

# Dunaújvárosi Egyetem Bánki Donát Technikum

## Projekt feladat dokumentáció

Projekt leírása: .....	1
Technikai részletek: .....	2
Főbb funkciók és technikai megvalósítás: .....	2
Az alkalmazás: .....	3
Adatbázis-kezelés és MySQL Architektúrája: .....	3
Továbbfejlesztési lehetőségek (Future Improvements) .....	4
Önreflexió .....	5

Tantárgy neve: Programfejlesztés

Projekt tervező: Szabó Dávid Róbert

Projekt címe: Bűnügyi nyilvántartó

Dátum: 2026.01.15.

## Projekt leírása:

Egy olyan Windows-alapú asztali alkalmazás fejlesztése (C# .NET / Windows Forms környezetben), amely képes strukturált adatok fogadására és azok tartós tárolására egy külső **MySQL adatbázisban**. A program célja a büntetés-végrehajtási adminisztráció digitalizálása és gyorsítása.

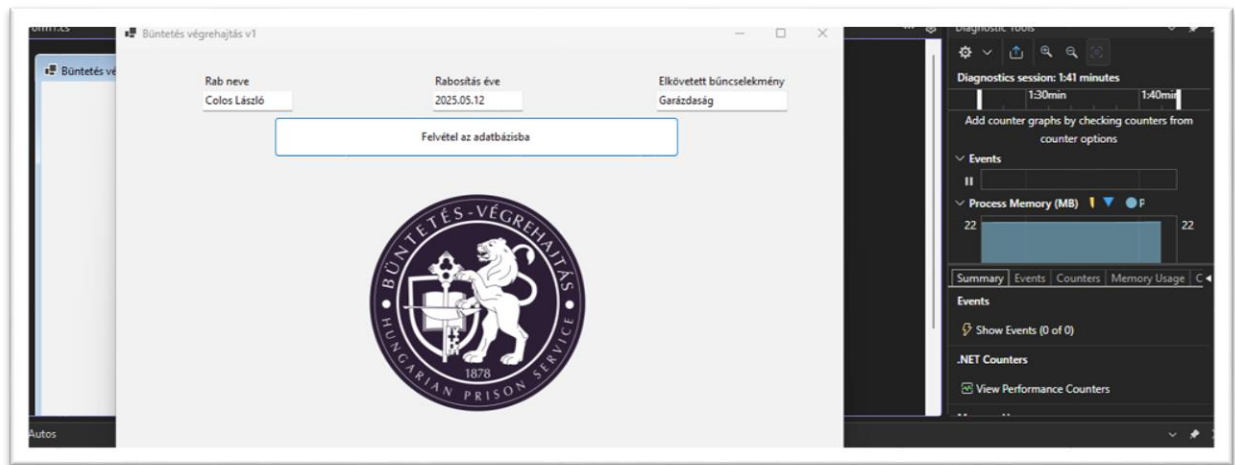
## Technikai részletek:

- **Fejlesztőkörnyezet:** Visual Studio, Windows Forms Application (C#).
- **Adatkezelés:** MySQL hálózati kapcsolat, `MySqlConnection` és `MySqlCommand` osztályok használatával.
- **Működés:** A felhasználó által megadott sztringeket és számértékeket a program SQL-parancsokká fűzi össze, majd aszinkron módon továbbítja a MySQL szervernek.

## Főbb funkciók és technikai megvalósítás:

- **Grafikus felhasználói felület (GUI):** A felhasználóbarát felületen három beviteli mező (TextBox) található a legfontosabb adatok számára: *Rab neve*, *Rabosítás éve*, és az *Elkövetett bűncselekmény*.
- **Adatbázis-kapcsolat:** A program egy MySQL szerverhez csatlakozik, ahová SQL lekérdezések (INSERT INTO) segítségével továbbítja a bevitt adatokat.
- **Adatvalidáció:** A fejlesztés során kiemelt figyelmet fordítottam arra, hogy csak érvényes adatok kerülhessenek a rendszerbe (pl. az évszámnak számnak kell lennie).
- **Eseményvezérelt programozás:** A "Felvétel" gomb megnyomásakor a szoftver automatikusan lekezeli a szerverkapcsolatot, végrehajtja az írást, és visszajelzést küld a felhasználónak a sikeres rögzítésről.

Az alkalmazás:



# Adatbázis-kezelés és MySQL Architektúrája:

**A modul célja:** A szoftver hátterét egy relációs adatbázis-kezelő rendszer (RDBMS) biztosítja. A cél az volt, hogy a program által kezelt adatok (rabok adatai) ne vesszenek el a program bezárása után, hanem strukturáltan, bármikor lekérdezhető formában tárolódjanak egy külső szerveren.

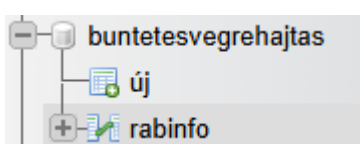
## Az adatbázis felépítése:

- **Adatbázis neve:** buntetesvegrehajtas.
- **Tábla neve:** rabinfo.
- **Mezők struktúrája:**
  - rabnev: A személy teljes neve (szöveges típus).
  - rabev: A rabosítás éve vagy ideje (számszerű vagy szöveges adat).
  - rabbunc: Az elkövetett bűncselekmény megnevezése.

## Technikai megvalósítás:

- **Kapcsolódás:** A Visual Studio és a MySQL közötti kommunikációt egy dedikált Connection String biztosítja, amely tartalmazza a szerver elérhetőségét és a belépési adatokat.
- **Adatátvitel:** A "Felvétel" gomb megnyomásakor a szoftver egy `INSERT INTO` parancsot generál, amely a Windows Forms mezőibe beírt adatokat (pl. "Colos László", "Garázdaság") beküldi a `rabinfo` táblába.
- **Adatintegritás:** A MySQL Workbench (vagy phpMyAdmin) felületén ellenőrizhető, hogy az adatok pontosan, oszlopokra bontva jelentek meg, biztosítva a későbbi kereshetőséget és listázhatóságot.

rabnev	rabev	rabbuncs
Colos László	5	
Colos László	2025.05.12	Garázdaság



## Továbbfejlesztési lehetőségek (Future Improvements)

Bár a jelenlegi alkalmazás stabilan ellátja az alapvető adatfelvételi funkciókat, a jövőben az alábbi modulokkal tervezem bővíteni a rendszert a professzionálisabb felhasználás érdekében:

- **CRUD funkciók teljessé tétele:** A jelenlegi „Create” (létrehozás) mellé beépíteném a „Read” (megjelenítés listában), „Update” (adatok módosítása) és „Delete” (törlés) funkciókat, hogy egy teljes körű adatbázis-kezelőt kapjunk.
- **Keresés és szűrés:** Egy keresőmező implementálása, amellyel név vagy a bűncselekmény típusa alapján azonnal szűrhetővé válnának a rögzített rekordok.
- **Felhasználói jogosultságkezelés:** Egy biztonságos bejelentkező felület (Login System) létrehozása, ahol különböző jogosultsági szintek (adminisztrátor, írnok) határoznák meg, ki férhet hozzá az adatokhoz.
- **Adatbiztonság fokozása:** Az SQL Injection elleni védelem megerősítése „Prepared Statements” használatával, illetve az adatbázis-jelszavak titkosított tárolása.
- **Exportálási lehetőségek:** Olyan funkció hozzáadása, amellyel a nyilvántartott adatok PDF vagy Excel formátumba exportálhatók a hivatalos jelentések elkészítéséhez.
- **Modernizált UI/UX:** A Windows Forms felület lecserélése WPF (Windows Presentation Foundation) technológiára, ami modernebb megjelenést és skálázhatóbb grafikai elemeket tenne lehetővé.

## Önreflexió

A programfejlesztés során a Windows Forms App segítségével megtanultam, hogyan építsek fel egy működő grafikus felületet, de a legfontosabb tapasztalatot a külső adatbázis-kapcsolat kezelése adta. Izgalmas volt látni, ahogy a Visual Studio-ban megírt kódom hatására a fizikai adatok ténylegesen megjelentek a MySQL táblákban. Megértettem a változók típusának fontosságát és azt, hogy mekkora fegyelmet igényel a szintaktikai pontosság, különösen az SQL-lekérdezések összefűzésekor. Ez a projekt segített átlátni a szoftverek háttérfolyamatait, és büszke vagyok rá, hogy egy valódi adminisztrációs problémára alkottam megoldást.