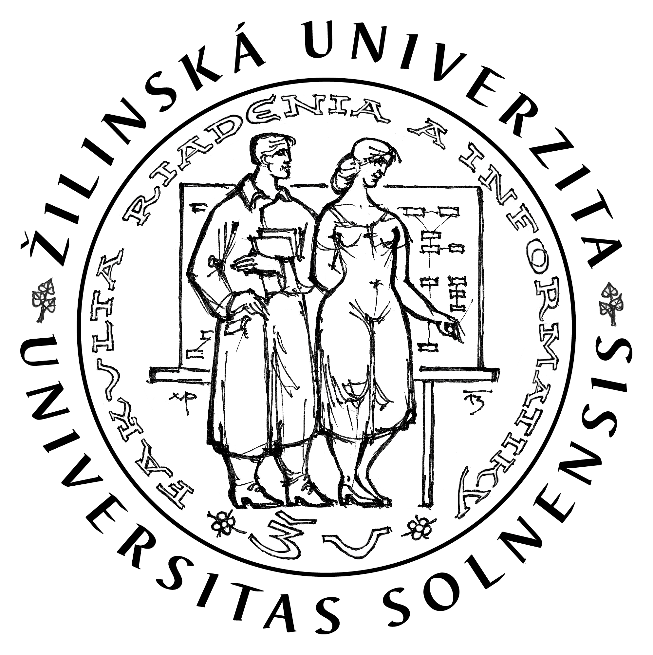
**ŽILINSKÁ UNIVERZITA**

FAKULTA RIADENIA A INFORMATIKY



POKROČILÉ DATABÁZOVÉ SYSTÉMY

SEMESTRÁLNA PRÁCA

Adrián Václavek 5ZIS11

Matej Majer 5ZIS11

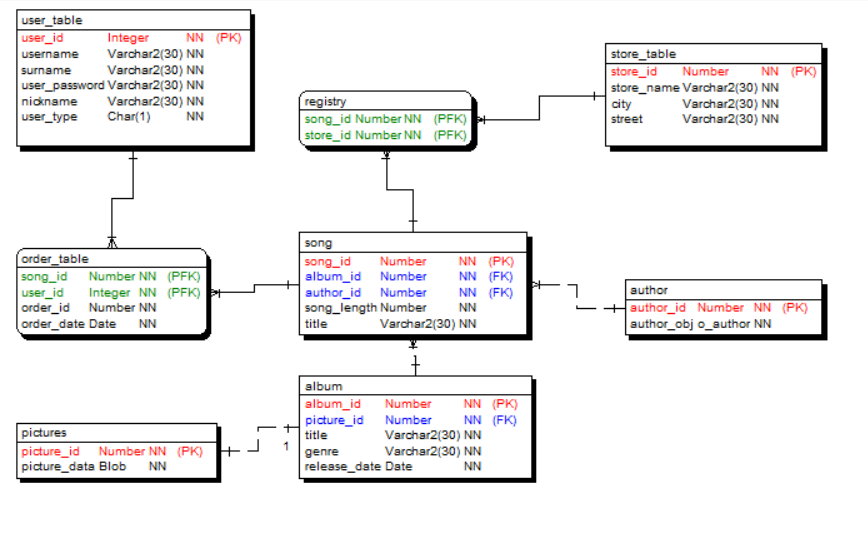
Karol Kojda 5ZIS11

2020/2021

# 1 Špecifikácia zadania

Cieľom našej semestrálnej práce bolo navrhnúť a implementovať informačný systém a navrhnúť dátový model zvolenej témy. Téma našej semestrálnej práce bol informačný systém pre obchod s hudbou. V našom systéme budeme evidovať pobočky obchodov s hudbou. Ďalej budeme evidovať databázu pesničiek, ktoré budú mať príslušných autorov, albumy a podobne.

# 2 Dátový model



Dátový model k našej semestrálnej práci je zložený z ôsmich tabuliek. Popis tabuliek:

**user\_table :** tabuľka, ktorá reprezentuje užívateľov, ktorý využívajú informačný systém.

* user\_id – primárny kľúč reprezentujúci unikátne identifikačné číslo užívateľa.
* username – krstné meno užívateľa.
* surname – priezvisko užívateľa.
* nickname – prihlasovacie meno užívateľa.
* user\_password – heslo pre prihlásenie užívateľa.
* user\_type – typ užívateľa, ktorý udáva, či je užívateľ zároveň adminom alebo nie.

**pictures :** tabuľka, ktorá reprezentuje obrázky priradené albumom.

* picture\_id – primárny kľúč s unikátnym číslom pre obrázok.
* picture\_data – konrétny obrázok.

**author :** tabuľka, ktorá reprezentuje autorov pesničiek. Táto tabuľka je zároveň tabuľka objektov, kde každý autor je objekt so svojimi atribútmi.

* author\_id – primárny kľúč s unikátnym identifikačným číslom autora.
* author\_name – krstné meno autora.
* surname – priezvisko autora.
* nationality – národnosť autora.

**store\_table :** tabuľka, ktorá reprezentuje jednotlivé pobočky, v ktorých sa dá nakupovať.

* store\_id – primárny kľúč s unikátnym identifikačným číslom pobočky.
* store\_name – názov pobočky.
* city – mesto, v ktorom je pobočka.
* street – ulica pobočky.

**album :** tabuľka, ktorá reprezentuje jednotlivé albumy pesničiek.

* album\_id – primárny kľúč s unikátnym identifikačným číslom albumu.
* picture\_id – cudzí kľúč s identifikačným číslom priradeného obrázku.
* title – názov albumu.
* genre – žáner albumu.
* release\_date – dátum, kedy bol album vydaný.

**song :** tabuľka, ktorá reprezentuje jednotlivé pesničky, ktoré sa môžu nakúpiť.

* song\_id – primárny kľúč s unikátnym identifikačným číslom pesničky.
* author\_id – cudzí kľúč s identifikačným číslom priradeného autora.
* album\_id – cudzí kľúč s identifikačným číslom priradeného albumu.
* song\_length – dĺžka pesničky v sekundách.
* title – názov pesničky.

**order\_table :** tabuľka, ktorá reprezentuje objednávky užívateľov.

* song\_id – identifikačné číslo pesničky, ktorú si objednal užívateľ.
* user\_id – identifikačné číslo užívateľa, ktorý si pesničku objednal.
* order\_id – číslo objednávky.
* order\_date – dátum objednávky.

**registry :** tabuľka, ktorá reprezentuje konkrétne pesničky, ktoré sa nachádzajú v jednotlivých obchodoch.

* song\_id – identifikačné číslo pesničky, ktorá sa nachádza v obchode.
* store\_id – identifikačné číslo obchodu, v ktorom sa nachádza pesnička.

# 3 Implementácia

Informačný systém sme navrhli v programovacom jazyku Java. Aplikácia je logicky rozdelená do balíčkov pre back-end a front-end.

Obe časti aplikácie sú napísané v Jave, pričom na front-ende sa používa knižnica SWING. Aplikácia je ošetrená na front-ende aj na back-ende. Na back-ende aplikácie sa pracuje s knižnicou sql, pomocou ktorej sme boli schopní odosielať jednotlivé queries na server.

Dáta su uložené na školskom serveri asterix. Pri spustení aplikácie sa vytvorí connection na server, pomocou ktorého aplikácia komunikuje s databázou. Tento connection zaniká pri zatvorení aplikácie.

GUI je veľmi intuitívne, vo väčšine prípadov sú pre usera k dispozícii navigačné tooltipy v prípade núdze. Ak sa user pomýli alebo zabudne vyplniť nejaké pole, bude zachytená výnimka a bude o chybe riadne informovaný.

# 4 Práca s aplikáciou

**User:**

Používateľ sa po spustení aplikácie musí zaregistrovať ak nemá konto. Po úspešnej registrácii bude presmerovaný späť na prihlasovaciu obrazovku, kde vyplní svoje prihlasovacie údaje, čím sa dostane do aplikácie. Následne sa mu zobrazí používateľská obrazovka, v ktorej môže napríklad:

* zobraziť všetky piesne alebo albumy nachádzajúce sa v systéme
* zobraziť si náhľad piesne alebo albumu
* vyhľadať konkrétnu pieseň podľa názvu (alebo znakov, ktoré obsahuje názov)
* vyhľadať všetky albumy ktoré boli vydané v rozmedzí dvoch dátumov
* zobraziť si všetky piesne a albumy, ktoré mu už patria
* privlastniť si pieseň zo zoznamu piesní, ktoré ešte nemá
* zobraziť si detail piesne alebo albumu, ktorý mu už patrí
* vidieť svoju prezývku
* vidieť presný čas svojho prihásenia do aplikácie
* odhlásiť sa...

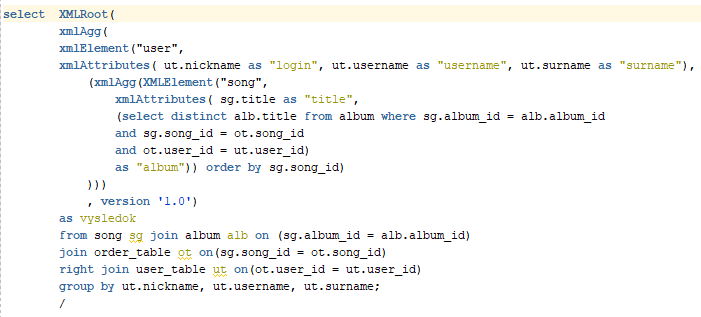
**Admin:**

Administrátor v aplikácii je len jeden. Nie je možné vytvoriť nové administrátorské konto priamo z aplikácie, je nutný programátorský prístup. Administrátor po prihlásení môže napríklad:

* zobraziť si určitý počet používateľov s najväčším počtom vlastnených piesní
* zobraziť si určitý počet piesní s najdlhším trvaním
* zobraziť si počet predajov rozdelených podľa dňa v týždni
* zobraziť si v ktorých pobočkách sa nachádza konkrétna pieseň
* zobraziť si všetky albumy v systéme
* zobraziť si všetky piesne v systéme
* vidieť detail albumu
* vidieť detail piesne
* odhlásiť sa…

# 5 Analýza príkazov Select

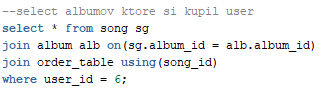
**XML Report:**



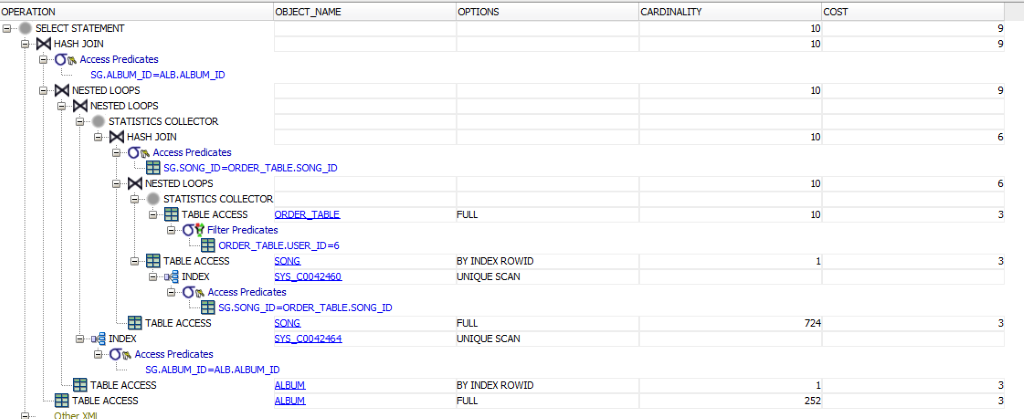
**Výstup:**



**Analýza príkazu Select**



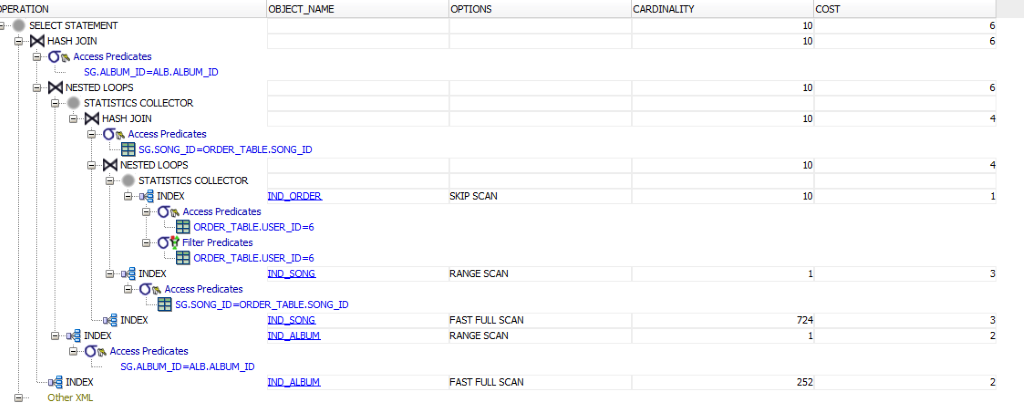
**Plán vykonania bez indexácie:**



Môžeme vidieť Table access: FULL to znamená, že pri hľadaní vhodných prechádzame všetkými dátami v tabuľkách.

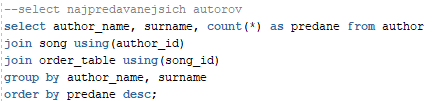
**Pridali sme nasledovné indexy:**



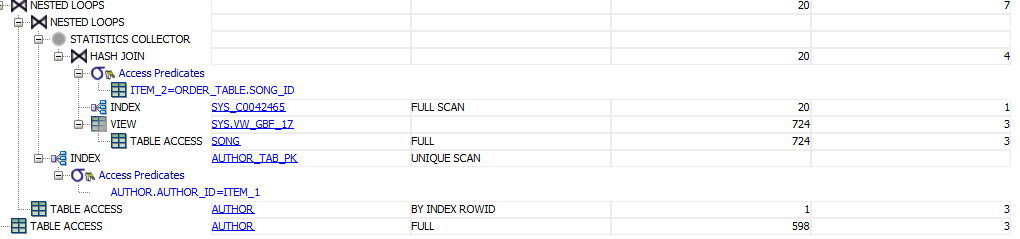


Vidíme, že sa zmenil prístup z table access: FULL na INDEX: FAST FULL SCAN a znížila sa cena vyhľadávania

**Analýza príkazu Select**



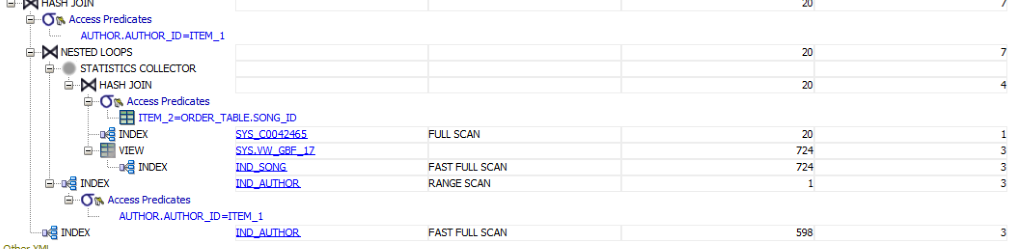
**Plán vykonania bez indexácie:**



Môžeme vidieť Table access: FULL to znamená, že pri hľadaní vhodných prechádzame všetkými dátami v tabuľkách.

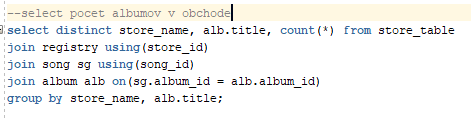
**Pridali sme nasledovné indexy:**



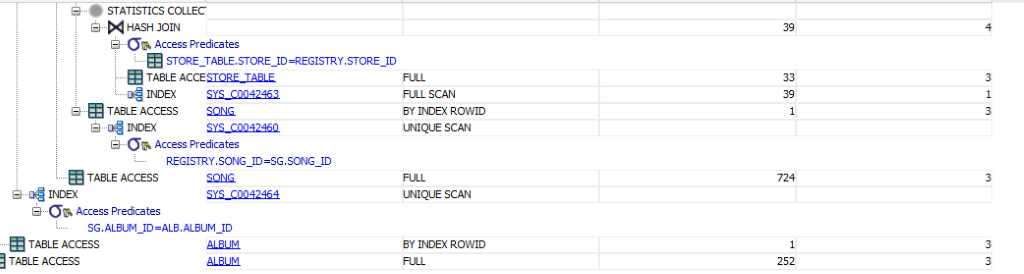


Vidíme, že sa zmenil prístup z table access: FULL na INDEX: FAST FULL SCAN ale cena sa nezmenila.

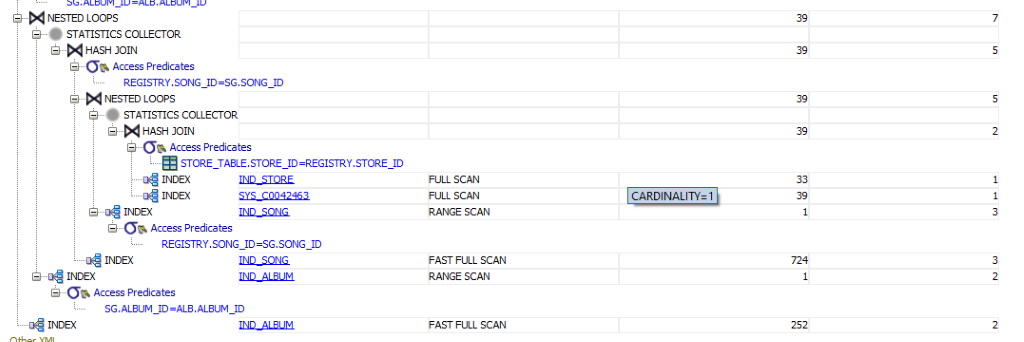
**Analýza príkazu Select**



**Plán vykonania bez indexácie:**



**Pridali sme nasledovné indexy:**



Vidíme, že sa zmenil prístup z table access: FULL na INDEX: FAST FULL SCAN alebo FULL SCAN a znížila sa cena vyhľadávania.