'Password must be longer than 8 characters.']}

```
registerUser($username, $email, $password)
```

- Регистрира нов корисник, проверува уникатност на username и email, валидира и зачувува во база
- Хешира лозинка со password_hash()
- Враќа: ['success' => bool, 'message' => string]

```
loginUser($username, $password)
```

- Најава на корисник, прима влезни податоци за корисникот, прави валидација на лозинката
- Генерира session token
- Поставува session cookie
- Враќа: ['success' => bool, 'message' => string, 'user' => array|null]

```
getCurrentUser()
```

- Враќа информации за најавениот корисник или null ако не е најваен

```
getAllUsers()
```

- Враќа листа од сите корисници

```
updateUserInfo($userId, $username, $email, $firstName = '', $lastName = '')
```

- Ги ажурира корисничките податоци
- Проверува дали корисникот го едитира својот профил. валидира и проверува уникатност
- Ажурира сесија
- Враќа: ['success' => bool, 'message' => string]

```
updatePassword($userId, $currentPassword, $newPassword, $confirmPassword)
```

- Промена на лозинката
- Верификација на тековната лозинка
- Валидација на нова лозинка и дали се совпаѓа со старата
- Хешира и ја зачува новата лозинка
- Враќа: ['success' => bool, 'message' => string]

```
deleteUser($userId, $password)
```

- Брише кориснички профил според id на корисник
- Верифицира лозинка за потврда
- Брише од база
- Враќа: ['success' => bool, 'message' => string]

Session функции (config/session.php)

```
generateSessionToken()
```

- Генерира 64-карактен hex токен
- Користи random_bytes() и bin2hex()
- Враќа string (session token)

```
setSessionTokenCookie($token)
```

- Поставува session token како cookie
- Истекува после 7 дена
- HttpOnly и Secure (ако е HTTPS)

```
clearSessionTokenCookie()
```

- Брише session token cookie
- Поставува expire на минус време

```
isLoggedIn()
```

- Проверува дали корисникот е најавен, session token и cookie
- Споредува cookie token со session token

```
requireLogin()
```

- Пренасочува на login.php ако не е најавен

```
requireGuest()
```

- Пренасочува на index.php ако е најавен

```
logout()
```

- Брише session token cookie
- Брише session податоци
- Брише session cookie
- Уништува сесија

3. Забелешки

- Базата се креира автоматски при прво повикување
- Session cookie истекува после 7 дена
- First Name и Last Name се опционални
- Табелата со корисници покажува само ID, Username, Email и датум на регистрација
- Се користат пастелни бои и едноставен, минималистички дизајн.

Проектот имплементира основни безбедносни практики за веб-апликација за автентикација.