// profile.js

import {

  getFirestore,

  doc,

  getDoc,

  setDoc

} from "https://www.gstatic.com/firebasejs/11.10.0/firebase-firestore.js";

import {

  getAuth,

  onAuthStateChanged

} from "https://www.gstatic.com/firebasejs/11.10.0/firebase-auth.js";

window.addEventListener("DOMContentLoaded", () => {

  const db = getFirestore();

  const auth = getAuth();

  let userProfile = {};

  const profileName = document.getElementById("profileName");

  const profileSchool = document.getElementById("profileSchool");

  const profileAddress = document.getElementById("profileAddress");

  const profileYearLevel = document.getElementById("profileYearLevel");

  const profileContact = document.getElementById("profileContact");

  const editProfileBtn = document.getElementById("editProfileButton");

  function renderProfile() {

    profileName.textContent = userProfile.name;

    profileSchool.textContent = userProfile.school;

    profileAddress.textContent = userProfile.address;

    profileYearLevel.textContent = userProfile.yearLevel;

    profileContact.textContent = userProfile.contact;

  }

  onAuthStateChanged(auth, async (user) => {

    if (user) {

      const userRef = doc(db, "users", user.uid);

      const snap = await getDoc(userRef);

      if (snap.exists()) {

        userProfile = snap.data();

      } else {

        // Create a new user profile if it doesn't exist

        userProfile = {

          uid: user.uid,

          name: "",

          school: "",

          address: "",

          yearLevel: "",

          contact: "",

          readingHistory: [] // Initialize readingHistory for new users

        };

        await setDoc(userRef, userProfile);

      }

      renderProfile();

    }

  });

  editProfileBtn.addEventListener("click", async () => {

    const n = await showPrompt("Enter new Student's Name:", userProfile.name);

    if (n !== null) userProfile.name = n.trim();

    const s = await showPrompt("Enter new School:", userProfile.school);

    if (s !== null) userProfile.school = s.trim();

    const a = await showPrompt("Enter new Address:", userProfile.address);

    if (a !== null) userProfile.address = a.trim();

    const y = await showPrompt("Enter new Year Level:", userProfile.yearLevel);

    if (y !== null) userProfile.yearLevel = y.trim();

    const c = await showPrompt("Enter new Contact Number:", userProfile.contact);

    if (c !== null) userProfile.contact = c.trim();

    const userRef = doc(db, "users", userProfile.uid);

    await setDoc(userRef, userProfile, { merge: true });

    renderProfile();

    showAlert("Profile updated successfully!");

    // Update the profile in Firestore

    if (auth.currentUser) {

      const userRef = doc(db, "users", auth.currentUser.uid);

      try {

        await setDoc(userRef, userProfile, { merge: true }); // Use merge: true to only update specified fields

        console.log("User profile updated in Firestore.");

      } catch (error) {

        console.error("Error updating user profile in Firestore:", error);

      }

    }

  });

});