

# Testovací scénář 1: Instalace a spuštění aplikace

**Projekt:** Library DB System (D1)

**Verze:** 1.0

**Datum:** 2025-01-11

---

## 1. Předpoklady

- Počítač s nainstalovaným:
    - Python 3.10 nebo novější
    - Microsoft SQL Server (nebo přístup k SQL Serveru)
    - Microsoft SQL Server ODBC Driver 17 for SQL Server
    - Git
- 

## 2. Stažení projektu

**Krok 1: Klonování repozitáře**

```
git clone <URL_REPOZITARE>
cd library-db-system
```

**Očekávaný výsledek:** Projekt je stažen do složky library-db-system.

---

## 3. Instalace závislostí

**Krok 2: Instalace Python knihoven**

```
pip install -r requirements.txt
```

**Očekávaný výsledek:**

- Flask a pyodbc jsou nainstalovány bez chyb - Výpis: Successfully installed  
flask pyodbc

---

## 4. Konfigurace aplikace

**Krok 3: Nastavení config.json**

Otevřete soubor config/config.json a vyplňte údaje:

```
{
  "server": "localhost",
  "database": "library",
  "username": "sa",
```

```
"password": "your_password",  
"driver": "ODBC Driver 17 for SQL Server"  
}
```

**Poznámka:** Pro školní PC obvykle: - server: "localhost" - database: "library"  
- username: "sa" - password: "student" (nebo jiné podle nastavení)

**Očekávaný výsledek:** Konfigurace je nastavena.

---

## 5. Inicializace databáze

### Krok 4: Spuštění setup scriptu

```
python setup.py
```

#### Očekávaný výsledek:

```
Database setup completed.
```

Script automaticky: - Vytvoří databázi (pokud neexistuje) - Spustí sql/ddl.sql  
(vytvoří tabulky) - Spustí sql/views.sql (vytvoří pohledy)

Pokud vše proběhlo úspěšně, zobrazí se zpráva "Database setup completed."

---

## 6. Spuštění aplikace

### Krok 5: Spuštění Flask aplikace

```
python src/app.py
```

#### Očekávaný výsledek:

```
* Serving Flask app 'app'  
* Debug mode: on  
* Running on http://0.0.0.0:5000
```

### Krok 6: Otevření aplikace v prohlížeči

Otevřete webový prohlížeč a přejděte na:

```
http://localhost:5000
```

**Očekávaný výsledek:** - Zobrazí se domovská stránka "Library Management System" - Menu obsahuje odkazy: - Books, Authors, Categories, Users, Loans - Import JSON - Reports

---

## 7. Test základní funkcionality

### Krok 7: Přidání kategorie

1. Klikněte na “Add Category”
2. Vyplňte:
  - Name: Science Fiction
  - Type: fiction
3. Klikněte “Add Category”

**Očekávaný výsledek:** - Zpráva: “Category added successfully!” - Kategorie se zobrazí v seznamu

### Krok 8: Přidání autora

1. Klikněte na “Add Author”
2. Vyplňte Name: Isaac Asimov
3. Klikněte “Add Author”

**Očekávaný výsledek:** - Zpráva: “Author added successfully!” - Autor se zobrazí v seznamu

### Krok 9: Přidání uživatele

1. Klikněte na “Add User”
2. Vyplňte:
  - Full Name: John Doe
  - Email: john@example.com
3. Klikněte “Add User”

**Očekávaný výsledek:** - Zpráva: “User added successfully!” - Uživatel se zobrazí v seznamu

---

## 8. Možné problémy a řešení

### Problém 1: “ODBC Driver not found”

**Řešení:** Nainstalujte Microsoft ODBC Driver 17 for SQL Server z: <https://docs.microsoft.com/en-us/sql/connect/odbc/download-odbc-driver-for-sql-server>

### Problém 2: “Login failed for user”

**Řešení:** - Zkontrolujte username a password v `config/config.json` - Zkuste použít Windows Authentication

**Problém 3: “Database not found”**

**Řešení:** - Spusťte `python setup.py` znovu - Zkontrolujte název databáze v `config/config.json`

**Problém 4: “ModuleNotFoundError: No module named ‘flask’”**

**Řešení:**

`pip install flask pyodbc`

**Problém 5: Port 5000 je obsazen**

**Řešení:** Změňte port v `src/app.py`:

`app.run(debug=True, host='0.0.0.0', port=5001)`

---

## 9. Ukončení aplikace

Pro zastavení aplikace stiskněte **Ctrl+C** v terminálu.

---

## 10. Závěr

Po úspěšném dokončení tohoto scénáře: - Aplikace je nainstalována -  
Databáze je vytvořena a nakonfigurována - Aplikace běží na `http://localhost:5000`  
- Základní funkce fungují (přidání kategorie, autora, uživatele)

**Tester:** \_\_\_\_\_

**Datum:** \_\_\_\_\_

**Výsledek:** PASS / FAIL

**Poznámky:**