Asztali alkalmazás

Az alkalmazás egy .NET framework alapú Windows Forms applikáció.

Adatbázis/API kommunikáció

Az asztali alkalmazást az alkalmazottak használják az adatbázisban szereplő adatok kezelésére, így egy egyszerű gyors és biztonságos módot kell használni az adatbázissal való kommunikációra. Erre a feladatra az API-t használunk, ami az összes lekérdezést, hozzáadást, módosítást és törlést elvégzi, ami az asztali alkalmazásnak kell. Az API-val való kommunikációért az „ApiComm.cs” osztály végzi aminek csak egyetlen statikus „SendPost” metódusa van.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Net.Http;

using System.Windows.Forms;

using System.ComponentModel;

using System.Web.Script.Serialization;

namespace Desktop

{

internal class ApiComm

{

public static async Task<object> SendPost(Dictionary<string, string> args)

{

var client = new HttpClient();

//var content = new FormUrlEncodedContent(args);

string parameters = "";

foreach(KeyValuePair<string, string> line in args)

{

parameters += $"{line.Key}={line.Value}&";

}

parameters.Remove(parameters.Length - 1);

var content = new StringContent(parameters, Encoding.UTF8, "application/x-www-form-urlencoded");

var response = await client.PostAsync("http://localhost:8000/BACKEND/api.php", content);

var responseString = await response.Content.ReadAsStringAsync();

JavaScriptSerializer serializer = new JavaScriptSerializer();

var responseDir = serializer.Deserialize(responseString, typeof(List<Dictionary<string, string>>));

client.Dispose();

return responseDir;

}

}

}

Ez a metódus egy „Dictionary<string, string>” típusú szótárat kér argumentumként, ami a POST kérés változóit és azok értékeit tartalmazza. A kérés megformálásáért NEM a FormUrlEncodedContent metódus felelős, mert kép feltöltése esetén túl nagyok voltak a paraméterek. Ennek orvoslása végett a POST request-et a metódus rakja össze magának, így csak a XAMPP beállításai korlátozzák egy POST kérés méreteit. A kérés formátuma és visszatérési értékei a fejlesztői dokumentáció adatbázis részénél található. Választ JSON formátumban kap, amit dekódol és visszaküldi „List<Dictionary<string,string>> ” típusú formátumban.

Kinézeti elemek

Az alkalmazásban található összes bemeneti mező saját fejlesztés nem az alapértelmezett verzió. Elsősorban a kinézet testreszabhatósága végett készültek ezek az elemek.

Bejelentkezés

A program indításakor a „loginForm.cs” lesz meghívva a felhasználó bejelentkeztetése mellet ez tárolja a bejelentkezett alkalmazott adatait, illetve az alkalmazás MainFrom-jának a referenciáját is itt lehet találni.

Főoldal

Bejelentkezést követően a „MainForm.cs” mindig látható lesz, itt található a menü és a többi form is ezen belül fog megnyílni az „OpenChildForm” metódus segítségével. A felhasználói élmény növelésének érdekében a leggyakrabban használt kilistázó form-okat nem bezárjuk, hanem csak eltüntetjük, hogy a keresési szűrő megmaradjon és így kényelmesebb legyen a használat.

public void OpenChildForm(Form childForm)

{

if (activeForm != null)

{

if (activeForm == usersPage || activeForm == allBooksPage)

{

activeForm.Hide();

}

else

{

activeForm.Close();

}

}

prev = activeForm;

roundedCornerPanel\_PageContainer.Controls.Remove(activeForm);

activeForm = childForm;

childForm.TopLevel = false;

childForm.FormBorderStyle = FormBorderStyle.None;

childForm.Dock = DockStyle.Fill;

roundedCornerPanel\_PageContainer.Controls.Add(childForm);

roundedCornerPanel\_PageContainer.Tag = childForm;

ThemeManager.ApplyThemeToAllForms();

childForm.BringToFront();

childForm.Show();

childForm.Visible = true;

}

Kilistázó formok

Ebbe a kategóriába tartozik a „AllBooksPage.cs”, „BookingsPage.cs”, „BorrowingsPage.cs”, „PublishersPage.cs”, „ReservationsPage.cs”, „UsersPage.cs” és gyakorlatilag a „PopupSelect.cs” is de arról egy másik pontban részletesebben is lesz írva. Az összes ilyen form gyakorlatilag az adatbázis valamelyik tábláját jeleníti meg egy datagridview-ben és onnan lehet további műveleteket végezni specifikus sorokkal, hozzáadni újat, meglévőt módosítani.