

Checkpoint 2

EAR2020 - Hotel Reservation System

Popis aplikace a její struktury

Aplikace se je psaná v Javě (testováno s Javou 1.8 a 11) a používá springboot.

Součástí aplikace je pouze backend, frontend aplikace nemá, pro otestování je přiloženo několik souborů:

- InitData.sql
 - Příklad dat připravených k importu do DB. Není třeba importovat, i při připojení k nové databázi systém vytvoří potřebné tabulky (je třeba vytvořit přímo na DB prvotního admin uživatele, aby bylo možno aplikaci spravovat).
- EAR HotelReservationSystem - ADMIN.postman_collection.json
EAR HotelReservationSystem - CUSTOMER.postman_collection.json
 - Sestavy testovací dotazů určený pro program Postman. Některé obsahují přesná ID pro úpravy/smazání určitých objektů z databáze, jelikož jsou ID generována automaticky je pravděpodobné, že nebudou sedět. Před takovými kroky je ovšem nachystán krok s vypsáním potřebných údajů obsahujících i aktuální ID.

Dle informací v předchozím checkpointu (přiložen v repozitáři) jsou implementovány všechny funkce. Díky zpětné vazbě byly některé ovšem mírně modifikovány, či přidány.

REST endpointy mají dokumentaci.

Instalace aplikace

Aplikace je plně kompilovatelná a lze ji tedy spustit na jakémkoliv stroji.

Jak je zmíněno v popisu aplikace - při inicializaci naprosto nové databáze je potřeba v DB vytvořit první adminský účet.

Pro účely testování toto není nutné, neboť je již vytvořen (admin@admin.cz, heslo 12345).

Hesla ke všem předvytvořeným účtům jsou 12345.

Aplikace je defaultně připojena k této DB:

URL: kbss.felk.cvut.cz:5433/ear2020zs_5

username: ear2020zs_5

heslo: ear2020zs_5pass

Lessons learned

Naučit se tvorbu celé enterprise aplikace od začátku bylo velmi přínosné. Velmi dobře jsme si při tom vyzkoušeli tvorbu všech vrstev apod. A hlavně debuggování už velmi komplexní aplikace.

Jako největší překážka se nám jevil asi čas na implementaci, což bylo převážně způsobeno už 2. semestrem za dob pandemie a ne úplně ideálním odhadem časové náročnosti určitých částí.