**Документация**

**Уеб сайт с логин/регистър**

Изготвено от:

Радослав Доларов, 11а, №22  
Мариян Томов, 11а, №14

Езици, които използваме в проекта:

Нашето уеб приложение за регистрация и вход на потребители е изградено с използването на няколко езика и технологии.

HTML (Hypertext Markup Language) играе ключова роля в създаването на структурата на уеб страниците. Той ни позволява да дефинираме основни елементи като форми за вход и регистрация, бутони и текстови полета.

CSS (Cascading Style Sheets) е от съществено значение за стилизирането и оформлението на нашите уеб страници. Чрез CSS можем да приложим стилове за цветове, шрифтове и разположение на елементите, осигурявайки приятно визуално изживяване за потребителите.

JavaScript ни предоставя възможността за добавяне на интерактивност към уеб страниците. С него можем да управляваме събития като кликвания на бутони и валидация на форми. Той е от съществено значение за функционалността на нашето приложение, включително връзката със системата за аутентикация на Firebase.

**Код**

Проекта е съдържа 2 папки и 1 html файла. Първата папка озаглавена css съдържа css файл: style.css. В style.css файлът се съдържат ключови стилове, които определят външния вид на формите за вход и регистрация в уеб приложението. Някои от най-съществените части от този CSS файл включват:

1. **Селектори за форми**: Тези части на CSS файла определят стиловете, които се прилагат към HTML формите за вход и регистрация. Това може да включва определени цветове, размери на шрифтове и разположение на елементите.
2. **Класове за текстови полета и бутони**: Стиловете, дефинирани за текстовите полета и бутоните във формите, помагат за подобряване на външния вид и усещането при въвеждане на данни от потребителите.
3. **Позициониране на елементите**: Това включва правила за позициониране и подреждане на елементите вътре във формите, за да се осигури добър потребителски опит при взаимодействие с тях.
4. **Стилове за съобщения за грешка и успех**: CSS файла съдържа стилове за съобщения за грешка или успех, които се появяват при въвеждане на невалидни или валидни данни във формите.

Тези части на CSS файла играят критична роля в стилизирането на формите за вход и регистрация и осигуряват еднообразен и професионален външен вид на уеб приложението.

**Примерен код**:

body{

    display:flex;

    justify-content: center;

    align-items: center;

    min-height: 100vh;

    background: white;

    background-size: cover;

    background-position: center;

}

.wrapper{

    position: relative;

    width: 500px;

    height: 470px;

    margin-bottom: 65px;

    background: rgba(230, 233, 255, 0.5);

    border: 2px solid rgba(255, 255, 255, .5);

    border-radius: 10px;

    box-shadow: 0 0 30px rgba(0, 0, 0, .3);

    display:flex;

    justify-content: center;

    align-items: center;

    overflow: hidden;

}

.input-box{

    position: relative;

    width: 100%;

    height: 50px;

    border-bottom: 2px solid #151E3D;

    margin: 30px 0;

}

.input-box label{

    position: absolute;

    top:50%;

    left: 5px;

    transform: translateY(-50%);

    color:#888888;

    font-weight: 500;

    pointer-events: none;

}

В папката с наименование "scripts" се съдържат три JavaScript файла, които играят ключова роля в функционирането на уеб приложението за управление на регистрацията и влизането на потребители, използващо Firebase за автентикация.

1. **register.js**:
   * Този файл включва логиката за регистрация на нов потребител в системата. Използва се Firebase Authentication за създаване на нов потребителски акаунт. Важните части от кода включват:
     + Импортиране на необходимите функции от Firebase SDK, като **initializeApp** и **createUserWithEmailAndPassword**.
     + Конфигуриране на Firebase със специфичните данни за проекта (**firebaseConfig**).
     + Инициализиране на Firebase приложението.
     + Получаване на обект за управление на автентикацията (**auth**).
     + Добавяне на слушател за събитието "click" на бутона за регистрация (**btnRegister**). При кликване се взимат потребителското име (е-мейл) и парола от полетата във формата и се изпращат към Firebase функцията **createUserWithEmailAndPassword** за създаване на нов акаунт.
     + При успешна регистрация се извежда съобщение за успешно създаден акаунт.
2. **login.js**:
   * Този файл включва логиката за влизане на потребител в системата. Използва се Firebase Authentication за проверка на потребителските данни (потребителско име и парола). Важните части от кода включват:
     + Импортиране на необходимите функции от Firebase SDK, като **initializeApp** и **signInWithEmailAndPassword**.
     + Конфигуриране на Firebase със специфичните данни за проекта (**firebaseConfig**).
     + Инициализиране на Firebase приложението.
     + Получаване на обект за управление на автентикацията (**auth**).
     + Добавяне на слушател за събитието "click" на бутона за влизане (**btnLogin**). При кликване се взимат потребителското име и парола от полетата във формата и се изпращат към Firebase функцията **signInWithEmailAndPassword** за аутентикация.
     + При успешно влизане потребителят се пренасочва към друга страница (**sait.html**), а при грешка се извежда съобщение за грешка.
3. **script.js**:
   * Този файл съдържа обща логика за превключване между формите за вход и регистрация на потребителите. Важните части от кода включват:
     + Избиране на елементи от DOM (Document Object Model), като **.wrapper**, **.login-link** и **.register-link**.
     + Добавяне на слушатели за събития "click" на връзките за превключване между формите.
     + При кликване на връзката за регистрация се добавя клас "active" към **.wrapper**, който променя стиловете на формата за регистрация.
     + При кликване на връзката за вход се премахва класът "active" от **.wrapper**, връщайки формата към стандартния си изглед за вход.

Тези JavaScript файлове сътрудничат за осигуряване на функционалността за вход и регистрация на потребители в уеб приложението, като използват Firebase за автентикация и DOM манипулации за превключване между различните форми.

Файлът index.html е сърцевината на уеб страницата за вход и регистрация на потребители. В него са обяснени най-важните части от кода:

1. **Декларация за Тип на Документа (<!DOCTYPE html>)**:
   * Това е декларацията, която определя типа на документа като HTML5.
2. **Коренов HTML Елемент (<html>)**:
   * Този елемент отваря основния HTML контейнер и задава атрибута **lang=“en“**, указващ езика на документа като английски.
3. **Секция Head (<head>)**:
   * Секцията **<head>** съдържа метаданни и ресурси за уеб страницата, включително:
     + **<meta charset=“UTF-8“>**: Задава Unicode UTF-8 като кодиране за знаци.
     + **<meta name=“viewport“ content=“width=device-width, initial-scale=1.0“>**: Задава настройки за мащабиране и размери на устройства.
     + **<title>Log In | Registration</title>**: Определя заглавието на уеб страницата.
     + **<link rel=“stylesheet“ href=“css/style.css“>**: Свързва външен CSS файл за стилизиране на страницата.
     + Заявки за предварително свързване с Google Fonts, за да се използва шрифтът „Poppins“ в уеб страницата.
     + Зарежда JavaScript файловете (**script.js**, **register.js**, **login.js**), които са включени чрез модулна система (**type=“module“**).
4. **Секция Body (<body>)**:
   * Секцията **<body>** съдържа основното съдържание на уеб страницата.
     + Елементът **<div class=“wrapper“>** съдържа две форми (**login** и **register**), които се показват или скриват в зависимост от действията на потребителя.
     + **Форма за Вход** (**<div class=“form-box login“>**):
       - Включва полета за електронна поща (**<input type=“email“ id=“email“>**) и парола (**<input type=“password“ id=“password“>**).
       - Включва контроли за запомняне на потребителя и линк към "Забравена парола".
       - Включва бутон за изпращане на формата за вход (**<button type=“submit“ class=“btn“ id=“btnLogin“>Login</button>**).
       - Съдържа линк за прехвърляне към формата за регистрация (**<a href=“#“ class=“register-link“>Register</a>**).
     + **Форма за Регистрация** (**<div class=“form-box register“>**):
       - Включва полета за потребителско име (**<input type=“text“ id=“username“>**), електронна поща (**<input type=“email“ id=“email“>**) и парола (**<input type=“password“ id=“password“>**).
       - Включва контрол за приемане на „Условия и правила“.
       - Включва бутон за изпращане на формата за регистрация (**<button type=“submit“ class=“btn“ id=“btnRegister“>Register</button>**).
       - Съдържа линк за прехвърляне към формата за вход (**<a href=“#“ class=“login-link“>Login</a>**).

Този HTML файл представлява основната структура на уеб страницата за вход и регистрация на потребители, като използва външен CSS за стилизация и JavaScript за управление на поведението и взаимодействието с потребителите.

**Източници:**

[**https://firebase.google.com/docs**](https://firebase.google.com/docs)

[**https://www.w3schools.com/css/**](https://www.w3schools.com/css/)

[**https://www.w3schools.com/html/**](https://www.w3schools.com/html/)

[**https://www.w3schools.com/html/**](https://www.w3schools.com/html/)