

Dokumentace projektu Athletics Manager

Autor: Ondřej Krejčí

Kontakt: krejci3@spsejecna.cz

1. Úvod a Specifikace požadavků

1.1 Resumé projektu

Athletics Manager je desktopová aplikace vytvořená na platformě .NET 8 (WPF), která slouží k evidenci atletů, klubů, závodů a jejich výsledků. Aplikace využívá relační databázi MS SQL Server pro persistenci dat. Cílem projektu je usnadnit správu atletických statistik, umožnit hromadný import dat a generovat žebříčky výkonnosti.

1.2 Specifikace požadavků (Use Case)

Aplikace pokrývá následující případy užití pro roli Administrátora/Uživatele:

- Správa Klubů a Atletů:** Registrace nového klubu, přidání atleta, úprava údajů atleta (včetně deaktivace).
- Evidence Výsledků:** Zadávání výsledků závodů (čas, vzdálenost, body) s vazbou na disciplínu a soutěž.
- Hromadný Import:** Import atletů a soutěží ze strukturovaných souborů JSON s ošetřením chyb.
- Reporty a Statistiky:** Zobrazení statistiky klubů (počty medailí) a generování žebříčku TOP 100 výkonů.
- Export:** Export žebříčku do souboru JSON.

2. Architektura Aplikace

Aplikace je navržena jako třívrstvá (Layered Architecture):

- Prezentační vrstva (UI):** WPF okna (XAML + Code-behind) zajišťující interakci s uživatelem (MainWindow, AddAthleteWindow, TopPerformancesWindow).
- Logická/Datová vrstva (Repositories):** Třídy zapouzdřující SQL dotazy a logiku (AthleteRepository, ResultRepository). Zde se řeší transakce a mapování dat na objekty.
- Databázová vrstva:** MS SQL Server s uloženými procedurami, pohledy a funkcemi.

V aplikaci je využit návrhový vzor **Singleton** pro správu připojení k databázi (`DatabaseSingleton`), což zajišťuje efektivní sdílení connection stringu napříč aplikací.

3. Chování Aplikace (Behaviorální model)

Klíčovým procesem aplikace je **Import Dat**, který musí zajistit integritu databáze.

Popis procesu Importu (Activity Diagram flow):

1. Uživatel vybere JSON soubor.
2. Aplikace parsuje soubor.
3. Pro každou položku se zahájí validace (kontrola existence regionu).
4. Pokud jsou data validní -> Spustí se **databázová transakce**.
5. Uložená procedura zkontroluje duplicitu.
6. Vloží se Klub (pokud neexistuje) -> Vloží se Atlet.
7. Commit transakce.
8. V případě chyby -> Rollback a logování chyby.

4. Databázový Model (E-R Diagram)

Databáze je relační a normalizovaná. Názvy tabulek a atributů odpovídají anglické konvenci.

Hlavní entity a atributy:

- **Region:** RegionID (PK), Name (String).
- **Club:** ClubID (PK), Name (String), RegionID (FK).
- **Athlete:** AthleteID (PK), FirstName, LastName, BirthDate (Date), Gender (Enum M/F), IsActive (Bool), ClubID (FK).
- **Competition:** CompetitionID (PK), Name, Date, Type (Enum Indoor/Outdoor).
- **Discipline:** DisciplineID (PK), Name, UnitType (Enum Time/Meters/Points).
- **Result:** ResultID (PK), Performance (Float), AthleteID (FK), CompetitionID (FK), DisciplineID (FK).

Datové typy:

- V aplikaci jsou zastoupeny všechny povinné typy:

- **Reálné číslo:** Performance (decimal), Wind (float).
- **Logická hodnota:** IsActive (bit).
- **Výčet:** UnitType (Time, Meters...), Gender (M/F).
- **Řetězec:** Name, Note.
- **Datum:** BirthDate, CompetitionDate.

5. Schémata Importu a Exportu

Aplikace podporuje formát **JSON** pro přenos dat.

5.1 Import Atletů (JSON)

Povinné položky: FirstName, LastName, ClubName, ClubRegion.

Nepovinné položky: IsActive (default true).

JSON

```
[  
 {  
   "FirstName": "Jan",  
   "LastName": "Novák",  
   "BirthDate": "2000-01-01T00:00:00",  
   "Gender": "M",  
   "IsActive": true,  
   "ClubName": "Dukla Praha",  
   "ClubRegion": "Hlavní město Praha"  
 }  
 ]
```

5.2 Export TOP 100 Výkonů (JSON)

Aplikace generuje výstupní soubor s následující strukturou:

JSON

```
[  
 {  
   "Discipline": "100 m",  
   "Performance": 9.58,  
   "AthleteName": "Usain Bolt",  
   "CompetitionName": "World Championships",  
   "CompetitionDate": "2009-08-16T00:00:00"  
 }  
 ]
```

6. Konfigurace Aplikace

Veškerá konfigurace je uložena v souboru **App.config** (ve výsledném buildu AthleticsManager.dll.config).

Klíčové konfigurační volby:

Uživatel musí nastavit sekci appSettings pro připojení k databázi:

- **DataSource**: Adresa serveru (např. localhost nebo .\SQLEXPRESS).
- **Database**: Název databáze (zvolený při instalaci).
- **Name**: Uživatelské jméno SQL účtu.
- **Password**: Heslo SQL účtu.

XML

```
<appSettings>  
  <add key="DataSource" value="127.0.0.1" />  
  <add key="Database" value="AthleticsDB" />  
  <add key="Name" value="sa" />  
  <add key="Password" value="student" />  
</appSettings>
```

7. Instalace a Spuštění

Podrobný návod je uveden v souboru **README.md** (nebo .txt).

Stručný postup:

1. Na MS SQL Serveru vytvořte prázdnou databázi.
2. Spusťte přiložený SQL skript install_database.sql.
 - *Poznámka:* Skript je uzavřen v **jedné transakci**. Pokud nastane chyba, žádná tabulka se nevytvoří.
3. Upravte App.config dle údajů vaší databáze.
4. Spusťte soubor AthleticsManager.exe.

8. Použité knihovny třetích stran

Projekt využívá následující knihovny:

1. **System.Data.SqlClient:** Pro komunikaci s MS SQL Serverem (ADO.NET).
2. **System.Text.Json:** Pro serializaci a deserializaci JSON souborů (Import/Export).
3. **WPF (Windows Presentation Foundation):** Framework pro tvorbu UI (součást .NET).

9. Závěr

Projekt Athletics Manager splnil všechny stanovené cíle. Byla vytvořena funkční databázová aplikace umožňující kompletní správu atletických dat.

Klíčové přínosy:

- Robustní databázový návrh s využitím **View** a **Stored Procedures**.
- Bezpečné operace s daty díky použití **Transakcí** (zejména při importu a instalaci).
- Uživatelsky přívětivé rozhraní s možností filtrace a exportu dat.
- Rozšiřitelnost díky dodržení principů OOP a oddělení vrstev.

Aplikace je připravena k nasazení a splňuje požadavky na školní projekt včetně dokumentace a ošetření chybových stavů.