

Správa autopůjčovny

Matěj Halmich

Matejhalmich06@gmail.com

14.01.2025

Střední průmyslová škola elektrotechnická, Praha 2, Ječná 30

Tato dokumentace je součástí školního projektu

Konfigurační soubor .env

```
DB_SERVER=  
DB_NAME=  
DB_USER=  
DB_PASS=
```

Aplikace využívá konfigurační soubor .env do kterého uživatel vyplní údaje do svého Microsoft SQL Serveru. (Správné vyplnění je nezbytné pro správné fungování aplikace.)

DB_SERVER= ip adresa uživatelova serveru

DB_NAME= název databáze do které uživatel naimportoval databázi

DB_USER= název uživatele který se k databázi připojuje

DB_PASS= heslo které uživatel používá na svém Microsoft SQL Serveru

Upozornění:

Pokud zadáte nesprávné údaje aplikace se sice spustí, ale nebude možné pracovat s databází.

Specifikace požadavků na uživatele

Funkce aplikace:

Správa zákazníků:

- Umožňuje přidávání, úpravu, mazání a zobrazení zákazníků.
- Aplikace validuje vstupy, jako je správnost formátu e-mailu a kontrola věku (minimálně 18 let).

Správa výpůjček:

- Přidávání a mazání výpůjček.
- Kontrola vstupních dat, například že datum vrácení nemůže být dřívější než datum vypůjčení a cena nemůže být záporná.

Import dat:

- Import zákazníků a aut z CSV souborů.

Generování reportů:

- Možnost generování reportů do textového souboru se souhrnnými informacemi o zákaznících, autech a výpůjčkách.

Uživatelské rozhraní:

- Aplikace používá winforms pro jednoduché a přehledné ovládání.

Databáze:

- Microsoft SQL Server

Spuštění Aplikace

Důležité upozornění: Ke spuštění aplikace je nutné mít nainstalovaný .net

<https://dotnet.microsoft.com/en-us/>

Spouštění ve Visual studio:

- Zapněte svůj Ms SQL Server a vytvořte v něm databázi
- Do databáze nakopírujte soubor databaseImport.sql ze složky database_import
- USE [halmich] /*Změňte na databázi kterou používáte/*
- Celý skript executněte
- Do .env konfiguračního souboru vyplňte údaje k vašemu Ms SQL Serveru

```
DB_SERVER=  
DB_NAME=  
DB_USER=  
DB_PASS=
```

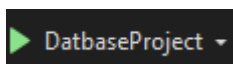
DB_SERVER= ip adresa uživatelského serveru

DB_NAME= název databáze do které uživatel naimportoval databázi

DB_USER= název uživatele který se k databázi připojuje

DB_PASS= heslo které uživatel používá na svém Microsoft SQL Serveru

- Klikněte ve Visual studio na spuštění aplikace



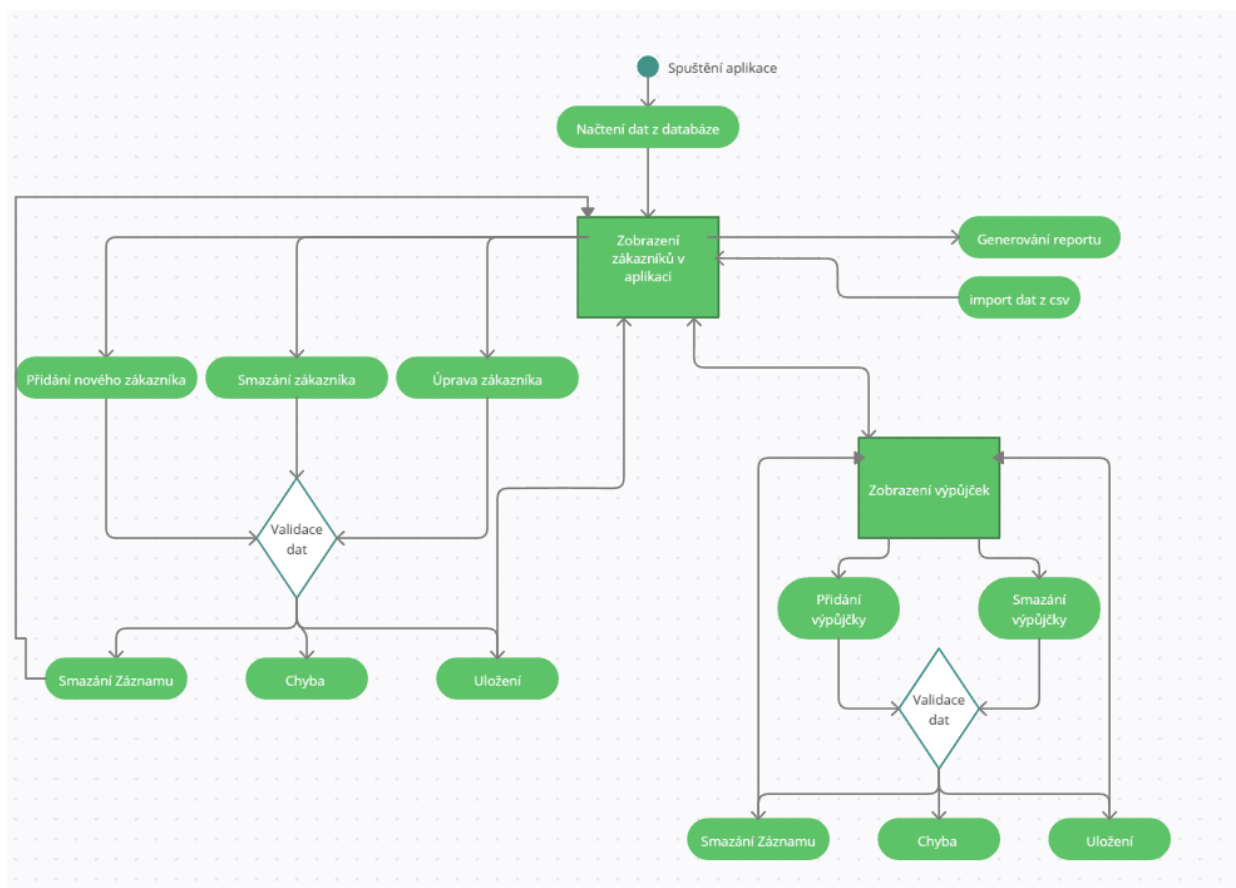
Uživatelské spuštění exe souboru:

- Otevřete složku databázový projekt
- Přejděte do složky bin\Release\net8.0-windows
- Spustíte program DatabaseProject.exe

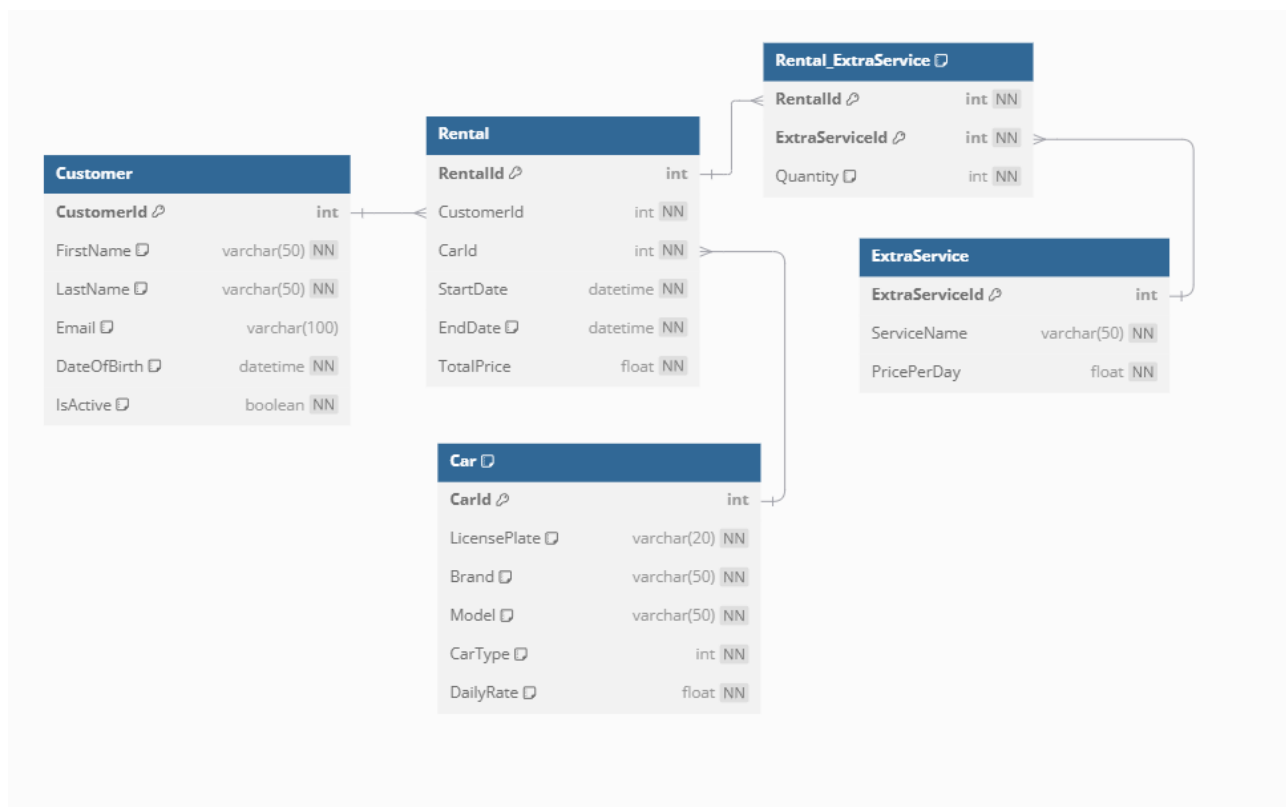
DatabazovyProjekt\DatbaseProject\DatbaseProject\bin\Release\net8.0-windows

Důležité upozornění: celá složka release využívá školní SQL server a proto mohou být data jiná než při spuštění na vlastním serveru ve Visual studio.

Popis projektu pomocí diagramů



Architektura databáze



Popis tabulek:

- **Customer:**
 - Uchovává informace o zákaznících (jméno, příjmení, e-mail, datum narození, aktivace).
- **Důležité omezení:**
 - E-mail musí být unikátní a validní.
 - Datum narození musí zajišťovat, že zákazník je starší 18 let.
- **Car:**
 - Obsahuje informace o dostupných autech (SPZ, značka, model, typ auta, denní sazba).
- **Důležité omezení:**
 - Značka i model musí být vyplněny.
- **Rental:**

- Zaznamenává informace o výpůjčkách (ID zákazníka, ID auta, datum začátku a konce výpůjčky, celková cena).
- **Důležité omezení:**
 - Datum ukončení výpůjčky musí být větší nebo rovno datu zahájení.
 - Cena výpůjčky nesmí být záporná a musí být číslo.
- **ExtraService:**
 - Uchovává data o doplňkových službách, jako je GPS nebo dětská sedačka.
- **Rental_ExtraService:**
 - Vazební tabulka mezi Rental a ExtraService (M:N relace). Uchovává počet jednotek dané služby využité během výpůjčky.

Chybové stavy

Aplikace zajišťuje:

- **Kontrolu při přidávání zákazníků:** Zákazník musí mít jméno, příjmení, email ve správném formátu. Více zákazníků nesmí mít stejný email. Zákazník musí být starší než 18 let. Pokud nejsou nějaké podmínky splněné vyhodí se chyba (program nespadne jen vyhodí chybovou hlášku) a záznam se nepřidá.
- **Kontrolu při úpravě zákazníků:** Nelze vymazat zákazníkovo nějaký údaj a nechat ho prázdný. Nelze upravit email na nějaký který již v databázi existuje. Nelze upravit datum narození aby byl zákazník mladší než 18 let. Pokud nejsou nějaké podmínky splněné vyhodí se chybová hláška a záznam se neupraví.
- **Kontrolu při mazání zákazníků:** Nelze vymazat zákazníka který má nějakou výpůjčku. Pokud budeme chtít vymazat zákazníka který má nějakou výpůjčku tak se vyhodí chybová hláška a záznam se nevymaže.
- **Kontrolu při importu dat do tabulek Car a Customer:** Pokud data v csv souboru nejsou správná nebo csv není ve správném formátu, tak se vyhodí chybová hláška a data se neimportují.
- **Kontrolu při přidávání výpůjčky:** Pokud se uživatel pokouší přidat výpůjčku a do ceny přidává záporné číslo a nebo znaky které nejsou číslo, tak se vyhodí chybová hláška a záznam se nepřidá.
- **Kontrolu při přidávání výpůjčky:** Pokud se uživatel pokouší přidat výpůjčku a do data vrácení dává menší datum než do data půjčení, tak se vyhodí chybová hláška a záznam se nepřidá.

Použité knihovny třetích stran

V projektu byly využity následující NuGet balíčky pro zajištění funkcionality:

1. **DotNetEnv** (verze 3.1.1)
 - Knihovna pro načítání a správu konfiguračních hodnot ze souboru .env.
 - Používá se pro správu citlivých údajů, jako jsou přihlašovací údaje k databázi (např. server, uživatelské jméno, heslo).
 - <https://github.com/tonerdo/dotnet-env>
2. **Microsoft.Data.SqlClient** (verze 5.2.2)
 - Oficiální klientská knihovna pro komunikaci s Microsoft SQL Serverem.
 - Používá se pro připojení aplikace k databázi a spouštění dotazů na SQL Server.
 - <https://github.com/dotnet/SqlClient>
3. **Microsoft.EntityFrameworkCore** (verze 9.0.0)
 - Hlavní balíček pro Entity Framework Core, který zajišťuje mapování objektů na databázové tabulky (ORM).
 - Umožňuje práci s databází v objektově orientovaném stylu, bez nutnosti psát přímo SQL dotazy.
 - <https://learn.microsoft.com/en-us/ef/core/>
4. **Microsoft.EntityFrameworkCore.SqlServer** (verze 9.0.0)
 - Rozšíření Entity Framework Core, které přidává podporu pro Microsoft SQL Server.
 - Obsahuje specifické funkce pro SQL Server, jako je práce s datovými typy a optimalizace dotazů.
 - <https://learn.microsoft.com/en-us/ef/core/>

Závěrečné resumé

1. Cíl projektu:

- Cílem projektu bylo vytvořit aplikaci pro správu autopůjčovny, která umožňuje efektivní správu zákazníků, výpůjček a vozového parku.

2. Hlavní funkce:

Aplikace umožňuje:

- Přidávat, upravovat a mazat záznamy zákazníků.
- Přidávat a mazat výpůjčky vozidel.
- Importovat dat z CSV souborů.
- Generovat souhrnný report o zákaznících, vozidlech a výpůjčkách.
- Validovat vstupy, aby se minimalizovaly chyby při práci s databází.

3. Přínosy projektu:

- Projekt poskytuje intuitivní uživatelské rozhraní, které usnadňuje správu autopůjčovny. Validace dat minimalizuje chyby a zlepšuje spolehlivost aplikace.

4. Použité technologie:

- **Programovací jazyk:** C# s využitím Windows Forms.
- **Databázi:** Microsoft SQL Server.
- **Knihovny:**
 - Microsoft.EntityFrameworkCore
 - Microsoft.EntityFrameworkCore.SqlServer
 - DotNetEnv
 - Microsoft.Data.SqlClient

7. Závěr:

- Projekt splnil všechny stanovené cíle a umožnil mi prohloubit znalosti v oblasti, databázových systémů a práce s Entity Framework Core. Aplikace je připravena na testování.

