

Střední průmyslová škola elektrotechnická

Programování a aplikace počítačů

Ječná 30, Praha 2

Email Client Application with MSSQL Server Integration

Richard Bílek

Programové vybavení

2022/2023

Cíl práce

Cílem této práce je vytvoření klientské aplikace, ve které bude moci uživatel komunikovat obdobně jako e-mailem s ostatními zaregistrovanými uživateli. Program využívá Windows Forms, nejedná se o konzolovou aplikaci. Aplikace bude umět odesílat a přijímat zprávy, mazat příchozí a odchozí poštu.

Minimální hardwarové požadavky

Procesor: x86 nebo x64.

RAM: 512 MB (minimum), 1 GB (doporučeno)

Pevný disk: 400 MB volného místa

Software

Tento program byl vyvíjen v JetBrains Rider.

Ke spuštění je vyžadován operační systém Windows.

Program

Metodu `GetUserId()` jsem využil při přihlašování nebo registraci, abych zjistil, zda přihlašovaný uživatel je zaregistrovaný, nebo jestli není uživatelské jméno již obsazené.

```
public int GetUserId(string username)
{
    var conn:SqlConnection = DatabaseSingleton.GetInstance();
    var id = 0;

    using (var command = new SqlCommand(cmdText: "GetIdByName", conn))
    {
        command.CommandType = CommandType.StoredProcedure;
        command.Parameters.AddWithValue(parameterName: "@username", username);

        using (var reader = command.ExecuteReader())
        {
            if (reader.Read())
            {
                id = Convert.ToInt32(reader["id"]);
            }
        }
    }

    return id;
}
```

Obrázek 1: Metoda `GetUserId()`

Dále jsem při přihlašování využil metodu `PasswordMatches()`, která vrací informaci, zda se zadané heslo shoduje se zahashovaným heslem v databázi.

```
public bool PasswordMatches(User u)
{
    string receivedPassword;
    var conn:SqlConnection = DatabaseSingleton.GetInstance();
    using (var command = new SqlCommand(cmdText: "SELECT Password FROM Accounts WHERE Username = @name", conn))
    {
        command.Parameters.Add(new SqlParameter("@name", u.Username));
        using (var reader = command.ExecuteReader())
        {
            if (reader.Read())
            {
                receivedPassword = reader[0].ToString();
            }
            else
            {
                return false;
            }
        }
    }

    return PasswordEncryption.VerifyPassword(u.Password, hashedPassword: receivedPassword);
}
```

Obrázek 2: Metoda `PasswordMatches()`

O samotné porovnávání se stará metoda `VerifyPassword()`. Ta porovnávané heslo zahashuje úplně stejně jako heslo v databázi podle algoritmu Aragon2id.

```

public static bool VerifyPassword(string password, string hashedPassword)
{
    // Extract the salt value and hashed password from the stored hashed password
    string[] parts = hashedPassword.Split(params separator: '.');
    byte[] salt = Convert.FromBase64String(parts[0]);
    byte[] hashedPasswordBytes = Convert.FromBase64String(parts[1]);

    // Hash the input password with the extracted salt using the Argon2id algorithm
    Argon2id argon2 = new Argon2id(password: Encoding.UTF8.GetBytes(password));
    argon2.Salt = salt;
    argon2.DegreeOfParallelism = 8; // Number of threads to use
    argon2.MemorySize = 8192; // Amount of memory to use in kibibytes
    argon2.Iterations = 4; // Number of iterations

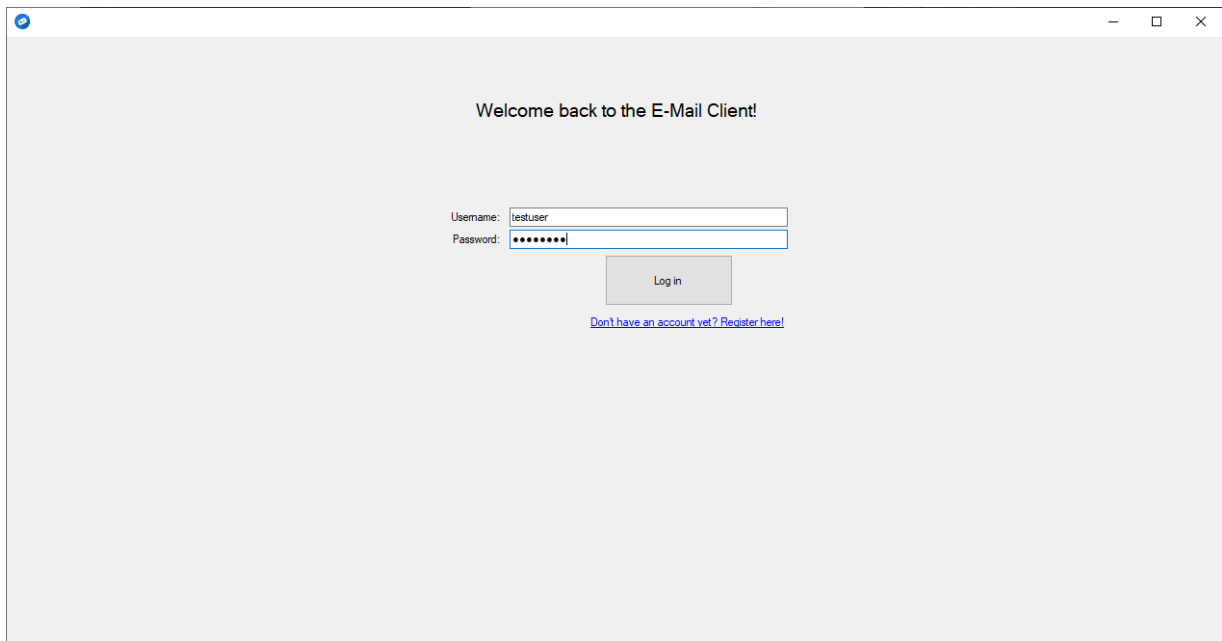
    byte[] hashedInputPassword = argon2.GetBytes(bc: 32); // Hash length of 32 bytes

    // Compare the hashed input password with the stored hashed password
    return StructuralComparisons.StructuralEqualityComparer.Equals(hashedPasswordBytes, hashedInputPassword);
}

```

Obrázek 3:Metoda VerifyPassword()

Po spuštění se zobrazí přihlašovací obrazovka, ve které se můžete přihlásit, nebo přejít na registraci.



Welcome back to the E-Mail Client!

Username:

Password:

[Don't have an account yet? Register here!](#)

Obrázek 4: Přihlašovací obrazovka

Register your account

Username:

Password:

Repeat password:

[Already have an account? Log in!](#)

Obrázek 5: Registrační obrazovka

Po přihlášení se zobrazí obrazovka s příchozí poštou.

Inbox:

	Sender	Send Date	Subject
▶	testuser	08.06.2023 16:21	Dobry den

Log Out

Recently logged in users

	Username	Last Online
▶	testuser	08.06.2023 17:24
	kokos	
	asir1	
	flip	
	kukelka	

Obrázek 6: Příchozí pošta

Závěr

Tento program je celý moje práce, pouze metodu na šifrování hesel jsem vytvořil s velkou podporou ChatGPT. Při vytváření programu jsem se nesetkal s žádným problémem. Program by neměl obsahovat žádné chyby a je plně funkční.

Seznam obrázků

Obrázek 1: Metoda GetUserId()	5
Obrázek 2: Metoda PasswordMatches()	5
Obrázek 3: Metoda VerifyPassword()	6
Obrázek 4: Přihlašovací obrazovka	6
Obrázek 5: Registrační obrazovka	7
Obrázek 6: Příchozí pošta	7
Obrázek 7: Otevřený přijatý e-mail	8
Obrázek 8: Příchozí pošta po odstranění e-mailu	8