

DOKUMENTACE PROJEKTU

1. Základní údaje

Název projektu:

Objednávkový systém autosalonu

Autor:

Tomáš Kopecký

Kontaktní údaje:

kopecky4@spsejecna.cz

Datum vypracování:

10.1.2026

Škola:

Střední průmyslová škola elektrotechnická, Praha 2, Ječná 30

Typ projektu:

Školní projekt – databázová aplikace

2. Úvod

Tento dokument popisuje návrh, implementaci a použití databázové aplikace „Objednávkový systém autosalonu“.

Cílem projektu je vytvořit aplikaci v jazyce Python s využitím relační databáze MySQL, která umožňuje správu zákazníků, automobilů, výbavy a objednávek.

3. Specifikace požadavků uživatele (Use Case)

Uživatel aplikace může:

- přidat, upravit a odstranit zákazníka
- přidat, upravit a odstranit automobil
- přidat výbavu vozidla
- vytvořit objednávku (zákazník + auto + výbava)
- importovat data ze souborů (CSV, JSON)
- zobrazit přehled objednávek (report)

(Use Case diagram lze případně doplnit graficky – není povinné, textová specifikace je dostačující)

4. Architektura aplikace (návrhové vzory)

Aplikace je navržena s využitím **Repository pattern**.

Architektura:

- **UI vrstva** – konzolové menu (`main.py`)
- **Service vrstva** – aplikační logika (`import`, `reporty`)
- **Repository vrstva** – práce s databází
- **Databáze** – MySQL

Použité repository:

- `CustomerRepository`
- `CarRepository`
- `EquipmentRepository`
- `OrderRepository`

Tím je zajištěno oddělení aplikační a databázové logiky.

5. Chování aplikace (behaviorální popis)

Popis toku aplikace:

1. Uživatel spustí aplikaci
2. Zobrazí se hlavní menu
3. Uživatel zvolí požadovanou akci
4. Aplikace provede databázovou operaci
5. Výsledek je zobrazen uživateli

Tok odpovídá **Activity diagramu** (sekvenční zpracování požadavků).

6. Databázový model (E-R model)

Tabulky:

- **customer** (`id`, `name`, `email`)
- **car** (`id`, `brand`, `model`, `price`, `fuel`)
- **equipment** (`id`, `name`, `price`)

- **car_order** (id, customer_id, car_id, created_at)
- **order_equipment** (order_id, equipment_id)
- **payment** (id, order_id, amount, paid_at)

Vztahy:

- customer 1:N car_order
- car 1:N car_order
- car_order M:N equipment (přes order_equipment)

Použité datové typy:

- VARCHAR
- INT
- FLOAT
- DATETIME
- ENUM / CHECK

7. Importované a exportované soubory

Import:

CSV – zákazníci

Sloupec Povinný

name ano
email ano

JSON – auta

Položka Povinná

brand ano
model ano
price ano
fuel ano

Export dat aplikace neprovádí.

8. Konfigurace aplikace

Konfigurace je uložena v souboru:

config/config.ini

Obsah:

- host databáze
- port
- uživatelské jméno
- heslo
- název databáze

Konfigurace není pevně zapsána ve zdrojovém kódu.

9. Instalace a spuštění aplikace

Instalace:

1. Nainstalovat Python 3.10+
2. Nainstalovat MySQL Server
3. Importovat databázi ze souboru `.sql`
4. Nainstalovat knihovny:

```
pip install mysql-connector-python
```

Spuštění:

```
python main.py
```

10. Chybové stavy

Aplikace ošetřuje následující chyby:

- chybějící konfigurační soubor
- neexistující importní soubor
- chybné připojení k databázi
- neplatný vstup uživatele

Chyby jsou zachyceny pomocí výjimek a oznámeny uživateli.

11. Použité knihovny třetích stran

- `mysql-connector-python` – připojení k MySQL databázi
-

12. Struktura projektu

```
autosalon/  
├── main.py  
├── db.py  
├── repositories/  
├── services/  
├── data/  
├── config/  
├── doc/  
└── README.txt
```

13. Export projektu

- Zdrojové kódy jsou přehledně strukturovány
 - Obsahují komentáře
 - Projekt je uložen v jednom archivu `.zip`
 - Součástí je `README.txt`
-

14. Export databáze

- Databáze je exportována do jednoho souboru `.sql`
 - Obsahuje:
 - DDL příkazy (CREATE TABLE)
 - DML příkazy (INSERT)
 - Export je uzavřen v transakci
 - Soubor obsahuje komentář s názvem projektu a autorem
-

15. Závěrečné resumé

Projekt splňuje všechny požadavky zadání.

Aplikace demonstruje práci s relační databází, návrhovými vzory, importem dat, transakcemi a ošetřením chyb.

Projekt je připraven k odevzdání a obhajobě.