Text

Description automatically generated

**FACULTY OF APPLIED INFORMATION TECHNOLOGY**

**Field Of Study:** Information Technology

**Specialty:** Programming

**Name and Surname:** OLIMOV MUKHAMMAD

ID number: W68655

**Project Documentation: "Password Security Analyzer"**

**Overview:**

The "Password Security Analyzer" project is a web application designed for password strength analysis. Users can input a password through the website interface, and the application analyzes and displays the password's security level.



**Technologies:**

**Backend (Node.js with Express)**

The backend part of the application is implemented using Node.js and the Express framework. Express handles HTTP requests and routing. The backend also includes the following components:

**- \*\*body-parser\*\*:** Used for parsing JSON data sent from the frontend.

**- \*\*checkPasswordStrength()\*\*:** A function that assesses password complexity based on defined criteria.

**Routes**

**- \*\*POST /:\*\*** Handles requests for password analysis. Validates the password and returns its complexity level in JSON format.

**- \*\*GET /:\*\*** Serves the static HTML file containing the web application interface.

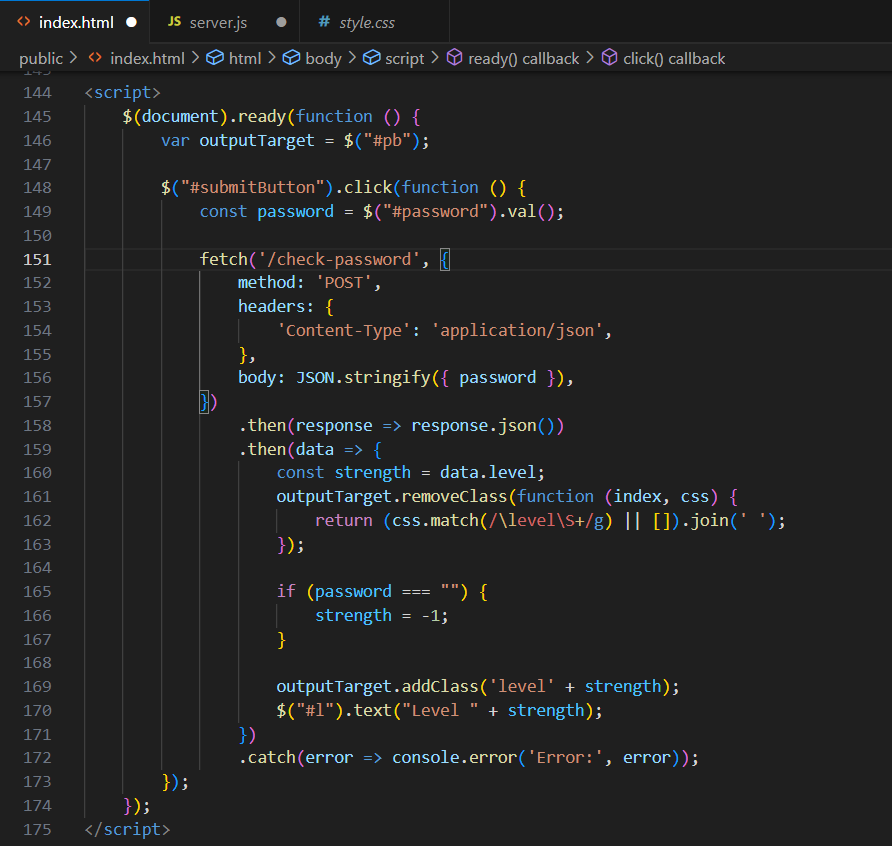


The thing is that when user input some random password (considering requirements) and clicks “Check Password” button, after that NODE.JS and server validates the password and returns its level in JSON format, after that user can see his password strength in HTML page. It’s POST and GET methods.

A screen shot of a computer program

Description automatically generated





So, and this is server.js code and usage of Express Framework. Consequently, here is my backend implementation.

**Frontend (HTML, CSS, Bootstrap, jQuery, AngularJS, JavaScript, WebFont.js, Fonts)**

The frontend part is built using the following technologies:

**HTML (Hypertext Markup Language):**

HTML forms the backbone of the web application's structure, defining the layout and elements of the webpage.

**CSS (Cascading Style Sheets):**

CSS is employed to style the HTML elements, including setting fonts, colors, backgrounds, and layout properties, enhancing the visual presentation of the application.

**Bootstrap Framework:**

The Bootstrap CSS framework is integrated into the application to leverage its pre-designed components and responsive styles, facilitating rapid development and ensuring cross-device compatibility.

Bootstrap JavaScript components are also utilized for enhanced interactivity and user experience.

**jQuery Library**:

jQuery, a JavaScript library, is utilized for simplified HTML document traversal and manipulation, event handling, and animation within the application.

**AngularJS Framework:**

Although included, AngularJS, a comprehensive JavaScript framework maintained by Google for building dynamic web applications, is not actively utilized in this code snippet.

**WebFont.js:**

WebFont.js, a JavaScript library, is utilized for loading web fonts from various sources. In this application, it is used to load Google Fonts for specified font families.

**JavaScript (Vanilla):**

Custom JavaScript code is written to handle form submission and password strength checking functionalities. This involves DOM manipulation, event handling, and making asynchronous HTTP requests using the Fetch API.

**Fonts:**

The application references the 'Orbitron' font from Google Fonts, although it's not actively used in the provided code snippet.

**Interface**

The frontend includes a form for entering a password and a button for analysis. The analysis results are displayed as the password complexity level and a corresponding progress bar.

**Running the Project**

1. Ensure that Node.js is installed.

2. Clone the project repository.

3. Navigate to the project directory.

4. Install dependencies using the command `npm install`.

5. Start the server with the command `node server.js`.

6. Open a browser and go to http://localhost:8080 (or http://localhost:3000, depending on your settings).

**Usage**

1. Open the website and enter user data in the form.

2. Click the "check password" button.

3. The password complexity analysis results will be displayed as the level and a corresponding progress bar.