**Water Quality Dashboard Web Application**

**File:** index.html  
**Location:** index.html  
**Languages Used:**

* HTML
* CSS
* JavaScript

**What Was Done**

1. **User Authentication System**
   * Implemented a simple login, signup, and password reset system using JavaScript and browser localStorage.
   * Users can create accounts, log in, reset forgotten passwords, and change their password.
   * A master login is always available for admin access.
2. **Responsive Dashboard UI**
   * Designed a clean, modern dashboard interface using HTML and CSS.
   * The dashboard displays four key water quality parameters: Temperature, TDS, pH, and Turbidity.
   * The layout is responsive and adapts to different screen sizes.
3. **Live Data Fetching from Firebase**
   * The dashboard fetches real-time sensor data from a Firebase Realtime Database.
   * The ESP32 device uploads sensor readings (temperature, TDS, pH, turbidity) to the /water path in Firebase.
   * JavaScript fetches this data every 5 seconds and updates the dashboard values automatically.
4. **Result Evaluation**
   * The dashboard evaluates the fetched sensor values against safe water quality thresholds.
   * It displays a "Safe" or "Unsafe" status based on these thresholds, with color coding for clarity.
5. **User Experience Features**
   * Profile widget with dropdown for changing password and logging out.
   * Clock widget showing the last update time.
   * All navigation and UI updates are handled dynamically with JavaScript.

**How It Works**

* **User Flow:**
  + On visiting the page, users see a welcome screen and can log in or sign up.
  + After logging in, the dashboard becomes visible, showing live water quality data.
  + Users can log out or change their password from the profile menu.
* **Data Flow:**
  + The ESP32 device (using Arduino code) uploads sensor data to Firebase.
  + The web dashboard fetches this data from Firebase using the Fetch API in JavaScript.
  + The dashboard updates the displayed values and evaluates if the water is "Safe" or "Unsafe".
* **Technologies & Environment:**
  + **Frontend:** Pure HTML, CSS, and JavaScript (no frameworks required).
  + **Backend/Data Source:** Firebase Realtime Database (cloud-hosted, no server code needed).
  + **Device:** ESP32 microcontroller running Arduino code, connected to sensors and WiFi.
  + **Browser:** Any modern web browser (tested on Chrome, Edge, Firefox).
  + **Operating System:** Developed and tested on Windows, but works cross-platform.

**Summary**

This file creates a complete, interactive web dashboard for monitoring water quality in real time. It combines user authentication, live data visualization, and responsive design, all using standard web technologies and cloud database integration. The dashboard is easy to use and can be accessed from any device with a web browser.