using System;

using System.Data;

using System.Data.SqlClient;

using System.Data.SQLite;

using System.Security.Cryptography;

using System.Text;

using System.Windows.Forms;

namespace Group\_Project\_Draft

{

public partial class Create : Form

{

// Path to store the SQLite database in the Debug folder

private string connectionString = "Data Source=UserDatabase.db;Version=3;";

public Create()

{

InitializeComponent();

CreateDatabase(); // Ensures the database and table exist

cmbTeam.SelectedIndexChanged += cmbTeam\_SelectedIndexChanged;

}

// Create the database file and Users table if they don't exist

private void CreateDatabase()

{

using (SQLiteConnection conn = new SQLiteConnection(connectionString))

{

conn.Open();

string query = @"CREATE TABLE IF NOT EXISTS Users (

id INTEGER PRIMARY KEY AUTOINCREMENT,

username TEXT UNIQUE,

email TEXT,

password\_hash TEXT

);";

using (SQLiteCommand cmd = new SQLiteCommand(query, conn))

{

cmd.ExecuteNonQuery();

}

}

}

// Hash password

public static string HashPassword(string password)

{

using (SHA256 sha256 = SHA256.Create())

{

byte[] hashBytes = sha256.ComputeHash(Encoding.UTF8.GetBytes(password));

return BitConverter.ToString(hashBytes).Replace("-", "").ToLower();

}

}

// Insert user into database

private void createBtn\_Click(object sender, EventArgs e)

{

string username = nameBox.Text;

string email = emailBox.Text;

string password = passBox.Text;

string confirm = passBox2.Text;

string favoriteSport = sportBox.SelectedItem.ToString(); // Get selected sport

string selectedTeam = cmbTeam.SelectedItem != null ? cmbTeam.SelectedItem.ToString() : null; // Get selected LAX team

// Update the user's favorite sport in the database

UpdateFavoriteSport(username, favoriteSport, selectedTeam);

if (password != confirm)

{

MessageBox.Show("Passwords do not match!");

return;

}

string hashedPassword = HashPassword(password);

using (SQLiteConnection conn = new SQLiteConnection(connectionString))

{

conn.Open();

string query = "INSERT INTO Users (username, email, password\_hash, favorite\_sport, favorite\_team) VALUES (@username, @email, @password, @favoriteSport, @favoriteTeam)";

using (SQLiteCommand cmd = new SQLiteCommand(query, conn))

{

cmd.Parameters.AddWithValue("@username", username);

cmd.Parameters.AddWithValue("@email", email);

cmd.Parameters.AddWithValue("@password", hashedPassword);

cmd.Parameters.AddWithValue("@favoriteSport", favoriteSport);

cmd.Parameters.AddWithValue("@favoriteTeam", (object)selectedTeam ?? DBNull.Value); // Insert NULL if no team selected

try

{

cmd.ExecuteNonQuery();

MessageBox.Show("Account created successfully!");

this.Close();

}

catch (SQLiteException ex)

{

MessageBox.Show("Error: " + ex.Message);

}

}

}

}

private void UpdateFavoriteSport(string username, string favoriteSport, string favoriteTeam)

{

string connectionString = "Data Source=UserDatabase.db;Version=3;";

using (SQLiteConnection conn = new SQLiteConnection(connectionString))

{

conn.Open();

string query = "UPDATE Users SET favorite\_sport = @favoriteSport, favorite\_team = @favoriteTeam WHERE username = @username";

using (SQLiteCommand cmd = new SQLiteCommand(query, conn))

{

cmd.Parameters.AddWithValue("@favoriteSport", favoriteSport);

cmd.Parameters.AddWithValue("@favoriteTeam", (object)favoriteTeam ?? DBNull.Value);

cmd.Parameters.AddWithValue("@username", username);

cmd.ExecuteNonQuery();

}

}

}

private void cmbTeam\_SelectedIndexChanged(object sender, EventArgs e)

{

if (sportBox.SelectedItem != null && sportBox.SelectedItem.ToString() == "Lacrosse")

{

cmbTeam.Visible = true; // Show the team dropdown

LoadLacrosseTeams(); // Load PLL teams

}

else

{

cmbTeam.Visible = false; // Hide if another sport is selected

cmbTeam.Items.Clear();

}

}

private void LoadLacrosseTeams()

{

string connectionString = "Data Source=(localdb)\\MSSQLLocalDB;Initial Catalog=PllStats;Integrated Security=True;";

using (SqlConnection conn = new SqlConnection(connectionString))

{

conn.Open();

string query = "SELECT DISTINCT TeamName FROM dbo.[pll-team-stats]";

using (SqlCommand cmd = new SqlCommand(query, conn))

{

SqlDataReader reader = cmd.ExecuteReader();

cmbTeam.Items.Clear();

while (reader.Read())

{

cmbTeam.Items.Add(reader["TeamName"].ToString());

}

}

}

}

private void backBtn\_Click(object sender, EventArgs e)

{

this.Close();

}

}

}