

Testovací scenar 1: Instalace a spusteni aplikace

Projekt: E-Shop (D1 Repository Pattern) **Autor:** zajicek3 **Datum:** 2026

1. Predpoklady

Pred spustenim testu overit:

- ☐ Nainstalovany Python 3.10+
- ☐ Nainstalovany Node.js 18+
- ☐ Nainstalovany ODBC Driver 17 for SQL Server
- ☐ Pristup k SQL Server databazi
- ☐ Nainstalovany SQL Server Management Studio (SSMS)

2. Klonovani repozitare

Kroky:

- Otevrit terminal
- Spustit prikaz:

```
git clone <url-repozitare>
cd e-shop
```

Ocekavany vysledek:

- Slozka `e-shop` vytvorena
- Obsahuje soubory: `README.md` , `package.json` , `requirements.txt`

Mozne chyby:

Chyba	Reseni
<code>git: command not found</code>	Nainstalovat Git
<code>Repository not found</code>	Zkontrolovat URL

3. Nastaveni databaze

3.1 Pripojeni k SQL Serveru

Kroky:

- Otevrit SQL Server Management Studio
- Pripojit se k serveru:
 - Server: `<adresa-serveru>`
 - Authentication: SQL Server Authentication
 - Login: `<uzivatel>`
 - Password: `<heslo>`

3. Kliknout Connect

Ocekavany vysledek:

- Pripojeni uspesne
- V Object Explorer vidite server

Mozne chyby:

Chyba	Reseni
Login failed (18456)	Zkontrolovat heslo
Cannot connect to server	Zkontrolovat adresu serveru
Network error	Zkontrolovat internetove pripojeni

3.2 Vytvoreni tabulek

Kroky:

1. V SSMS kliknout pravym na databazi -> New Query
2. Otevrit soubor `src/sql/01_create_tables.sql`
3. Zkopirovat obsah do Query okna
4. Kliknout Execute (F5)

Ocekavany vysledek:

- Zprava: "Commands completed successfully"
- V databazi 5 novych tabulek: users, categories, products, orders, order_items

Mozne chyby:

Chyba	Reseni
Table already exists	Nejdrive smazat existujici tabulky
Foreign key error	Spustit skripty ve spravnem poradi

3.3 Vytvoreni views

Kroky:

1. Otevrit soubor `src/sql/02_create_views.sql`
2. Spustit v SSMS

Ocekavany vysledek:

- 2 nove views: v_order_details, v_product_stats

3.4 Naplneni testovacimi daty

Kroky:

1. Otevrit soubor `src/sql/03_seed_data.sql`
2. Spustit v SSMS

Ocekavany vysledek:

- 3 kategorie, 3 uzivatele, 6 produktu, 3 objednavky
-

4. Nastavení backendu

4.1 Vytvoření .env souboru

Kroky:

1. Zkopírovat `.env.example` na `.env`
2. Otevřít `.env` v textovém editoru
3. Vyplnit hodnoty:

```
DB_SERVER=<adresa-serveru>
DB_NAME=<nazev-databaze>
DB_USER=<uzivatel>
DB_PASSWORD=<heslo>
DB_DRIVER=ODBC Driver 17 for SQL Server
```

4. Uložit

Očekávaný výsledek:

- Soubor `.env` vytvořen s vašimi údaji

Možné chyby:

Chyba	Řešení
Soubor se neukládá	Zkontrolovat práva k zápisu

4.2 Instalace Python závislostí

Kroky:

1. Otevřít terminál v složce `e-shop`
2. Spustit:

```
pip install flask pyodbc python-dotenv flask-cors
```

Očekávaný výsledek:

- Všechny balíčky nainstalovány
- Žádné chyby

Možné chyby:

Chyba	Řešení
pip not found	Přidat Python do PATH
Permission denied	Spustit jako administrator

4.3 Spuštění backendu

Kroky:

1. V terminálu přejít do `src/backend` :

```
cd src/backend
```

2. Spustit:

```
python app.py
```

Ocekavany vysledek:

```
=====
E-SHOP API SERVER
=====
Server: http://127.0.0.1:5000
Database: <server>/<database>
=====
* Running on http://127.0.0.1:5000
```

Mozne chyby:

Chyba	Reseni
ModuleNotFoundError	Nainstalovat chybejici modul
Login failed	Zkontrolovat .env soubor
Port already in use	Zmenit FLASK_PORT v .env

4.4 Overeni backendu

Kroky:

1. Otevrit prohlizec
2. Prejit na: `http://localhost:5000/api/health`

Ocekavany vysledek:

```
{"database": "connected", "status": "ok"}
```

5. Nastaveni frontendu

5.1 Instalace Node.js zavislosti

Kroky:

1. Otevrit NOVY terminal v slozce `e-shop`
2. Spustit:

```
npm install
```

Ocekavany vysledek:

- Slozka `node_modules` vytvorena

- Zadné kritické chyby

5.2 Spustení frontendu

Kroky:

1. Spustit:

```
npm run dev
```

Očekavany výsledek:

```
▲ Next.js 16.x.x
- Local: http://localhost:3000
```

5.3 Overení frontendu

Kroky:

1. Otevřít prohlížeč
2. Přejít na: `http://localhost:3000`

Očekavany výsledek:

- Zobrazí se homepage E-Shopu
- Vidíte doporučené produkty
- Navigace funguje

6. Kontrolní seznam

Po úspěšné instalaci overit:

- ☐ Backend běží na portu 5000
- ☐ Frontend běží na portu 3000
- ☐ `/api/health` vrací "connected"
- ☐ `/api/products` vrací seznam produktu
- ☐ Homepage zobrazuje produkty
- ☐ Navigace funguje (Produkty, Kosík, Admin)

7. Ukončení

Zastavení serveru:

1. V terminálu backendu: `Ctrl+C`
2. V terminálu frontendu: `Ctrl+C`

Test PASSED pokud všechny kroky proběhly úspěšně.