

# Zpráva o projektu Music Ticket

Autoři: Ekaterina Gulina, Iurii Lebedev

# Obsah

<b>Obsah</b>	<b>1</b>
<b>Popis aplikace a její struktury</b>	<b>2</b>
Struktura	2
Testování pomocí REST dotazů	2
<b>Instalace</b>	<b>3</b>
<b>Získané zkušenosti</b>	<b>3</b>

# Popis aplikace a její struktury

Aplikace je určena pro realizaci portálu Music tickets. Cílem aplikace je umožnit uživatelům nakupovat vstupenky na různé hudební akce (koncerty a festivaly) a také prohlížet informací o samotných akcích a kapelech, které této akce provádí.

## Struktura

Aplikace je rozdělena do packages.

- **Package config** obsahuje konfigurační třídy pro aplikaci, persistence a zabezpečení.
- **Package dao** obsahuje třídy DAO vrstvy napojené na modely. Třídy DAO obsahují základní CRUD funkcionalitu.
- **Package exception** obsahuje třídy použitých v projektu výjimek.
- **Package model** obsahuje reprezentaci entit z databáze. Třídy obsahují pojmenované dotazy, ordering a složené primární klíče. Navíc v tomto package se nachází další **package dto**, reprezentující mapování tříd modelu pro REST dotazy (např. aby se v dotazu neposílali celé objekty).
- **Package rest** obsahuje controllery a reprezentuje REST rozhraní celé aplikace.
- **Package security** obsahuje třídy pro autentizaci a autentifikaci uživatelů aplikace.
- **Package service** obsahuje třídy servisní vrstvy aplikace.

Uvnitř hlavního package mt se nachází třída **MTApplication**, která slouží pro spuštění Spring aplikace.

Vedle složky java v projektu se nachází složka s základními testy (pokrývají část dao, model a service packages). REST rozhraní aplikace se testovalo pomocí testových REST dotazu v aplikaci Insomnia.

## Testování pomocí REST dotazů

Testování pomocí REST dotazů je uděláno přes aplikaci **Insomnia**. Testovací data se nachází v GitLabu projektu ve složce **Insomnia\_Test** v souboru **restTests**.

RestTests obsahuje 4 testy.

1. **Testování eventů** – nově vytvořený event má položky id a je uložen do DB.
2. **Testování vstupenek** – nově vytvořená vstupenka má položku event a je uložena do DB.

3. **Testování nakupování vstupenek** – existující registrovaný uživatel může si koupit vstupenku s id = 51 a test vrátí HTTP status 200 OK.
4. **Testování uložení koupené vstupenky do DB** – ověří že nově zakoupená vstupenka má položku user.

## Instalace

Aplikace nevyžaduje žádnou instalaci a může být spuštěna přímo z IntelliJ IDEA. Aplikace běží na localhostu na portu 8080.

Pokud by bylo nutné přihlášení do DB, údaje se nachází v souboru application.properties, který se nachází ve složce resources.

## Získané zkušenosti

### Ekaterina Gulina

Během této semestrální práce jsem se poprvé seznámila a použila Spring. Díky této SP jsem se vyzkoušela Spring, získala nové znalosti a zkušenosti. Naučila jsem se používat nástroje pro testování REST dotazů (konkrétně Insomnia) a celkově teď jim lépe rozumím. Chtěla bych do budoucna obohatit naši SP o grafickou stránku a udělat její frontend, bohužel, v rámci SP na to už nezbyl čas.

Líbilo se mi, že jsme samostatně volili témata SP a navrhovali aplikací.

### Iurii Lebedev

Pro mě byla to úplně první zkušenost nejen ve světě Enterprise Arketiktury ale i ve světě Spring Bootu. Nemůžu říct že práce byla snadná nebo lehká, ale zajímavá určitě byla.

Budování celé aplikace i když je bez frontendu je dlouhý a náročný proces. Rád bych mohl tu práci obohatit, ale to se už bude vnějším projektem mimo semestrální práci.

Líbili se mi momenty když ta aplikace se spouštěla bez chýb i problémů. Doufám že tento projekt nebude poslední, rád bych se v tom učil i dál.