



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA INFORMAČNÍCH TECHNOLOGIÍ**

FACULTY OF INFORMATION TECHNOLOGY

**ÚSTAV INTELIGENTNÍCH SYSTÉMŮ**

DEPARTMENT OF INTELLIGENT SYSTEMS

**SPRÁVA NADSTANDARDNÍCH METADAT K HUDEBNÍM SOUBORŮM – DOHLEDÁNÍ INFORMACÍ O ORIGINÁLNÍCH VINYLOVÝCH DESKÁCH**

ADDITIONAL METADATA MANAGEMENT FOR MUSIC FILES - FINDING ORIGINAL VINYL RECORDS METADATA

**BAKALÁŘSKÁ PRÁCE**

BACHELOR'S THESIS

**AUTOR PRÁCE**

AUTHOR

**TOMÁŠ SVĚRÁK**

**VEDOUCÍ PRÁCE**

SUPERVISOR

**doc. Ing. VLADIMÍR JANOUŠEK, Ph.D.**

BRNO 2021

## Zadání bakalářské práce



Student: **Svěrák Tomáš**  
Program: Informační technologie  
Název: **Správa nadstandardních metadat k hudebním souborům: Dohledání informací o originálních vinylových deskách**  
**Additional Metadata Management for Music Files: Finding Original Vinyl Records Metadata**

Kategorie: Informační systémy

Zadání:

1. Prostudujte online služby, poskytující metadatata k hudebním souborům. Seznamte se s existujícími aplikacemi, která tato metadata využívají.
2. Stanovte požadavky na správu metadat hudebních souborů se zaměřením na historické nahrávky na vinylových deskách. Zaměřte se na možnost poloautomatického dohledání a případné ruční editace informací o původním vydání skladby - rok vydání, která společnost skladbu vydala, fotografii 45 rpm desky, autora skladby, případně i umístění v žebříčcích a jiné dohledatelné informace. To vše za předpokladu, že je znám pouze interpret a název skladby.
3. Navrhněte prostředky pro uživatelsky přívětivou správu metadat z bodu 2. Metadata mohou být uložena přímo v hudebních souborech, nebo v separátní databázi.
4. Navržené prostředky realizujte, ověřte jejich praktickou použitelnost a vyhodnoťte dosažené výsledky.

Literatura:

- ID3 Tags. URL: <http://id3.org/>
- MusicBrainz. URL: <https://musicbrainz.org/>
- 45cat. URL: <http://www.45cat.com>

Podrobné závazné pokyny pro vypracování práce viz <https://www.fit.vut.cz/study/theses/>

Vedoucí práce: **Janoušek Vladimír, doc. Ing., Ph.D.**

Vedoucí ústavu: Hanáček Petr, doc. Dr. Ing.

Datum zadání: 1. listopadu 2020

Datum odevzdání: 12. května 2021

Datum schválení: 11. listopadu 2020

## Abstrakt

Cílem této práce je navrhnout a vytvořit aplikaci, která v online hudebních databázích vyhledá informace o původních vinylových deskách. Jedná se o multiplatformní desktopovou aplikaci, která dohledává data v databázích Discogs, MusicBrainz, 45cat a 45worlds a ukládá je do lokální databáze. Uživateli pak umožňuje dohledané informace spravovat v uživatelsky přívětivém rozhraní a na základě nich se rozhodnout o originalitě nalezených nahrávek.

## Abstract

The main goal of this work is to design and create an application that searches online music databases for information about the original vinyl records. It is a cross-platform desktop application that searches for data in Discogs, MusicBrainz, 45cat and 45worlds databases and stores them in a local database. It then allows the user to manage the retrieved information in a user-friendly interface and on their basis to decide on originality of the found recordings.

## Klíčová slova

hudební soubory, metadata, vinylové desky, python, hudební databáze

## Keywords

music files, metadata, vinyl records, python, music databases

## Citace

SVĚRÁK, Tomáš. *Správa nadstandardních metadat k hudebním souborům – dohledání informací o originálních vinylových deskách*. Brno, 2021. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce doc. Ing. Vladimír Janoušek, Ph.D.

# Správa nadstandardních metadat k hudebním souborům – dohledání informací o originálních vinylových deskách

## Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana doc. Ing. Vladimíra Janouška, Ph.D. Uvedl jsem všechny literární prameny, publikace a další zdroje, ze kterých jsem čerpal.

.....

Tomáš Svěrák  
11. května 2021

## Poděkování

Rád bych poděkoval panu doc. Ing. Vladimírovi Janouškovi, Ph.D. za vedení a odbornou pomoc při vypracování této práce.

# Obsah

<b>1</b>	<b>Úvod</b>	<b>2</b>
<b>2</b>	<b>Způsob uložení zvukových dat</b>	<b>3</b>
2.1	Digitalizace zvuku . . . . .	3
2.2	Formáty hudebních souborů . . . . .	4
2.3	Metadata . . . . .	8
2.4	Nadstandardní metadata vinylových desek . . . . .	10
<b>3</b>	<b>Zhodnocení současného stavu</b>	<b>14</b>
3.1	Databáze uchovávající hudební metadata . . . . .	14
3.2	Nástroje pro práci s metadaty . . . . .	17
<b>4</b>	<b>Návrh aplikace</b>	<b>20</b>
4.1	Základní specifikace . . . . .	20
4.2	Získávání dat z hudebních databází . . . . .	22
4.3	Návrh lokální databáze . . . . .	22
<b>5</b>	<b>Výsledný nástroj</b>	<b>24</b>
5.1	Použité technologie . . . . .	24
5.2	Vyhodnocení nalezení shody . . . . .	25
5.3	Implementace vyhledávání v hudebních databázích . . . . .	26
5.4	Spuštění vyhledávání . . . . .	30
<b>6</b>	<b>Testování a vyhodnocení výsledků</b>	<b>32</b>
6.1	Testování vyhledávání . . . . .	32
6.2	Návrh možných rozšíření . . . . .	34
<b>7</b>	<b>Závěr</b>	<b>35</b>
	<b>Literatura</b>	<b>36</b>

# Kapitola 1

## Úvod

V dnešní době téměř každý z nás vlastní chytrý mobilní telefon, na kterém může poslouchat své oblíbené skladby a stále více lidí využívá hudebních streamovacích služeb. I přes to se stále najdou jedinci, kteří nedají dopustit na klasické vinylové desky a nejlépe si hudbu vychutnají právě z nich. Často se jedná o vášnivé sběratele, kteří vlastní rozsáhlé sbírky vinylových desek. A přesně takovým lidem by mohl přijít vhod nástroj, kterým se zabývá tato bakalářská práce.

Tito nadšenci mezi sebou často sdílí informace o svých kolekcích a jednotlivých deskách. Za tímto účelem dokonce vznikly různé webové stránky a databáze, kam kdokoli může přispět, případně hodnotit již vložené nahrávky a diskutovat o nich. Může se jednat o databáze zaměřené přímo na vinylové desky, či na širokou škálu formátů ať už jde o paměťové pásky nebo moderní Blu-ray disky.

Tato práce si dává za cíl, vytvořit desktopovou aplikaci, která by využívala těchto databází a zaměřovala se na dohledání informací o původních nahrávkách na vinylových deskách. Uživatel pak může pouze na základě názvu interpreta a skladby nechat nahrávky dohledat. Jelikož častokrát nelze jednoznačně určit, která z nahrávek je původní, rozhodnutí o originalitě je již ponecháno na samotném uživateli.

Kapitola 2 pojednává o různých formátech, které slouží pro uložení hudebních souborů. Dále se v ní lze dočíst, co znamená pojem metadata a co si pod ním představit v souvislosti s hudebními soubory a na závěr jaká metadata mohou existovat v souvislosti s vinylovými deskami. Následně jsou v kapitole 3 detailně popsány jedny z největších hudebních databází, které tyto data poskytují. Poté jsou uvedeny příklady již existujících aplikací, které s databázemi nějakým způsobem také pracují. Návrh aplikace a specifikace požadavků na ni jsou popsány v kapitole 4. V další kapitole 5 jsou popsány implementační detaily, především se zaměřením na vyhledávání nahrávek a technologie, kterých bylo využito. Závěrečná kapitola 6 se zaměřuje na testování aplikace a vyhodnocení dosažených výsledků.

## Kapitola 2

# Způsob uložení zvukových dat

Tato kapitola nastiňuje základní principy uložení hudebních souborů v počítači. Na začátku pojednává o samotné transformaci zvuku z analogové do digitální podoby (digitalizaci) Poté uvádí možné formáty, ve kterých mohou být data uloženy včetně případné možnosti jejich komprese. Na závěr je vysvětleno, co si lze představit pod pojmem metadata v souvislosti s hudebními soubory.

### 2.1 Digitalizace zvuku

Veškerý zvuk kolem nás, který lze slyšet je analogový. Pro nahrání je potřeba zvukové vlny převést na elektrický signál. Klasickým příkladem takového zařízení je mikrofon. Tento signál je stále analogový a musí se pomocí analogově digitálního převodníku (A/D) převést na digitální. K tomu lze využít několika metod. Jednou z nich je pulzní kódová modulace, běžně označovaná zkratkou PCM (Pulse Code Modulation).

**Digitalizace se skládá ze tří hlavních operací:**

1. Vzorkování
2. Kvantování
3. Kódování

#### Vzorkování

Spojité signály není možné v počítači uložit v jeho původní podobě. Řešením je takový signál diskretizovat v čase. Tedy zaznamenávat s určitou periodou jednotlivé vzorky. Obvykle se uvádí jako vzorkovací frekvence (sample rate), která říká kolik vzorků je vytvořeno za jednotku času [11].

#### Kvantování

Po navzorkování signálu je získán konečný počet diskrétních vzorků. Ovšem stále je v nich obsaženo obrovské množství informací. Je nutné tyto hodnoty zaokrouhlit na konečné množství hodnot. K tomu je využíváno kvantizačních hladin. Jejich počet je udáván bitovou hloubkou (bit depth). V případě použití 16-bitového kvantování může signál nabývat 65536 různých hodnot [8].

## Kódování

Poslední fází digitalizace signálu je kódování. Jedná se o přiřazení binárního čísla jednotlivým kvantizačním hladinám. Naivním přístupem by bylo běžné převedení z desítkové do dvojkové soustavy. V praxi toto není příliš vhodné a využívá se takzvaných kodeků. Toto slovo vzniklo spojením slov kodér a dekodér. Z hardwarového pohledu jde většinou o jedno zařízení, které obsahuje A/D převodník i digitálně analogový převodník (D/A). V případě softwaru se jedná o program, který může zajišťovat i kompresi signálu.

## 2.2 Formáty hudebních souborů

Slovo formát v souvislosti s hudebními soubory může být komplikovanější, než se na první pohled zdá. Často je používáno nesprávně, nebo přinejmenším nepřesně. Je potřeba rozlišovat mezi kontejnerem a kodekem.

Kontejner si lze představit jako schránku, která obsahuje samotná data a k nim připojená metadata což jsou informace o samotných datech (více o nich se lze dočíst v sekci 2.3). Některé kontejnery podporují různé typy obsahu a existují i kontejnery, které v sobě mohou uchovávat zároveň zvukovou, i video stopu.

Naopak kodek udává, jak jsou data zakódována, jestli jsou komprimována a zda se jedná o ztrátovou či bezztrátovou kompresi.

## Nekomprimované formáty

Jak již název napovídá jedná se o formáty, kde původní zvuk nebyl žádným způsobem upraven. Pouze zachycen a převeden do digitální podoby. Kvalita zvuku je tedy na vysoké úrovni, ale v porovnání s komprimovanými formáty zabírají data velké množství místa. Jako příklad lze uvést, že minutová nahrávka se 24bitovou hloubkou, vzorkovací frekvencí 192 kHz a dvěma kanály (běžné u DVD-Audio) by zabírala 69 MB na uložení.

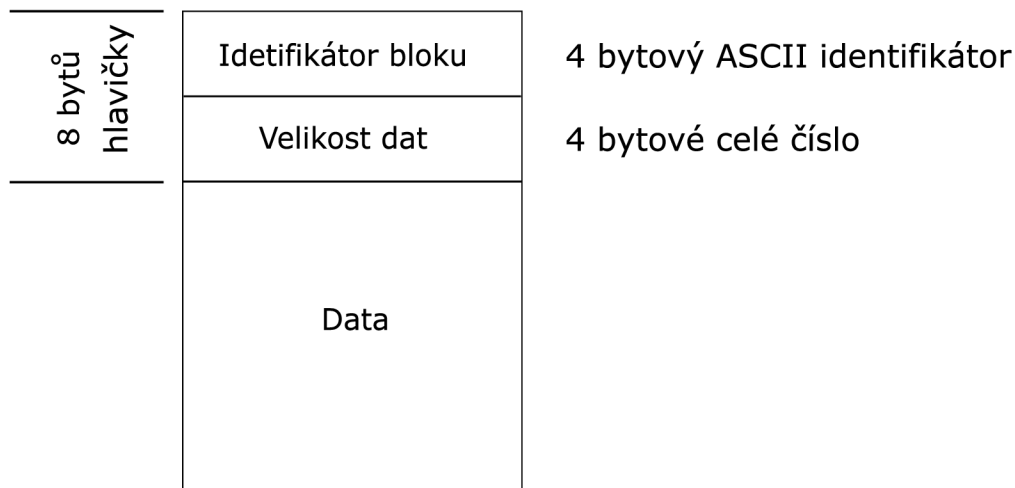
### AIFF

Audio Interchange File Format (AIFF) byl vytvořen společností Apple a je využíván především na počítačích Macintosh. Vychází z Interchange File Formátu (IFF), jenž byl využíván nejen pro hudební formáty, ale i pro formáty grafické, nebo dokonce textové. Všechny takové formáty využívají takzvaných bloků (chunks). Ty mají společnou strukturu, kde první 4 bajty slouží pro identifikaci typu bloku (chunk id), a další 4 bajty udávající počet datových bajtů, které následují (chunk data size). Tato obecná podoba je zobrazena v obrázku 2.1. Soubor je potom tedy kolekcí takovýchto bloků. Ve společném bloku (common chunk) jsou uloženy základní metadata jako vzorkovací frekvence, počet kanálů, bitová hloubka a počet vzorků. Lze jej ale rozšířit tagy ID3v2. Ačkoliv tento formát slouží pouze pro uložení nekomprimovaných dat, existuje varianta, která v sobě může nést data komprimovaná a pro odlišení je označována jako AIFF-C. V takovém případě je ve společném bloku uveden typ komprese.

### WAV

Waveform Audio File Format (WAV) je zvukový formát, který vyvinuly společnosti Microsoft a IBM. Formát se řadí mezi nejběžněji používané formáty na uložení takzvaných





Obrázek 2.1: Obecná podoba bloku, který se používá ve formátu IFF a ostatních formátech, které z něho vycházejí

raw (neupravených) zvukových souborů. I když jde o kontejner, jenž by teoreticky mohl obsahovat i komprimovaná data, není to běžné. WAV je příkladem Resource Interchange File Formátu (RIFF), který je založen na výše zmiňovaném IFF. Opět je tedy soubor tvořen jednotlivými bloky. Jedním z nich je info blok, který slouží pro tagy a obsahuje většinu standardních tagů jako je název interpreta a skladby, žánr, autorské právo a další. Může být ovšem rozšířen o další tagy, jako jsou ID3 a to díky tomu, že lze připojit libovolné množství bloků. Ty sice aplikace nemusí rozpoznat, ale specifikace formátu RIFF uvádí, že neznámé bloky musí být přeskočeny.

## Bezeztrátově komprimované formáty

Tyto formáty provádí kompresi původního signálu, čímž se snaží docílit snížení kapacity potřebné k jeho uložení. Z takto komprimovaného souboru lze získat plnohodnotný původní signál, jako by k žádné kompresi nedošlo, proto jsou tyto formáty vhodné k archivaci dat.

### FLAC

Free Lossless Audio Codec (FLAC) je asi nejrozšířenější zvukový formát, který používá bezeztrátovou kompresi. Důvodů je hned několik a k jejich pochopení stačí rozebrat samotný název formátu. Free – formát je zdarma a volně dostupný i pro komerční použití. Zároveň i jeho zdrojový kód spadá pod otevřené licence BSD a GNU GPL. Lossless – jak již bylo uvedeno je bezeztrátový, ale zároveň dokáže snížit velikost původních dat o 30–40 %. Mezi další výhody patří rychlost dekódování, široká hardwarová podpora a podpora streamování. FLAC podporuje tagování a je snadno rozšiřitelný o další tagy. Často bývá používán v kombinaci s Ogg kontejnerem. Díky všem těmto vlastnostem je velice oblíben mezi zastánci vinylových desek, kteří si digitalizují nebo archivují své sbírky [9].

## ALAC

Apple Lossless Audio Codec (ALAC) je formát vytvořený firmou Apple, původně jako proprietární, ale později v roce 2011 se stal volně šířitelným. Často bývá používán v kombinaci s MP4 kontejnerem. Co se komprese týče, je o něco méně efektivní než FLAC.

## Ztrátově komprimované formáty

Tyto formáty spoléhají na nedokonalost lidského ucha. Záměrně jsou zahazovány některé informace, u kterých je vyhodnoceno, že by je posluchač stejně nezaznamenal. Například velmi nízké frekvence, nebo tiché zvuky, které jsou přehlušeny jinými zvuky. Dochází tedy k degradaci kvality a nelze dosáhnout původní kvality zvuku dekompresí.

### MP3

MPEG-1 Audio Layer III nebo MPEG-2 Audio Layer III (MP3) je ztrátový formát kódování zvuku, vyvinutý skupinou MPEG (Motion Picture Experts Group). Původní formát standardu MPEG-1 byl později rozšířen o podporu více zvukových kanálů, vyšších přenosových rychlostí (bit rate) a stal se součástí standardu MPEG-2.

Komprese pomocí MP3 funguje na principu, snižování přesnosti, nebo aproximaci určitých zvukových komponent. Vychází z psychoakustické analýzy<sup>1</sup> a tento způsob komprese je běžně nazýván psychoakustickým modelováním.

MP3 využívá z této oblasti dvou technik. První z nich nazývaná sluchové maskování (auditory masking). Pokud ve stejnou chvíli zazní dva tóny, první s frekvencí 1000 Hz a hlasitostí 0 dB a druhý s frekvencí 1100 Hz a hlasitostí -10 dB, druhý tón nelze slyšet nejen z důvodu, že je tižší, ale také kvůli podobnosti jejich frekvencí. Pokud by u druhého tónu frekvence postupně stoupala na 4000 Hz (hlasitost zůstává na -10 dB), tóny si začnou být čím dál méně podobné, až v určitém bodě lidské ucho zaznamená dva různé tóny, kdy jeden bude hlasitější než druhý.

Druhá technika s názvem dočasná maskování (temporal masking) je více spjatá s časem, než samotnými frekvencemi. Bylo zjištěno, že lidé mají problém rozpoznat rozdílně znějící zvuky, které následují rychle po sobě. Pokud tedy zazní v téměř stejnou chvíli hlasitý a následně po něm tižší tón, tak tižší tón nebude lidským uchem zaznamenán (tento jev je nazýván postmasking). Klíčovým prvkem je zde interval mezi těmito tóny. Jedná se o přibližně 5 milisekund v případě, že jde opravdu pouze o dva samostatné tóny. Samozřejmě se tato doba liší i v závislosti na množství tónů znějících naráz. Tento jev funguje i opačně, tedy pokud nejdříve zní tichý tón a po něm následuje hlasitý (tento případ je nazýván premasking) [6].

Při konverzi do MP3 jediným parametrem, kterým lze ovlivnit kvalitu a velikost výsledného souboru, je přenosová rychlost. Ta se u standardu MPEG-2 může pohybovat od 8 kbit/s do 160 kbit/s. Čím vyšší je přenosová rychlost, tím je lepší kvalita zvuku, ale zároveň i výsledný soubor zabírá více místa. Kompresí lze dosáhnout snížení velikosti až o 90 % oproti původní nahrávce.

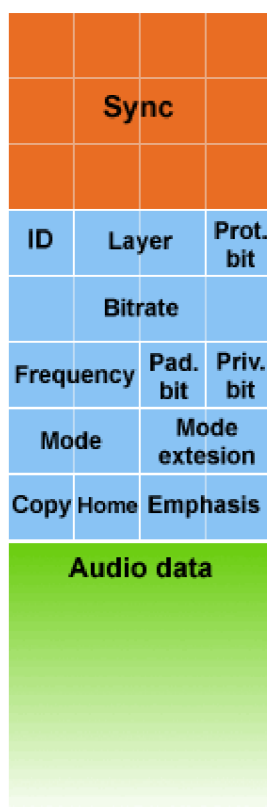
MP3 soubor nemá žádnou hlavičku, kterou by byl identifikován jeho začátek. Místo toho se skládá z posloupnosti malých rámců (frames). Každý rámeček má svoji vlastní hlavičku a data obsahující zvuk. Jednotlivé rámečky jsou na sobě nezávislé, je tedy možné vzít

---

<sup>1</sup>Psychoakustická analýza – vychází z vědního oboru zvaný psychoakustika, který se zaměřuje na reakce lidí spojené s určitými zvuky

jakoukoliv část souboru a správně ji přehrát. Všechny rámce mají konstantní časovou délku přibližně 260 ms. Velikost v bajtech se může lišit v závislosti na vzorkovací a přenosové rychlosti [3].

Obrázek 2.2 znázorňuje strukturu jednoho rámce. Jeho hlavička má celkem 32 bitů. Prvních 12 bitů je vždy nastavena na hodnotu 1 a tvoří takzvaný synchronizační rámec (frame sync), který identifikuje začátek nového rámce. Následující bit značí, zda se jedná o soubor MPEG-1, či MPEG-2. Layer bity říkají, o jakou vstupu se jedná (Layer I, II, III) a v závislosti na nich a na ID bitu je interpretována hodnota bitrate bitů. Posledním běžně užívanými bity, jsou frekvenční bity. Jejich hodnota může mít taktéž různou interpretaci, tentokrát pouze v závislosti na ID bitu.



Obrázek 2.2: Struktura MP3 rámce (frame). Celková velikost hlavičky rámce (oranžově a modře zvýrazněná část) je 32 bitů. Frame sync má všech 12 bitů nastaveno na 1 pro identifikaci začátku rámce. ID a Layer bity dohromady udávají přesný typ (standard), jaký je v rámci obsažen. Bitrate a frequency jsou interpretovány v závislosti obsahu ID a Layer bitů. Zbylé bity nemají, tak zásadní význam. Po nich následují již samotná zvuková data. Převzato z [7].

MP3 soubory samy o sobě nedefinují žádný tagovací formát. Časem se ale rozšířili ID3v1 a ID3v2 tagy a staly se neoficiálním standardem pro tagování MP3. Další možností je použití APEv2 tagů. Všechny tyto varianty se mohou vyskytovat buď na začátku, nebo na konci samotných MP3 souborů a nezasahují do datové části.

## AAC

Advanced Audio Coding je dalším zvukovým formátem se ztrátovou kompresí a byl navržen jako následník MP3. Lze pomocí něj dosáhnout lepší kvality zvuku, než u MP3 při použití stejné přenosové rychlosti. Dále podporuje větší rozsah vzorkovací frekvence, která se pohybuje od 8 do 96 kHz n rozdíl od MP3, kde při standardu MPEG-2 je rozsah pouze od 16 do 24 kHz.

## 2.3 Metadata

Metadata jsou, jak uvádí definice, data obsahující informace o jiných datech. Vyskytují se ve všech různých odvětvích ať už jsou to webové stránky (meta tag v hlavičce), digitální fotografie (datum a čas jejího pořízení), dokonce i v souborových systémech (názvy souborů a adresářů). V souvislosti s hudebními soubory se jedná zejména o:

- Název skladby
- Interpret
- Délka skladby
- Vzorkovací frekvence

Může jich být ale mnohem více, pro různé formáty a s různou strukturou.

## ID3

ID3 tagy jsou jednoznačně nejrozšířenější. Hlavním důvodem je, že byly vytvořeny cíleně pro formát MP3, který je sám také nejpoužívanějším hudebním formátem. Ale mohou být použity i s jinými formáty jako například WAV (neobsahuje žádné tagy), nebo AIFF. ID3 by se daly tagy využít i u dalších formátů, ale aplikace by poté nemusely daný formát rozpoznat.

### ID3v1

Existuje několik různých verzí ID3 tagů. První vznikla již v roce 1996, kdy byl na konec MP3 souboru připojen blok o délce 128 bajtů, který měl předem definované položky a jejich maximální délky. Tato verze je označována jako ID3v1. Jeho podoba je zobrazena v tabulce 2.1. Hlavním důvodem přidání tagu na konec souboru byla zpětná kompatibilita s přehrávači, které tag nerozpoznaly. Pevnou délku měl zase proto, aby byl jednoduše dohledatelný a stačilo se pokusit vyhledat 128 bajtů od konce souboru, zda obsahuje výraz TAG o velikosti 3 bajtů, který identifikuje začátek tagu. Dále tag obsahoval název skladby, jméno interpreta, název alba a komentář, kde všechny tyto položky mohly mít délku až 30 bajtů. Ještě obsahoval rok (4 bajty) a žánr (1 bajt). Pokud by obsah některé z položek byl kratší, zbytek by měl obsahovat binární 0. Žánr měl velikost pouze 1 bajt, protože byl definován seznam 80 žánrů, kterým odpovídaly hodnoty této položky od 0 do 79.

Necelý rok poté byla tato verze upravena, kde byl komentář zkrácen na 28 bajtů a 2 bajty byly využity pro pořadí dané skladby na CD. Přesněji pokud první bajt obsahoval binární 0 (tedy neobsahoval komentář), následující bajt označoval pořadí. Tato verze nese označení ID3v1.1. [10]

## ID3v2

ID3v1 tagy měly několik značných limitací. Především fixní velikost tagu 128 bajtů. S tím také souvisí fixní velikost jednotlivých položek a kvůli výskytu tagu až na konci souboru, se jednalo o poslední věc, která by dorazila při streamování.

Proto v roce 1998 skupina autorů vytvořila verzi ID3v2, která se stala neformálním standardem. Na obrázku 2.3 lze vidět ilustraci struktury souboru s tagem ID3v2. Jedná se o blok dat předcházející zvukovým datům. Tento blok je dále složen z jednotlivých položek (rámců). Velikost jednoho rámce je omezena na 16 MB, přičemž maximální velikost celého tagu je 256 MB. Podstatnou změnou je umístění tagu na začátek souboru, před data obsahující zvuk. Tím se vyřešil problém při streamování. Aby se nemohlo stát, že se někde v tagu vyskytne synchronizační rámec, který přehrávačům signalizuje začátek MP3 rámce, obsahují ID3 tagy mechanismus, který tuto posloupnost bitů nahradí. Protože se jedná o kontejnerový formát, je jednoduše rozšiřitelný o nové typy rámců. Mezi nimi může být například i obrázek a v podstatě jakýkoliv soubor.

Existují celkem tři revize, které jsou vždy zpětně kompatibilní. První z nich nesoucí označení ID3v2.2 je dnes považována za zastaralou. Nejrozšířenější revizí je ID3v2.3. Hlavní změnou oproti předchozí verzi je rozšíření identifikátoru rámce ze 3 bajtů na 4. Poslední revize je rozšířena o podporu kódování UTF-8.

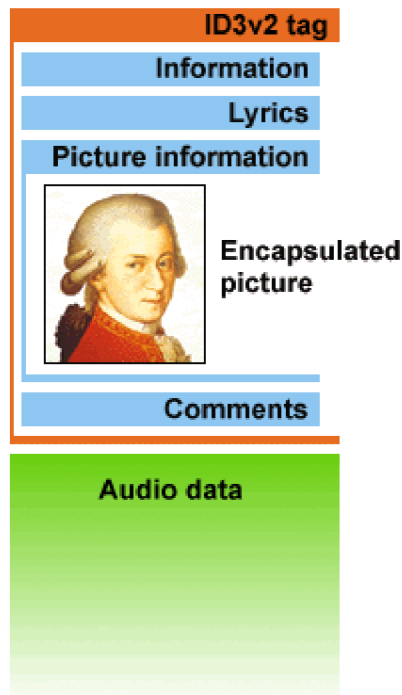
### Vorbis comment

Vorbis comment kontejner obsahující metadata a může být použit například s formáty Vorbis, FLAC a Opus. Jeho účelem je uchování stručných, textových informací. Má velice jednoduchou strukturu.

Vždy se jedná o dvojici 'jméno položky'='hodnota', tedy například ARTIST=Rolling Stones. Předdefinovaných je pouze 15 základních položek. Ovšem kdokoliv si může vytvořit položku s libovolným názvem. Tím nastává problém se standardizací. Jméno jakékoliv položky může být uvedeno několikrát, třeba v případě, že se na skladbě podílelo více interpretů, nebo nemusí být uvedeno vůbec.

Název položky	Velikost [B]
TAG	3
Název skladby	30
Interpret	30
Název alba	30
Rok	4
Komentář	30
Žánr	1

Tabulka 2.1: Struktura ID3v1 tagu. Položka TAG slouží k identifikaci začátku tagu. U ostatních položek kromě žánru, jejich velikost udává maximální počet znaků. Pro položku žánr byla vytvořena speciální tabulka, která interpretovala hodnoty od 0 do 79 jako jednotlivé žánry.



Obrázek 2.3: Přibližná podoba souboru obsahující ID3v2 tag. Hlavní změnou oproti předchozí verzi, je výskyt tagu na začátku souboru, před zvukovými daty. Jednotlivých položek (rámců) může být obsaženo mnohem více a s proměnlivou délkou. Převzato z [10].

## 2.4 Nadstandardní metadata vinylových desek

Problematika metadat je u vinylových desek podstatně složitější. Pokud by se jednalo o CD, jednotlivé skladby již metadata zpravidla obsahují (alespoň ty základní). Ovšem když se někdo rozhodne digitalizovat svoji vinylovou kolekci, musí pracovat pouze s fyzickou kopií a informace dohledat na potisku (label), nebo obalu desky. Případně lze využít některé z komunitních internetových databází viz. sekce 3.1, kde mezi sebou lidé sdílí informace, poznatky a rady o nahrávkách. Dále jsou uvedeny metadata, které mohou být při této činnosti užitečné.

- **Názvy skladeb, jména interpretů a název nahrávky** – Tyto metadata by šlo zařadit mezi ty běžně používané i u ostatních médií. Ačkoliv i u vinylových desek se jedná o ty nejdůležitější metadata. Díky nim je zpravidla možné dohledat většinu informací uvedených dále.
- **Datum vydání** – Další položka, která je jedna z nejdůležitějších. Ovšem na potisku vinylových desek tento údaj zpravidla nebývá uveden. Často není známo ani přesné datum vydání, ale pouze rok. Ve výjimečných případech se může objevit na obalu vinylové desky.
- **Potisk (label)** – Anglickým výrazem label může být myšlen samotný potisk desky a nebo značka, respektive jméno uvedené na desce a nebo se může jednat zároveň o značku a distributora. Label ve smyslu potisk, který je nalepen na desce bývá hlavním zdrojem informací. Lze z něho zjistit jména různých společností, které se podílely

na dané nahrávce. Může jít například o studio, kde byla nahrávka pořízena, či jaká společnost skladbu vydala. V obou případech může být uvedeno i místo odkud společnost pochází. Na obrázku 2.4 lze vidět, kolik důležitých informací se na potisku může nacházet.

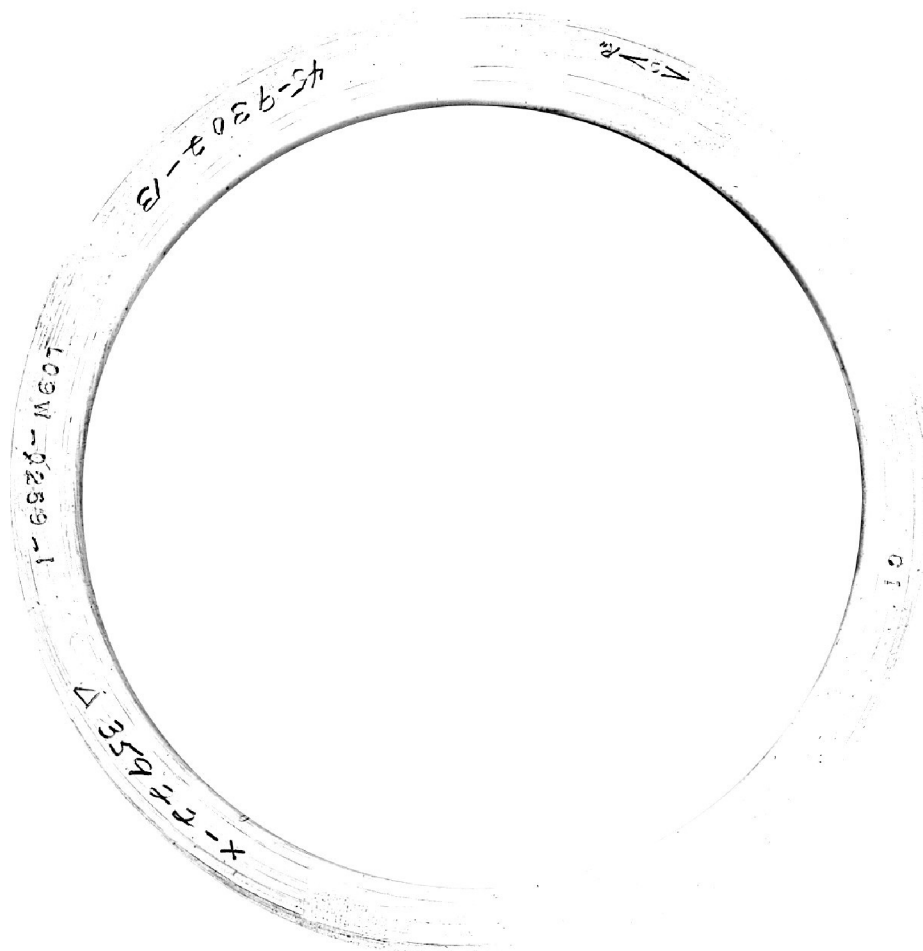


Obrázek 2.4: Potisk vinylové desky. Lze na něm dohledat spoustu užitečných informací o nahrávce. Číslo 1 – název nahrávky. Číslo 2 – jméno interpreta. Číslo 3 – jméno značky (label) pod kterou je deska vydána. Číslo 4 – katalogové číslo desky, což je jednoznačný identifikátor nahrávky. Číslo 5 – společnosti, které se podílely na vydání. Číslo 6 – zásluhy (credits) v tomto případě jsou shodné s interpretem. Číslo 7 – země původu, GEMA je označení Německa. Číslo 8 – délky jednotlivých skladeb. Číslo 9 – názvy skladeb. Číslo 10 – matriční číslo na potisku. Číslo 11 - kód značky (label code) je jednoznačný identifikátor pro label. Převzato z [4].

- **Obal na desku** – Je stejně jako potisk, obrovským zdrojem informací. Některé údaje se mohou shodovat s těmi na potisku. Bývají zde ale uvedeni například členové kapely, kteří skladbu nahráli nebo designér obalu.
- **Katalogové číslo** – Jedná se o identifikátor konkrétního vydání. Bývá obvykle nejvýraznějším číslem, které lze na nahrávce najít. Může být uvedeno na potisku, přední, či zadní straně obalu, nebo na všech místech zároveň.

- **Země původu** – Desky jsou vydávány po celém světě. Napříč zeměmi mohou sdílet většinu informací, včetně značky pod kterou jsou vydávány. Bez tohoto údaje by až na výjimky, nebylo možné jistě říct, odkud nahrávka pochází.
- **Typ desky** – Vinylové desky jsou děleny podle několika kategorií. První z nich je velikost. Ta je udávána v palcích. Běžně jsou desky vyráběny ve velikosti 7, 10 a 12 palců. Druhou kategorií je frekvence otáčení (RPM, Revolutions Per Minute). Běžné jsou 33 1/3 RPM a 45 RPM. Existují i desky, které jsou vyráběny na rychlost 78 RPM, ale ty nejsou tolik rozšířené. Poslední kategorií je délka přehrávání. Do ní spadají singly nebo jsou někdy označovány jako short play (SP). Dále extended play (EP) a long play (LP). Tyto kategorie jsou mezi sebou různě kombinovány a to ovlivňuje maximální čas přehrávání. Obecně tato hodnota nepřekračuje 30 minut na jednu stranu desky, ale existují i výjimky, kdy se dosáhlo času přehrávání delšího než hodinu.
- **Společnosti (companies)** – Různé společnosti, které se podílely na vytvoření nahrávky. Do této skupiny může spadat například distributor, nahrávací studio, vydavatelství nebo společnost která desku vylisovala.
- **Zásluhy na vydání (credits)** – Mohou obsahovat velice podobné údaje jako záznam o společnostech. V tomto případě se ovšem jedná o jednotlivé osoby. Může se jednat o designéra potisku nebo obalu, technika, textaře a další jednotlivci, kteří se jakýmkoliv způsobem podíleli na vydání nahrávky.
- **Matriční číslo** – Nachází se u drážek, po kterých se pohybuje jehla při přehrávání skladby. Většinou se jedná o kombinace písmen a čísel, které na první pohled neznalému člověku nepřinesou žádnou užitečnou informaci. Číslo je gravírováno do desky při její výrobě a primárně slouží pro interní potřeby společnosti. Slouží například pro odlišení stran desky, aby byl nalepen správný potisk. Ovšem každý výrobce má jiné označení. O to náročnější je z tohoto označení získat užitečnou informaci. Nadšenci do vinylových desek tyto údaje zkoumají a sdílejí mezi sebou. Jedním z možných údajů co v sobě matriční číslo nese, může být rozlišení mono a stereo nahrávky nebo díky němu lze identifikovat první vylisovanou desku. Matriční číslo se může v určité podobě vyskytovat i na potisku desky. V některých případech lze na desce nalézt takových čísel větší množství a někdy se na ní mohou vyskytovat i různé symboly. Na obrázku 2.5 lze vidět příklad těchto označení.
- **Ostatní identifikátory** – Může se jednat o čárkový kód uvedený na obalu. Dalším identifikátorem může být kód značky desky (label code). Jedná se o čtyřciferné nebo pěticiferné označení konkrétní značky, před kterým bývá uvedena zkratka LC (Label Code) [4].





Obrázek 2.5: Matriční čísla a ostatní symboly vygravírované u drážek vinylové desky. Běžně jsou používána ve firmách, které tyto desky lisují, aby rozeznaly o jakou desku se jedná, předtím než je na ni nalepen potisk. Matriční čísla jsou fakt důležitá i mezi sběrateli, protože z nich lze zjistit různé informace o původu desky. Převzato z [4].

## Kapitola 3

# Zhodnocení současného stavu

Tato kapitola pojednává o současných možnostech a existujících nástrojů pro spravování hudebních metadat. V sekci 3.1 se lze dočíst o internetových databázích, která shromažďují hudební metadata hudebních souborů. Sekce 3.2 představuje již vytvořené aplikace, které s těmito metadaty pracují.

### 3.1 Databáze uchovávající hudební metadata

Existují různé databáze, které se snaží shromáždit co největší množství metadat a sdílejí je s ostatními. V komunitě nadšenců a sběratelů vinylových desek je takhle skutečnost velice ceněná, protože bez ní by nemuseli vždy být schopni o svých deskách zjistit tolik informací jako s těmito databázemi.

Všechny dále popsané databáze jsou založeny na stejném principu. Může do nich přispět úplně každý. To sebou nese i svá úskalí, jelikož ne všechny informace musí být na 100 % pravdivé. Aby bylo dosaženo co nejpřesněji uložených informací, mají jednotlivé databáze striktně nastavená pravidla na vkládání nových záznamů. Dále obsahují fóra ve kterých lidé mohou diskutovat o konkrétních příspěvcích, nebo se poradit se s ostatními o způsobu vložení informací. Pod každým příspěvkem je přítomná sekce s komentáři a recenzemi, kam mohou uživatelé psát své názory a připomínky.

Dalším pomocníkem je systém hodnocení a ověřování. Vložené příspěvky musí být schváleny ostatními uživateli, aby získali status, že se jedná o důvěryhodný zdroj pro jiné uživatele. Zároveň kdokoliv může existující příspěvky upravovat.

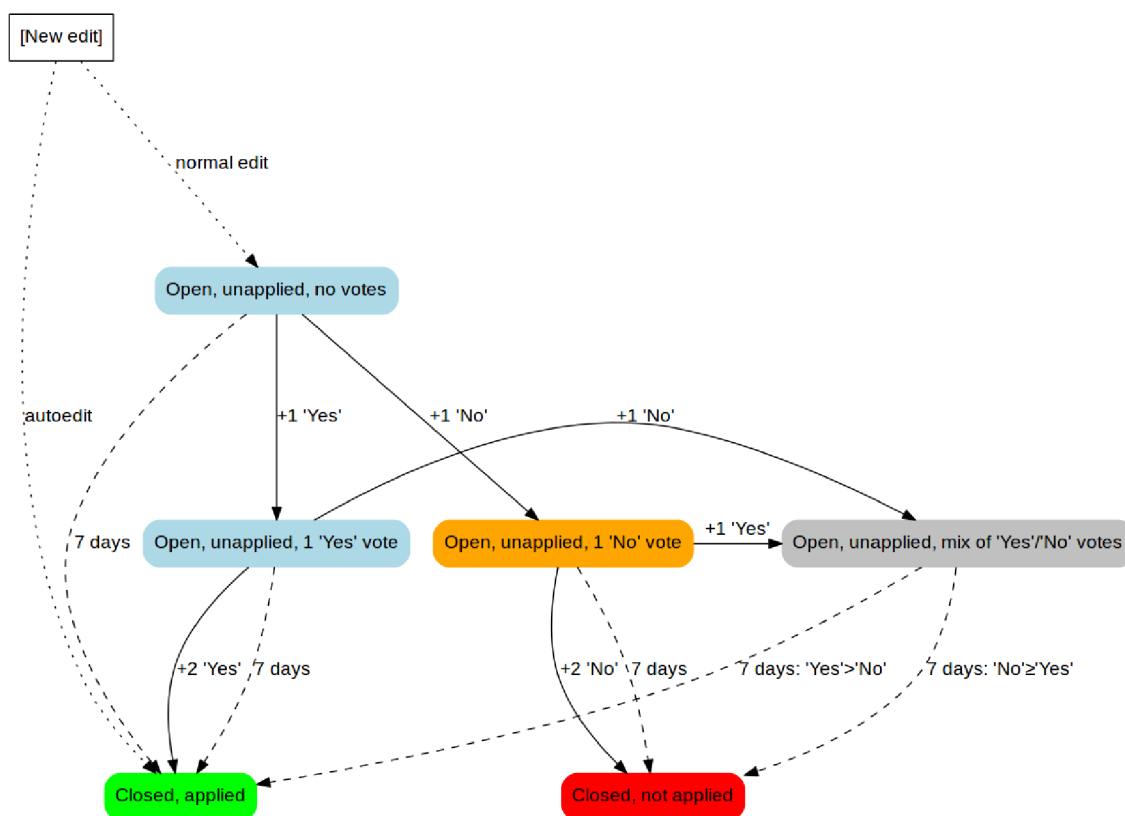
#### MusicBrainz

Cílem stránky MusicBrainz je být online encyklopedií, která je zdarma. Nápad pro vytvoření této stránky vznikl díky databázím Compact Disc Database (CDDDB)<sup>1</sup> a FreeDB. CDDDB původně fungovala také zdarma a jak její název napovídá, byla zaměřená na CD. Později se stala komerční a vznikla FreeDB, která fungovala na poslední verzi CDDDB, jenž byla přístupná zdarma. Ovšem každý záznam v této databázi byl uložen jako samostatný soubor. Proto přišlo MusicBrainz s komplexní relační databází, a navíc se systémem ověřování a hodnocení vložených příspěvků. To především z důvodu, aby nepřevyšovala kvantita nad kvalitou.

---

<sup>1</sup>Compact Disc Database, která přispěla k nápadu založení databáze MusicBrainz – <https://www.oasiscd.com/cddb/>

Volebního systému MusicBrainz se nemůže zúčastnit každý nově registrovaný uživatel, ale je potřeba aby měl alespoň 14 dní starý účet a alespoň 10 schválených úprav příspěvků ostatních uživatelů. Systém úprav je také značně propracovaný. Jeho zjednodušená verze je zobrazena na obrázku 3.1. Pokud někdo provede úpravu, zůstane tato úprava nerozhodnutá po následujících 7 dní. Výjimkou je, pokud ji potvrdí takzvaný auto-editor, což je uživatel se speciálními právy. Nebo se může stát, že do té doby dostane úprava 3 po sobě jdoucí kladná hodnocení a žádná záporná, čímž je úprava přijata, naopak při 3 záporných hodnoceních a žádných kladných je zamítnuta. Pokud se nestane ani jeden z těchto případů, po 7 dnech je úprava vyhodnocena a pokud má více kladných hodnocení, je brána za platnou.



Obrázek 3.1: Systém hodnocení úpravy příspěvku internetové databáze MusicBrainz. Pokud je zadána úprava, ubíhá časomíra 7 dní, během kterých může dojít k následujícím akcím. Úprava je potvrzena uživatelem se speciálními právy (auto-editor). Nebo úprava dostane 3 po sobě jdoucí hodnocení, kladná nebo záporná a je buď přijata nebo zamítnuta. Pokud se nestane žádná z těchto akcí, po 7 dnech je úprava vyhodnocena a musí mít alespoň o 1 kladný hlas více, než hlasů záporných. Převzato z [5].

Základním kamenem databáze je entita vydání (release). Na ni jsou vázány ostatní entity, jako je interpret (artist), média (medium) na kterých je konkrétní vydání k dostání, oblast (area) vydání a značka (label) pod kterou vydání proběhlo. Jednotlivá vydání mohou být součástí skupiny vydání (release group), která seskupuje podobná vydání. Například vydání obsahující stejné skladby a stejného interpreta. Informace o formátu vydání se nachází v entitě medium. MusicBrainz podporuje celou řadu různých formátů od těch, se kterými

se v dnešní době běžný člověk nesetká (diskety, VHS) až po ty nejmodernější (Blu-ray). Vinylové desky jsou rozlišeny na základě jejich velikosti, tedy 7", 10" a 12" desky.

Databáze umožňuje veřejnosti k ní přistoupit hned několika způsoby. Buď přes jejich internetové stránky, kde je běžný vyhledávač. Vyhledávání používá indexované hledání SOLR což je open-source nástroj od firmy Apache, který nabízí pokročilé vyhledávání. Při vyhledávání na MusicBrainz lze také využít speciální syntaxi výrazů. Další možností je použití veřejného API Webservice<sup>2</sup>, které umožňuje vývojářům se ze svých programů dotazovat na data uložená v MusicBrainz databázi a získávat její obsah v podobě formátu JSON nebo XML. Na tento typ vyhledávání se vztahuje jediné omezení, že lze provést pouze 50 dotazů za sekundu (čehož dle dokumentace dosud nebylo dosaženo).

## Discogs

Z databází uvedených v této práci je Discogs rozhodně největší, co se týče obsahu. Lze v ní dohledat téměř 13 milionů nahrávek a přes 7 milionů umělců. Umělec nemusí být nutně interpretem, jelikož mezi ně spadají i osoby, které se podíleli na vzniku nahrávky i jiným způsobem (producent, designér, střihač apod.). Svoji komunitu se snaží rozšířit i nabízením služby na způsob tržnice (marketplace), kde uživatelé mohou obchodovat s nahrávkami které vlastní.

Discogs také obsahuje systém založený na hlasování. Stejně jako u MusicBrainz nemůže hlasovat každý nový uživatel, ale musí právo volit získat. To lze obecně jakoukoliv aktivitou, jako přidávání komentářů, pravidelným navštěvováním databáze, nebo přidáváním kvalitních vlastních verzí nahrávek. Obecně musí mít pro platformu nějaký přínos. Přičemž mu toto právo může být kdykoliv odebráno.

Discogs nabízí i svoji aplikaci na chytré telefony, konkrétně na operační systémy iOS a Android. Ta zpřístupňuje veškeré funkce desktopové verze webových stránek v přehledném provedení. Do aplikace byla také poměrně nedávno integrována funkce, pomocí které lze skladby přehrát na Apple Music. Pokud uživatel není předplatitelem Apple Music, může si přehrát pouze ukázkou skladby.

Strukturou je databáze Discogs velice podobná databázi MusicBrainz. Alespoň z uživatelského hlediska. Přesná podoba databáze není nikde zdokumentována a lze se jen domnívat jakou má podobu. Několik vodítek obsahuje například dokumentace s postupem pro vkládání nových příspěvků. Co se týče formátů vydání, rozlišuje jich podobné množství jako MusicBrainz. Částečně jde ale o něco více do hloubky, protože u jednotlivých formátů se nachází ještě informace s popisem. U vinylových desek to znamená, že nejsou rozlišovány pouze na základě velikosti, ale i rychlosti otáček a délky hraní.

Databáze nabízí přístup k datům pomocí webových stránek, nebo veřejného API. Discogs API vrací výsledky ve formátu JSON. Má však jednu velkou nevýhodu a tou je, omezení maximální počet poslaných dotazů za minutu. Pokud je uživatel nebo aplikace ověřena, umožňuje odeslat až 60 dotazů za minutu. V opačném případě pouze 25 dotazů. Informace o zbývajícím počtu možných dotazů v minutovém okně je vracena v hlavičce odpovědi, a je na uživateli aby si ohlídal, že tuto hodnotu nepřekročí. Jinak API místo dat vrací chybovou hlášku.

---

<sup>2</sup>Veřejné API databáze MusicBrainz – [https://musicbrainz.org/doc/MusicBrainz\\_API](https://musicbrainz.org/doc/MusicBrainz_API)

## 45cat/45worlds

Poslední databázi, respektive databázemi, které stojí za zmínění v souvislosti s vinylovými deskami, jsou 45cat a 45worlds. Stránka 45cats je zaměřená pouze na vinylové desky s rychlostí otáček 45 RPM. Naopak 45worlds nabízí několik podkategorií zaměřené na jednotlivé formáty. Mezi nimi jsou CD, 12" singly, 78 RPM desky, nebo DVD & Blu-ray. Lze zde nalézt i záznamy o knihách a časopisech. Tyto databáze mají o poznání méně příspěvků než předchozí dvě databáze. I přes to se stále jedná o relevantní zdroj informací.

Neobsahují ani tak propracovaný systém přidávání příspěvků. Existují zde sice také přísná pravidla pro jejich vkládání, kterými se uživatelé musí řídit, ale je zde absence systému úprav a hodnocení příspěvků. Lze je hodnotit pouze slovně ve formě komentářů a recenzí. Nejvíce je využíváno fóra do kterého může přispět jakýkoliv registrovaný uživatel.

Po obsahové stránce toho nabízí také mnohem méně. Například se v databázi nevyskytují žádné informace o společnostech, které nahrávky vydaly, nebo různých identifikátorech. Jediné informace, které mají vlastní umístění jsou:

- Interpret
- Název vydání
- Značka (label)
- Země původu
- Katalogové číslo
- Formát
- Seznam skladeb
- Poznámky

Jakékoliv další informace mohou být umístěny pouze do pole poznámek. Ty zpravidla opravdu obsahují spoustu zajímavostí a detailů o nahrávce. Ovšem chybí zde přehlednost, jelikož poznámky jsou zobrazeny formou neformátovaného textu. Je nabízena alespoň jedna funkce, která není u předchozích databází. Jedná se o možnost vyhledat vydání na stránkách Amazon a eBay. Zde také bývá zobrazena orientační cena.

Jediným možným způsobem, jak v databázi vyhledávat je přes webové stránky. Lze sice vyhledat na základě různých údajů jako je název interpreta, název skladby, název nahrávky, země původu apod., ale jsou vráceny pouze přesné shody s hledaným výrazem. Jako příklad lze uvést vyhledání skladby dle jejího názvu, kdy uživatel nejdříve zadá výraz *day* a poté výraz *days*. V prvním případě jsou nalezeny skladby, kde se v jejich názvu toto slovo vyskytuje alespoň jednou. To samé platí i pro druhý případ, což se na první pohled může zdát v pořádku. Ovšem při porovnání obou výsledků hledání, je možné zjistit, že mezi těmito dvěma vyhledáními neexistuje žádný průnik. Tedy první vyhledávání neobsahuje žádné záznamy z druhého a naopak. Uživatel tedy musí být při zadávání výrazu, naprosto přesný. Navíc nelze kombinovat více kritérií pro vyhledávání, nebo hledat podle roku.

## 3.2 Nástroje pro práci s metadaty

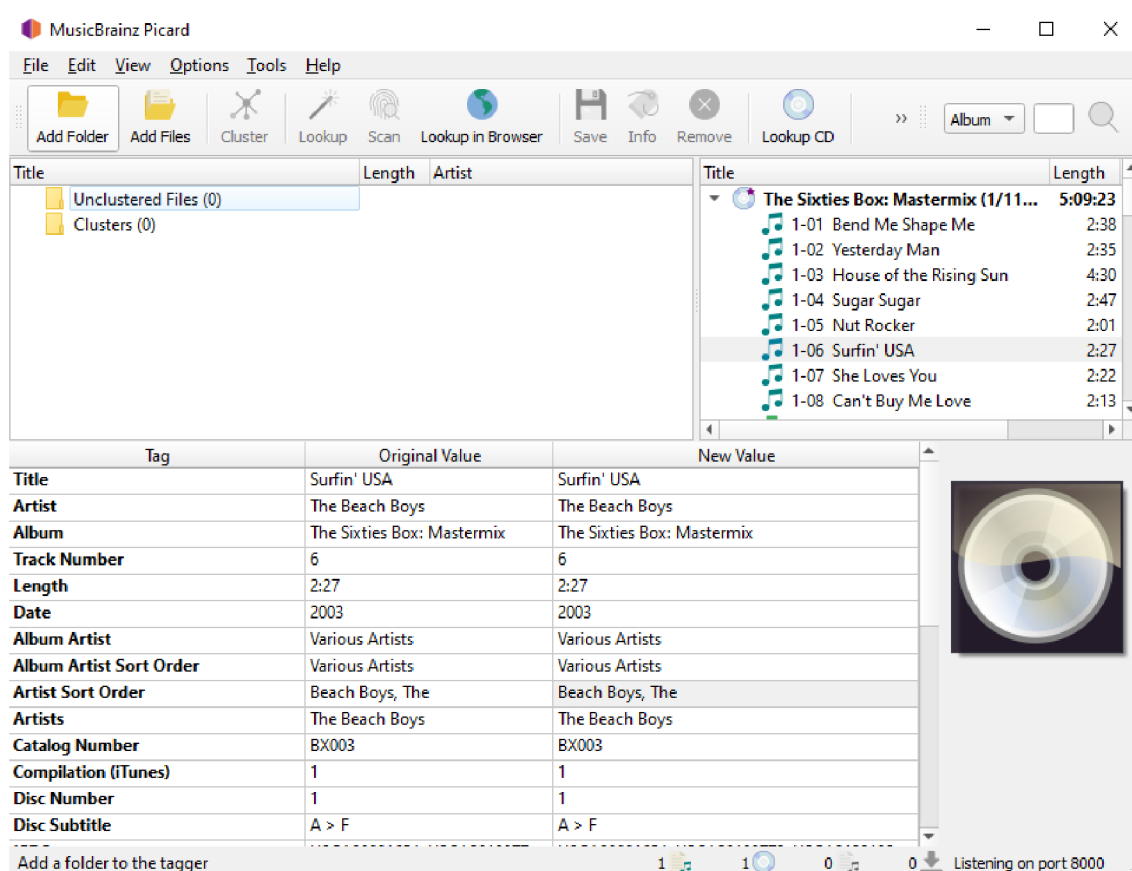
Mezi nástroji pro práci s metadaty se v hojné míře vyskytují především různé aplikace na tagování hudebních souborů nebo správu vlastní hudební knihovny. Pokročilejší z nich

nabízejí i dohledání metadat v některých internetových databázích, mezi kterými bývají právě výše zmiňované Discogs a MusicBrainz.

## MusicBrainz Picard

MusicBrainz Picard je multiplatformní tagovací nástroj hudebních souborů. Podporuje širokou škálu formátů. Další velice užitečnou funkcí je podpora AcousticID<sup>3</sup>, což je open-source software, který je schopen ze zvukové nahrávky vytvořit její otisk (audio fingerprint). Obsahuje vlastní databázi, která je také založená na příspěvcích od uživatelů. Tato databáze obsahuje mnoho záznamů, které jsou spojeny právě s metadaty z databáze MusicBrainz.

Mezi další využitelné funkce je dohledání kompletních informací CD, ze kterého pochází vybraná skladba. Dohledané informace lze případně upravit a poté vložit do tagů hudebních souborů. Případně lze přímo z aplikace přejít na stránky MusicBrainz, kde jsou vyhledané metadata zobrazeny, s některými informacemi navíc.



Obrázek 3.2: MusicBrainz Picard, multiplatformní software na tagování hudebních souborů. Podporuje dohledání metadat v databázi MusicBrainz. Také podporuje AcousticID, což je nástroj na vytváření a vyhledávání otisků (audio fingerprint) hudebních souborů.

<sup>3</sup>AcousticID je software pro identifikaci hudebních souborů na základě jejich zvukového otisku (audio fingerprint) – <https://acoustid.org/>

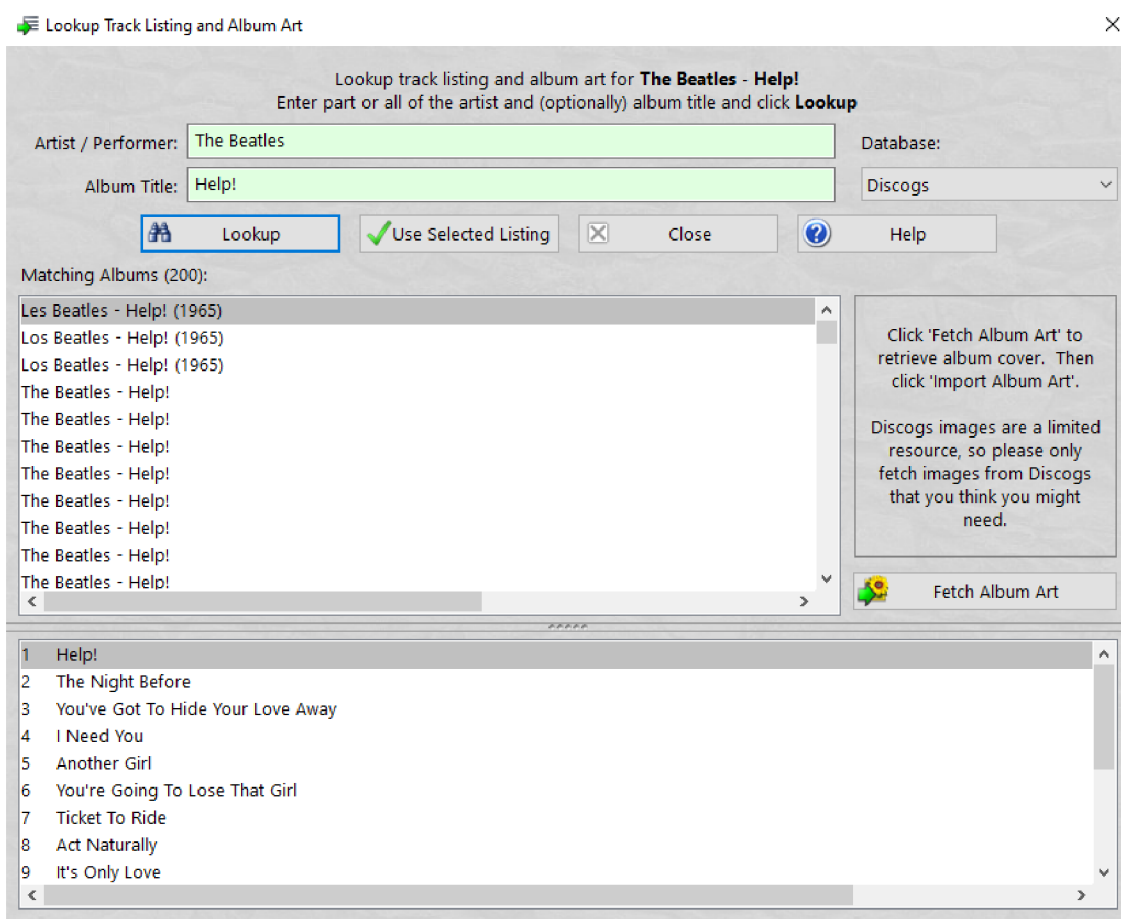
## VinylStudio

Jak již z názvu vypovídá, VinylStudio je software zaměřený na vinylové nahrávky. Jedná se o multifunkční nástroj, který slouží především pro digitalizování vinylových desek. Aplikace je multiplatformní, konkrétně podporuje operační systémy Windows a Macintosh. Nahrávání má spoustu užitečných nastavení. Lze například převést nahrávku z rychlosti 78 RPM na 45 RPM. Přičemž samozřejmě může dojít ke zhoršení kvality zvuku.

Skladby lze nahrát naráz a poté je rozdělit, bez nutnosti používat jinou aplikaci, protože VinylStudio obsahuje i stříhací nástroj, který je schopen detekovat ze zvukové stopy jednotlivé skladby. Lze také nastavit různé dlouhé odmlky na začátku nebo konci skladby.

Další velice užitečnou funkcí je přítomnost zvukových filtrů. Pomocí nich je možné z nahrávky odstranit nežádoucí zvuky, které mohly vzniknout při procesu nahrávání. Ze zvukové stopy je možné odstranit i šum. Jsou zde přítomny i speciální filtry, které slouží především pro vylepšení zvuku, pokud originální deska byla již opotřebovaná nebo znečištěná.

Aplikace také umí k dané nahrávce dohledat seznam skladeb a fotografii alba. Bohužel ostatní metadata již musí uživatel dohledat nebo zadat ručně.



Obrázek 3.3: VinylStudio, multiplatformní aplikace s širokou nabídkou funkcí. Zde lze vidět možnost vyhledání seznamu skladeb a fotografii alba z databáze Discogs. Další databází která je na výběr je MusicBrainz

## Kapitola 4

# Návrh aplikace

Tato kapitola obsahuje návrh aplikace, která by měla být schopná dohledat informace o originálních nahrávkách vinylových desek. Pro lepší upřesnění, a odkazování se na vytvářenou aplikaci, je bude dále označována pracovním názvem **VinylFinder**. Nejdříve je aplikace popsána obecně. Tedy komu je aplikace převážně určená a jaké funkce by měla nabízet. Dále jakým způsobem a ze kterých zdrojů budou informace dohledávány. Na konec jakým způsobem je sjednocena podoba těchto zdrojů v jednu lokální databázi.

### 4.1 Základní specifikace

Mělo by jít o desktopovou aplikaci, která by přes jednoduché uživatelské rozhraní zpřístupňovala uživateli všechny její funkce. Vzhledem k jejímu úzkému zaměření pouze na vinylové desky, a především tedy dohledávání podrobných informací o nahrávkách, je očekáváno, že osloví pouze malou skupinu uživatelů právě s těmito potřebami. Předpokládaný typ uživatele, pro kterého tato aplikace bude určena je nadšenec/sběratel vinylových desek. Tomu by měla umožnit dohledat informace na základě názvu interpreta a názvu skladby.

V závěru předchozí kapitoly, konkrétně v sekci 3.2, která se zabývala již existujícími aplikacemi, byl zmíněn nástroj VinylStudio sloužící mimo jiné na digitalizaci vinylových kolekcí. Umožňuje sice dohledání informací z některých databází, ovšem ve velice omezené míře. VinylFinder by mohl sloužit jako pomocník v kombinaci s nástrojem VinylStudio, kdy uživatel svoji vinylovou kolekci nejdříve digitalizuje, a poté se pokusí o ní nechat dohledat dodatečné informace.

Z uvedených příkladů již lze mít základní představu o tom, pro jakého uživatele je aplikace určena a jakým způsobem ji bude používat. Přehled hlavních akcí, které by měla aplikace uživateli umožňovat se nachází na obrázku 4.1.

Dále vzhledem k množství detailů, které mohou být dostupné pouze na potisku desky, případně na jejím obalu, by měla umožňovat získání a uložení obrázků. Ty by následně uživatel mohl pečlivě prozkoumat a dohledávat indicie k určení konkrétní nahrávky.

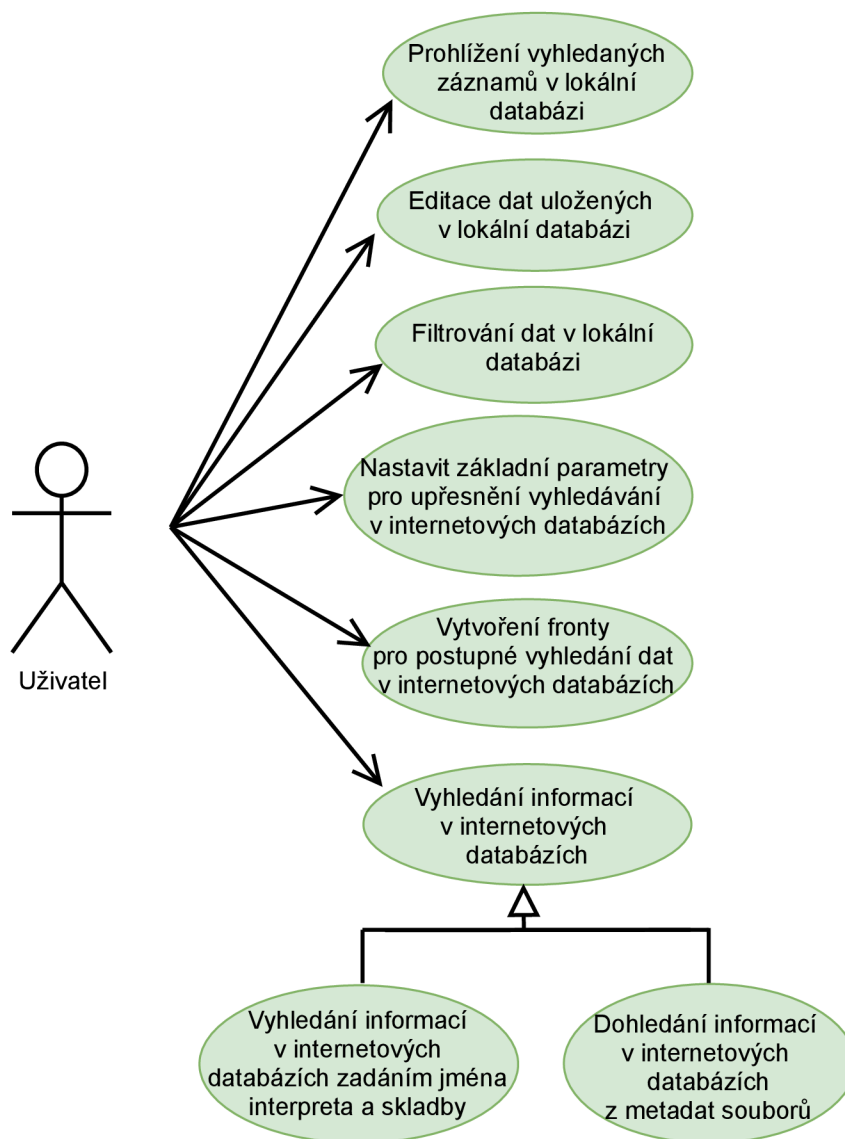
### Požadavky na aplikaci

Hlavní požadavky na aplikaci lze sepsat do několika hlavních bodů. Některé z nich již byly zmíněny výše, ostatní budou ještě dále rozepsány. Tyto body jsou následující:

- Vyhledávání informací z více zdrojů
- Zadání názvu ručně, či získání z metadat souboru



- Možnost upřesnit hledání, dodatečnými podmínkami
- Uchování dohledaných dat v lokální databázi
- Získání obrázků nahrávek
- Filtrace obsahu databáze dle různých parametrů
- Uživatelsky přívětivé zobrazení dat z databáze



Obrázek 4.1: Diagram případů užití ilustrující hlavní akce, které by měl uživatel být schopný s aplikací provádět

## 4.2 Získávání dat z hudebních databází

Pro dosažení co nejlepších výsledků hledání informací o nahrávkách, je zapotřebí nespolehat se pouze na jeden zdroj, ale pokusit se jich pokrýt co nejvíce. V sekci 3.1 byly detailně popsány existující komunitní databáze o hudebních nahrávkách. Vzhledem k tomu, že jsou tvořeny komunitou, lze předpokládat, že každá z nich bude obsahovat odlišné, a především různě detailní údaje. Pokud se někdo rozhodne do takové databáze přispívat, pravděpodobně si vybere jednu, která mu vyhovuje. Z toho důvodu budou použity všechny, konkrétně tedy Discogs, MusicBrainz, 45cat a 45worlds.

V případě Discogs a MusicBrainz, lze využít jejich veřejných API, skrze které umožňují v jejich databázi vyhledávat podle různých kritérií a výsledky posílají v dobře zpracovatelných formátech, ať už se jedná o XML, či JSON. Navíc dle jejich statistik dohromady obsahují záznamy o nahrávkách od téměř 10 milionů interpretů<sup>1</sup>.

Za to 45cat a 45worlds žádné takové API nemají a do počtu záznamů se předchozím databázím také nemohou rovnat. Ale minimálně 45cat by stále mohla být relevantním zdrojem informací, především z důvodu, že je zaměřená právě pouze na vinylové desky. Pro získávání dat ovšem bude zapotřebí použít zcela odlišné techniky, kterou je takzvaný web scraping.

Web scraping je pojem, který označuje získávání dat z webových stránek. Většina webových stránek má pevně danou strukturu. Toho faktu lze využít a po zanalyzování této struktury z nich získávat potřebné informace. Jedná se ovšem o získání pouze užitečné informace a ne stahování, respektive kopírování veškerého obsahu, co stránky obsahují. Konkrétní data mohou být dohledána pomocí identifikátorů a tříd jednotlivých HTML tagů a za pomoci odkazů lze jednoduše přecházet na další části daného webu. Tímto způsobem lze docílit podobného efektu, jako by se po stránkách pohyboval samotný uživatel.

## 4.3 Návrh lokální databáze

Každá z výše uvedených databází přistupuje různě k jednotlivým údajům o nahrávkách. Jako příklad lze uvést jakým způsobem lze přistoupit skrze API k informaci o formátu nahrávky nebo seznamu skladeb. V případě Discogs je centrálním bodem celé databáze entita Release (nahrávka). Všechny informace se vztahují právě k ní a nejsou v podstatě nijak více členěny. Daná nahrávka má tedy pevně daný formát (medium), na kterém byla vydána a seznam skladeb. Při pohledu na strukturu databáze MusicBrainz je vidět podstatný rozdíl. Databáze je velmi členitá a některé údaje zpřístupňuje skrze další tabulky<sup>2</sup>. Nahrávka tedy může mít různé formáty, kde jednotlivé formáty mohou obsahovat rozdílné skladby. Z toho důvodu je potřeba navrhnout lokální databázi takovým způsobem, aby umožňovala uchovávat data ze všech zdrojů. A to ideálně ve sjednocené formě.

Vzhledem k zaměření aplikace právě na nahrávky, byl zvolen podobný přístup jako u databáze Discogs. Nejdůležitější entitou je tedy nahrávka. O ní je potřeba uchovávat informace o názvu, roku vydání a katalogové číslo. Mohou u ní být i dodatečné poznámky, které mohou upřesňovat některé zajímavé detaily. Poznámky se mohou vyskytnout ve všech zdrojových databázích, největší význam mají ale v případě databází 45cat a 45world, kde v nich často bývají uloženy i různé identifikátory, protože pro ně nenabízí speciální položky.

<sup>1</sup>Údaje o počtu záznamů v databázi Discogs – <https://www.discogs.com/about> a statistiky databáze MusicBrainz – <https://musicbrainz.org/statistics>

<sup>2</sup>Kompletní schéma databáze MusicBrainz – [https://musicbrainz.org/doc/MusicBrainz\\_Database/Schema](https://musicbrainz.org/doc/MusicBrainz_Database/Schema)

Pro uchování informace o původu nahrávky, respektive v jaké databázi byla dohledána slouží atribut název zdroje a je uchováván i identifikátor ze zdrojové databáze. V případě 45cat a 45world může jako identifikátor zdrojové databáze posloužit URL adresa.

Jednotlivé nahrávky mohou obsahovat skladby od jednoho nebo více interpretů. Interpret tedy bude další entitou a bude stačit uchovávat jeho název. Stejně tak i v případě žánru, formátu a země ve které byla nahrávka vydána. Dále se na nahrávce mohla podílet i nějaká společnost, například vydavatelství, nahrávací studio a další. Kromě společnosti mohli při tvorbě nahrávky figurovat i další lidé jako dirigent a spisovatel. Společnost a člověk, co se nějak podílel na vytvoření nahrávky budou další dvě entity figurující v databázi. O těch bude uchováván název a jejich role.

Dalším neméně důležité údaje, které mohou o nahrávce mnoho prozradit jsou různé identifikátory. Například matriční číslo, nebo čárkový kód. Vzhledem k velkému množství možností, o jaký identifikátor se může jednat, bude potřeba o něm uchovávat jeho typ, tedy o jaký identifikátor se jedná. Dále hodnotu identifikátoru, kde by bylo uvedeno konkrétní matriční číslo a nakonec poznámka, která by mohla o identifikátoru prozradit více.

Každá nahrávka pak nese informaci o labelu. Každý label má název a také unikátní kód, který ovšem ne vždy bývá v databázích uveden. V poslední řadě, je nutné uchovávat jednotlivé skladby, které nahrávka obsahuje. Ty mají vždy název, délku trvání a pozici, která označuje, na které straně desky a případně v kolikátém pořadí se na desce nachází. Dále mohou být uvedeni i další lidé, kteří se zasloužili o nahrání dané skladby. To mohou být samotní členové kapely, textaři a další. Takovýto umělec bude také reprezentován zvláštní entitou, ale v tomto případě bude vázán na konkrétní skladbu místo nahrávky.

## Kapitola 5

# Výsledný nástroj

V této kapitole se lze dočíst, jaké technologie byly použity pro vytvoření výsledné aplikace. Dále se zaměřuje na popis samotné implementace, především se zaměřením na detaily jak probíhá vyhodnocení nalezení nahrávky a jak je implementováno vyhledávání v jednotlivých hudebních databázích.

### 5.1 Použité technologie

V této sekci se nachází popis jednotlivých technologií, které byly použity pro vytvoření finálního nástroje. Případně je zmíněno, k čemu se konkrétně používá.

#### Python

Python<sup>1</sup> je dynamický, interpretovaný, objektově orientovaný programovací jazyk. Je vyvíjen jako open-source (s otevřeným zdrojovým kódem) a je přenositelný mezi většinou široce rozšířených operačních systémů. Mezi jeho výhody se řadí jednoduchost, množství existujících knihoven a čitelnost zdrojového kódu. Konkrétně je použita verze 3.7.9 a to především z důvodu nekompatibility novějších verzí s použitými nástroji.

#### Python3 Discogs Client

Python3 Discogs Client<sup>2</sup> je oficiální knihovna Discogs API pro jazyk Python. Nabízí rozhraní pro získávání dat z databáze Discogs a přistupovat k nim v podobě objektů. Implementuje jednotlivé objekty jako je například interpret nebo nahrávka a zpřístupňuje jejich data v podobě instančních atributů. Podstatnou výhodou této knihovny je fakt, že do jisté míry zajišťuje aby nebyl překročen limit dotazů na Discogs API.

#### MusicBrainzGs

MusicBrainzGs<sup>3</sup> je knihovna, která mapuje koncové body MusicBrainz API (MusicBrainz Web Service 2) na metody jazyka Python. Nabízí rozhraní pro identifikaci vůči API a zjednodušuje práci se získanými daty.

---

<sup>1</sup>Python – <https://www.python.org/>

<sup>2</sup>Python3 Discogs Client – [https://github.com/joalla/discogs\\_client](https://github.com/joalla/discogs_client)

<sup>3</sup>MusicBrainzGs – <https://python-musicbrainzngs.readthedocs.io/en/v0.7.1/>

## Requests

Requests<sup>4</sup> je velice rozšířená knihovna, která umožňuje jednoduché zasílání HTTP požadavků. Kromě toho nabízí i pokročilejší techniky a úpravu jednotlivých dotazů. Díky tomu je ideálním nástrojem pro získávání obsahu z webových stránek.

## Mutagen

Mutagen<sup>5</sup> je nástroj sloužící pro zapisování a čtení metadat z hudebních souborů. Podporuje celou řadu rozšířených tagů a navíc je jednoduchý na použití.

## SQLAlchemy

SQLAlchemy [1] je sada databázových nástrojů, pro programovací jazyk Python. Nabízí dva hlavní způsoby použití, kde první z nich je označován jako SQLAlchemy Core. Ten nabízí pokročilé možnosti pro pracování s databázemi. Je vytvořen nad databázovým rozhraním jazyka Python (DBAPI) a umožňuje psaní SQL dotazů jako u běžných relačních databází.

Druhý přístup je nazýván SQLAlchemy ORM, který nabízí využít objektové relačního mapování (ORM). Jedná se o další vrstvu abstrakce, která je vytvořena nad SQLAlchemy Core. Umožňuje automatický převod dat mezi relační databází do objektu programovacího jazyka Python. Díky tomu je pak možné se záznamem z databáze pracovat jako s plnohodnotným objektem a jednoduše získávat data skrze atributy objektu.

## PySide2

PySide2<sup>6</sup> nebo také někdy nazýván Qt for Python tvoří vazbu Pro jazyk Python s frameworkem Qt původně vytvořeným pro jazyk C++. Jedná se o multiplatformní sadu nástrojů pro tvorbu uživatelských rozhraní.

## 5.2 Vyhodnocení nalezení shody

Hlavním úkolem aplikace VinylFinder, je dohledání informací z hudebních databází. Proto na tuto část implementace byl kladen velký důraz. Prioritou není, aby hledání bylo rychlé, ale naopak důkladné. Tato sekce je zaměřená na způsob určení, že byla nalezena hledaná nahrávka a další možné parametry, kterými lze ovlivnit výsledky vyhledávání.

### Porovnávání hledaného výrazu

Aby bylo vyhledávání přesnější je zapotřebí použít pokročilejší techniky pro porovnávání hledaného a zadaného výrazu. V datech obsažených v hudebních databázích mohou být různé odlišnosti nebo dokonce chyby. Některé databáze například obsahují více záznamů pro jednoho interpreta. Ať už z důvodu, že skutečně existují dva interpreti se stejným názvem, nebo z jakéhokoliv jiného. Tito interpreti mohou být poté odlišeni například číslovkou. Příkladem z databáze Discogs jsou interpreti *Carmel* a *Carmel (2)*. O této skutečnosti samozřejmě uživatel aplikace VinylFinder nemůže mít tušení. Pokud by chtěl vyhledat informace o skladbě od interpreta Carmel (2), zadal by do vyhledávání výraz Carmel. Proto

---

<sup>4</sup>Requests – <https://docs.python-requests.org/>

<sup>5</sup>Mutagen – <https://mutagen.readthedocs.io/en/latest/>

<sup>6</sup>PySide2 – <https://doc.qt.io/qtforpython-5/>

při porovnávání výrazu není používáno běžného porovnání řetězců, kdy by musely být řetězce naprosto shodné, ale je použita Levenštejnova vzdálenost [2]. Jedná se o metriku, která nad řetězci připouští operace vložení, smazání a nahrazení znaku. Výsledkem je potom nejmenší počet takovýchto operací, které je nutné nad řetězcem provést, aby se rovnal druhému řetězci. Konkrétně je použita implementace z knihovny FuzzyWuzzy<sup>7</sup>, kde je využito poměru podobnosti výrazů. Tou je hodnota od 0 do 1, kde hodnota 1 znamená, že jsou řetězce totožné.

## Nastavení parametrů vyhledávání

Aby měl uživatel možnost ovlivnit výsledky vyhledání i jiným způsobem, než pouze zadáním názvu interpreta a skladby, jsou v nastavení aplikace dostupné tyto položky:

- Počet let navíc od nejstarší nalezené nahrávky
- Hranice pro vyhodnocení shody s hledaným výrazem
- Nejvyšší rok nahrávky, která má být vyhledána

První položka ve většině případů ovlivňuje počet dohledaných nahrávek a také čas nutný pro hledání. Pokud je například první nalezená nahrávka z roku 1970 a tato hodnota by byla nastavena na 1, výsledkem hledání mohou být pouze nahrávky z let 1970 a 1971. Vzhledem k hlavnímu cíli aplikace, dohledat originální nahrávky, je výchozí hodnota nastavena na 1. To z důvodu aby, byl prostor pro nalezení většího počtu nahrávek. Ovšem pokud by uživatel chtěl striktnější vyhledávání, nebo naopak větší rozsah let, je mu to umožněno.

Další položkou je hranice pro vyhodnocení shody hledaného a nalezeného výrazu. Testováním bylo zjištěno, že ideální hodnotou je 75, kdy jsou ignorovány nerelevantní výsledky, a zároveň jsou brány v potaz možné odchylky názvů způsobené tím, v jaké podobě jsou data v databázích uchovávána. Ovšem stále je možnost, že by uživatel mohl chtít si tuto hodnotu upravit svým potřebám.

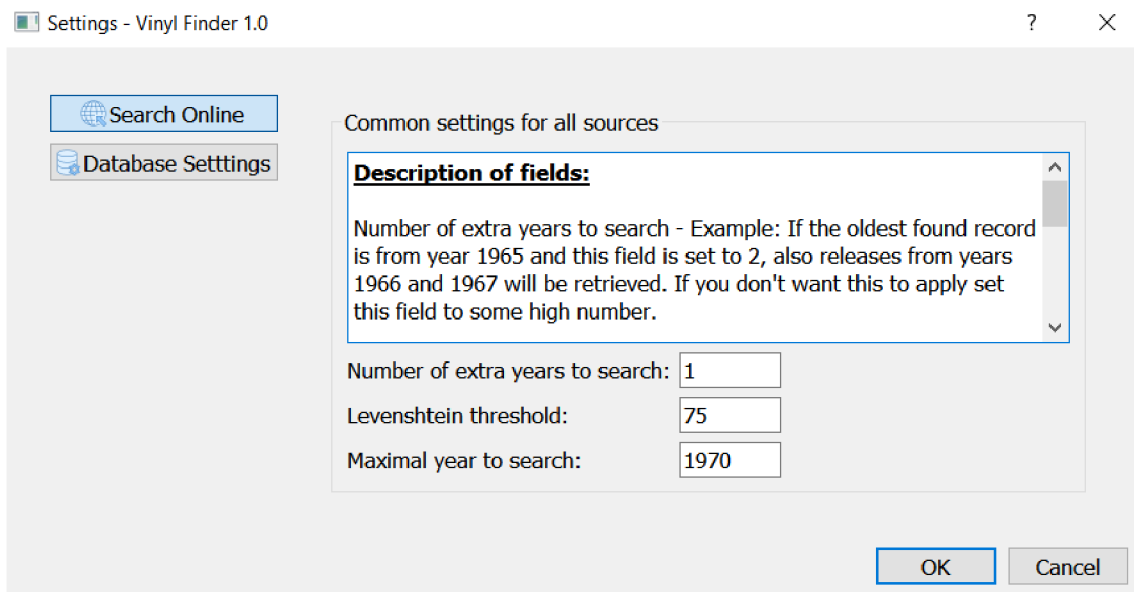
Poslední položkou v nastavení, je nejvyšší rok, který má být obsažen ve výsledcích hledání. Tato hodnota může být použita v různých kombinacích s první položkou a tím upravovat interval let nalezených nahrávek. Pokud by uživatel měl konkrétnější představu o roku nahrávky, může touto hodnotou vyhledání zúžit. Například zadáním hodnoty 1980 zajistí, že se ve výsledcích vyhledávání neobjeví nahrávky z let 1981 a pozdějších.

Podoba nastavení, přímo v prostředí aplikace VinylFinder, je zobrazena na obrázku 5.1.

## 5.3 Implementace vyhledávání v hudebních databázích

Pro jednotlivé zdroje, tedy hudební databáze jsou implementovány vlastní třídy. Vyhledávání z databáze Discogs zajišťuje třída **DiscogsDownloader.py**, pro MusicBrainz třída **MusicBrainzDownloader.py** a pro databáze 45cat a 45worlds je společná třída **WebScrapper.py**. Ačkoliv každá z těchto tříd využívá trochu jiných knihoven a postupů pro získání dat, mají několik společných vlastností. Proto všechny tyto třídy dědí ze společného základu, kterou je třída **BaseDownloader.py**. Zjednodušenou podobu dědičnosti ilustruje obrázek 5.2.

<sup>7</sup>Knihovna FuzzyWuzzy použitá pro porovnávání řetězců při vyhledávání <https://github.com/seatgeek/fuzzywuzzy>



Obrázek 5.1: Nastavení parametrů vyhledávání v aplikaci VinylFinder, které umožňuje uživateli získat specifitější nebo naopak obecnější výsledky

## BaseDownloader

Ve třídě BaseDownloader, přesněji v každé instanci této třídy jsou uchovávány informace o názvu interpreta a skladby, které mají být vyhledány. Dalším atributem je session z knihovny Requests, protože i když jsou pro získání dat z některých zdrojů využívány zvláštní knihovny, je nutné v určitých případech získat data zasláním HTTP dotazu. Například pro stažení obrázku, kde ani jedna z knihoven nenabízí rozhraní pro jejich stažení.

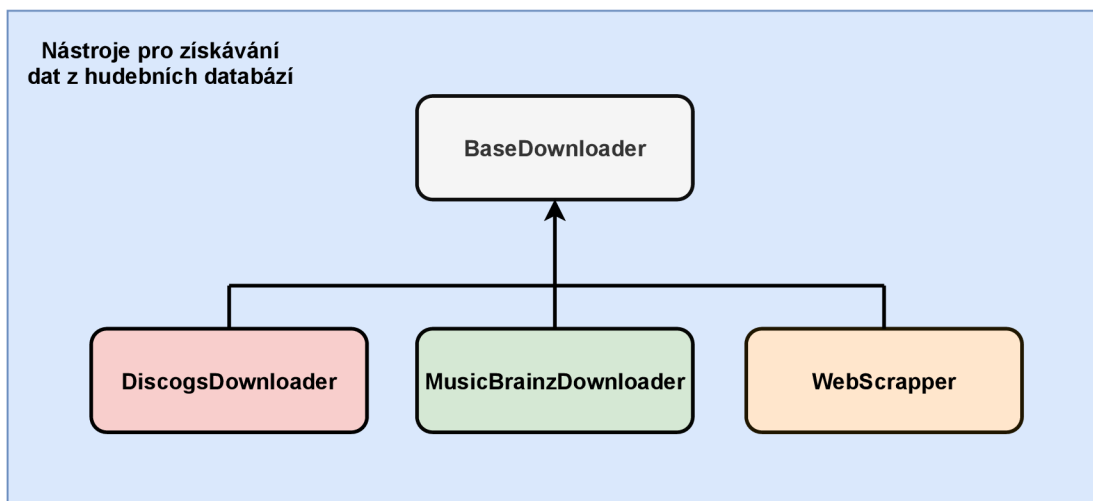
Dalšími atributy jsou nastavitelné parametry vyhledávání, které jsou nastavitelné uživatelem a byly popsány výše v této sekci.

Dále třída implementuje metody pro ukládání dat do lokální databáze. A signály aplikace pro indikaci, že nastala chyba, anebo bylo dokončeno vyhledávání.

## DiscogsDownloader

Tato třída implementuje vyhledávání v databázi Discogs, k čemuž je využíváno knihovny python3-discogs-client. Pro využití plného potenciálu Discogs API je nutné se na jejich stránkách zaregistrovat a při dotazování na API posílat některý z přiřazených identifikátorů pro ověření. Možností ověření je více, aplikace VinylFinder používá vygenerovaný uživatelský token.

Pro zahájení vyhledávání stačí, aby byl k dispozici název interpreta a skladby. Postup hledání je poté následující. Nejdříve se na Discogs API pošle dotaz, který vrátí seznam interpretů, který funguje na principu stránkování. Poté jsou procházeny názvy všech interpretů na stránce a porovnávány s názvem hledaného interpreta. Pokud je vyhodnocena shoda, ještě se zkontroluje, zda už se interpret nevyskytl ve výsledcích dříve, aby nebyly duplikované informace, a nakonec se záznam uloží do pole **possibleArtists** pro pozdější zpracování. Možnost výskytu duplikovaných informací byl zjištěn na základě pozdějších testů popsaných dále v sekci 6. Další věcí zjištěnou při testování, byl fakt, že na násle-



Obrázek 5.2: Třídy zajišťující vyhledání a stáhnutí dat z internetových hudebních databází

dujících stránkách jsou vyhledáni interpreti, kteří jsou vůči hledanému výrazu naprosto irelevantní, a proto se prochází pouze první dvě strany výsledků.

Dalším krokem je projití nahrávek jednotlivých interpretů, u kterých byla vyhodnocena shoda. Seznam nahrávek je stejně jako interpreti reprezentován jednotlivými stránkami, kde nahrávky jsou seřazeny od nejnižšího roku. V databázi Discogs jsou navíc dva druhy nahrávek. Běžné nahrávky reprezentující konkrétní nahrávku a speciální druh nahrávky takzvaný *Master release* (hlavní nahrávka), která seskupuje více nahrávek, na kterých jsou nahrány stejné skladby, ale například byly vydány v jiné zemi apod.

Ať už se jedná o jakýkoliv z těchto druhů nahrávky, je u ní zkontrolován rok. Pokud u nahrávky není informace o roku dostupná, je přeskočena a pokračuje se kontrolou další nahrávky. V opačném případě se dále zjišťuje, zda rok vyhovuje nastaveným parametrům pro vyhledávání. Díky tomu, že jsou nahrávky seřazeny od nejmenšího roku, pokud není některá z podmínek splněna, hledání končí a pokračuje se prohledáváním nahrávek dalších nalezených interpretů. Jinak hledání pokračuje dále.

Dalším krokem je zjištění typu nahrávky. Pokud se jedná o hlavní nahrávku, zjistí se zda se v seznamu skladeb nachází hledaná skladba. Pokud ano, prochází se její jednotlivé verze a opakuje se proces s kontrolou roku. Následující kontroly jsou totožné, ať už se jedná o běžnou nahrávku, nebo verzi hlavní nahrávky. Je zapotřebí zjistit její formát. V případě Discogs nejsou vinylové desky dále děleny a jsou jednoduše označeny jako vinyl. Poté se projde seznam skladeb, kdy v případě nalezení shody je nahrávka uložena do seznamu výsledků.

Pro nalezené výsledky se poté stahují obrázky, které jsou ukládány mimo lokální databázi do souborového systému a do databáze je ukládána pouze cesta k souboru.

### MusicBrainzDownloader

Ve třídě MusicBrainzDownloader je implementováno hledání informací v databázi MusicBrainz. K tomu je použita knihovna musicbrainzngs, která zjednodušuje volání jednotlivých koncových bodů MusicBrainz API. Data jsou z API vracena běžně ve formátu XML, či JSON. Knihovna již vrací zpracovaná data v podobě slovníku.



Postup vyhledávání má jisté podobnosti s postupem použitým pro hledání v databázi Discogs. Ale na rozdíl od ní, MusicBrainz API prakticky nemá omezení na počet dotazů. A jak ukázaly testy 6.1, je hledání ve většině případů mnohem rychlejší než u databáze Discogs. Není tedy potřeba vyhledávání nijak závažně limitovat. Další podobností je, že hledání také funguje na principu stránkování a umožňuje specifikovat i počet výsledků, který má být vrácen. Tato hodnota může být 25 – 100. V některých případech, ale API vracelo menší počet výsledků, proto je nutné volit offset (posun) pro získání následující stránky, dle skutečného počtu vrácených výsledků.

Nejdříve se vyhledají interpreti, dle zadaného názvu. Nalezení interpreti jsou opět porovnávání dle podobnosti názvu. Hledání interpretů pokračuje, dokud nebyli zkontrolováni všichni, nebo nebylo prohledáno alespoň 1000 interpretů. Pro každého interpreta je poté prohledáván seznam nahrávek, dokud nejsou zkontrolovány všechny. U každé nahrávky se nejdříve zkontroluje seznam médií (formátů), na kterých se nahrávka nachází. Možná označení pro vinylovou desku jsou následující:

- Vinyl
- 7"Vinyl
- 10"Vinyl
- 12"Vinyl

Pokud odpovídá formát, je procházen seznam skladeb. V případě nalezení skladby se již dále formáty neprocházejí a nahrávka se ukládá do seznamu s výsledky. Tento seznam se následně seřadí dle roku a vyfiltrují se nahrávky, které by nevyhovovaly kritériím pro vyhledání roku. Nakonec jsou pro jednotlivé nahrávky staženy obrázky.

## WebScrapper

Třída WebScrapper.py slouží k nalezení a získání dat o nahrávkách z webových stránek 45cat a 45worlds. Struktura těchto stránek je ve většině případů stejná. I kdyby byla struktura naprosto totožná, mají rozdílné doménové jméno v URL adrese. Proto instance této třídy obsahují atribut typ. Ten může nabývat následujících hodnot:

- 7singles
- 12singles
- 78rpm
- albums

Hodnota 7singles označuje stránku 45cat. Ostatní hodnoty rozlišují jednotlivé kategorie pro 45worlds.

Pro získání obsahu stránek je použita knihovna Requests společně s knihovnou BeautifulSoup, která ulehčuje zpracování zdrojového HTML kódu. Po přejití na další odkaz se čeká v intervalu 2–5 sekund. Hodnota je z tohoto intervalu náhodně vygenerována. Děje se tak z důvodu, aby se chování programu alespoň částečně přiblížilo chování běžného uživatele a nebylo agresivní, kdy by se dotazy posílaly v rychlém sledu.

Pro nalezení nahrávek je použito vyhledávače, jenž stránky nabízí. Přesněji je poslán GET dotaz s parametry, jako by uživatel vyplnil vyhledávací pole a stiskl tlačítko vyhledat. Vyhledávat lze dle různých kritérií jako jsou název skladby, interpret, název nahrávky.

Ovšem každá z těchto možností nabízí rozdílné zobrazení dat, z nichž nejuvhodnější pro automatické zpracování je při vyhledání dle názvu skladby. Tím se vyhledávání liší od předchozích dvou zdrojů, kde byl nejdříve dohledán interpret a k němu procházeny jednotlivé nahrávky. Výsledky hledání jsou pak zobrazeny v tabulce. Struktura jednoho řádku tabulky je znázorněna v tabulce 5.1. Na té lze vidět, že jsou v ní obsaženy všechny potřebné informace pro rozhodnutí, zda má smysl tento záznam zkoumat podrobněji.

Název skladby	Interpret	Vydavatelství (label)	Typ*	Rok vydání
---------------	-----------	-----------------------	------	------------

Tabulka 5.1: Podoba jednoho řádku při vyhledávání na stránce 45cat nebo 45worlds dle názvu skladby.

\* – typ je uveden pouze na stránce 45cat

V každém řádku tabulky se zkontroluje, zda je uvedena informace o roku, jinak je řádek přeskočen. Následně je zkontrolován název skladby a interpreta. Pokud je u obou vyhodnocená shoda, uloží se o daném řádku odkaz na detail o nahrávce a rok. Po projití všech řádků tabulky, se zkusí vyhledat odkaz na další stránku s výsledky. V případě, že je stránek více opakuje se celý proces i pro další stránku.

Po prohledání všech výsledků hledání je seznam výsledků seřazen dle roku. Poté jsou znovu procházeny všechny nalezené záznamy a vyfiltrovány roky, které by nevyhovovaly nastaveným kritériím. Následně se již přechází přes získané odkazy na detaily o nahrávkách. Zde se z hlavní tabulky s informacemi zpracují relevantní informace. Pak se získá seznam skladeb z další tabulky a případně poznámky, ve kterých se můžou nacházet další podrobnější informace o nahrávce. Nakonec se dohledají obrázky, ze kterých je získán odkaz a stáhnou se do souborového systému.

## DownloaderByTrack

Všechny výše zmíněné třídy jsou instanciovány ve třídě **DownloaderByTrack.py**. Tato třída dědí od třídy **QRunnable**, která nabízí rozhraní pro části kódu, které mají být vykonávány opakovaně. Třídě jsou předány parametry s informací o interpretovi a skladbě, která má být vyhledána. Jsou vytvořeny instance tříd pro jednotlivé zdroje, a instance třídy **QThreadPool**, která pomáhá spravovat vlákna aplikace. Následně je vyhledávání pro každý ze zdrojů spuštěno v samostatném vlákně, přičemž zároveň mohou být spuštěna až tři vlákna. Jako první se spouští vyhledávání v databázi Discogs, MusicBrainz a 45cat. Jakmile je vyhledávání u jednoho z nich dokončeno, začne se vyhledávat v databázi 45worlds. Jakmile všechny vlákna dokončí svoji práci, jsou výsledky získané ze všech zdrojů postupně ukládány do databáze.

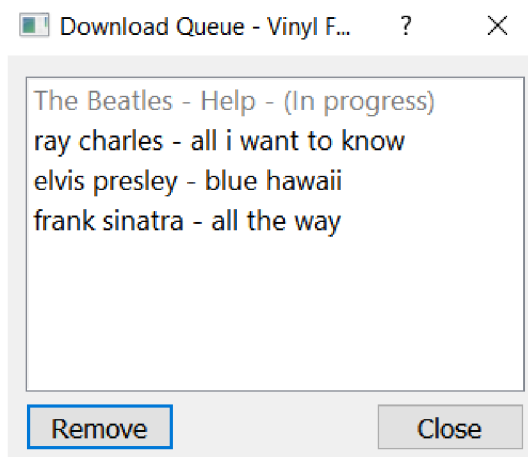
## 5.4 Spuštění vyhledávání

Pokud se uživatel rozhodne, že chce vyhledat informace o nahrávce, má dvě možnosti. Ikony pro obě z nich se nachází v hlavní liště aplikace. První možností je zadáním názvů ručně. Po kliknutí na tlačítko *Search Online* se otevře jednoduché dialogové okno, s textovými poli pro zadání názvu interpreta a skladby.

Druhá možnost je získáním názvu interpreta a skladby ze souboru. Po kliknutí na ikonu *Search From Files* se otevře dialogové okno. To je implementováno ve třídě **FileAndFolderDialog** a umožňuje vybrat jak soubor formátu MP3, nebo FLAC tak, složku. Což běžně není možné a bylo potřeba vynutit, aby i při vybrání složky bylo možné potvrdit výběr.

Vybraná položka je předána třídě **TagReader**. V případě vybrání složky, jsou procházeny všechny soubory a podsložky, které složka obsahuje a dohledány soubory formátu, který byl vybrán v dialogovém okně. U všech souborů je pak z metadat přečtena informace interpretovi a názvu skladby a vyslán signál, že má začít vyhledávání v internetových databázích.

Ve skutečnosti, pokud je přijat signál, že má začít vyhledávání, a již nějaké probíhá, hledání se nespustí. Vytvoří se instance třídy `DownloaderByTrack`, ale ta je uložena do fronty. Jakmile je dokončeno aktivní vyhledávání, teprve poté začne vyhledávání, které se nachází ve frontě. Velikost fronty není ničím omezená a její aktuální stav si může uživatel zobrazit kliknutím na ikonu `DownloadQueue`, nacházející se v hlavní liště aplikace. Z fronty je možné prvky i odebírat, tedy kromě vyhledávání, které už probíhá. Podobu fronty lze vidět na obrázku 5.3.



Obrázek 5.3: Dialogové okno zobrazující frontu vyhledávání

## Kapitola 6

# Testování a vyhodnocení výsledků

Tato kapitola se zaměřuje na způsoby testování výsledné aplikace. Porovnání jednotlivých zdrojů, ze kterých byly informace získávány. Ovlivnění výsledků vyhledávání různými nastaveními vyhledávacích parametrů. Zhodnocením dosažených výsledků a návrhy pro možné rozšíření a vylepšení.

### 6.1 Testování vyhledávání

Pro ověření použitelnosti aplikace VinylFinder bylo nutné provést řadu testů. Při jednotlivých testech byla zkoumána celá řada parametrů jako například počet dohledaných nahrávek, čas strávený vyhledáváním, kolik záznamů bylo celkově prohledáno, kolik interpretů bylo prohledáno, u kolika interpretů byl nalezen výsledek a další.

Aby bylo docíleno pro jednotlivé testy stejných podmínek, jsou pro ně stanovena pravidla, která byla v průběhu testování dodržována. Důvodem jejich zavedení bylo především, aby při vyhledávání se stejným názvem interpreta a skladby, ale jinými nastaveními aplikace, nebyla možnost, že budou výsledky ovlivněny již existujícími záznamy v lokální databázi. Dalším důvodem bylo ulehčení kontroly nalezených záznamů s ohledem na jejich relevantnost k hledanému výrazu. Před každým provedeným testem tedy byly vymazány všechny stažené obrázky a obsah databáze.

#### Optimalizace vyhledávání v databázi Discogs

Před prováděním konečných testů, které dále budou popsány, bylo provedeno několik optimalizací v rámci vyhledávání v databázi Discogs, pro dosažení kratšího času hledání. Při prvním testování, dosahoval čas vyhledávání mnohonásobně vyšších hodnot než ve finální verzi. Konkrétním příkladem je vyhledání skladby *Help!* od kapely *The Beatles*. V prvotní verzi byly výsledky nalezeny po více než hodině. Tento čas se podařilo snížit na 18 minut.

Jedním z důvodů bylo, že Discogs API může stejného interpreta vrátit ve výsledcích hned několikrát. Z čehož vyplývá, že nahrávky jednoho interprety mohly být procházeny dvakrát, nebo i vícekrát. Další optimalizací je omezení počtu interpretů, u kterých se porovnává jejich název. Discogs API většinou vrací větší počet výsledků. Při podrobné analýze se ukázalo, že relevantní výsledky, pro které je následně vyhodnocená shoda, se zpravidla nacházely hned na první, maximálně druhé stránce. Další stránky obsahovaly interprety, kde jejich název ani zdaleka neodpovídal hledanému výrazu. Proto je v konečné implementaci počet procházených stránek omezen na dvě.

Při výběru interpretů bylo cílem, aby konečný seznam byl různorodý především s ohledem na žánr a roky působení. Vzhledem k zaměření aplikace hlavně na dohledání originálních nahrávek, všichni z vybraných interpretů byli aktivní převážně v druhé polovině 20. století. Pokud není u testu řečeno jinak, byl prováděn s výchozím nastavením aplikace.

## Porovnání jednotlivých zdrojů

První testy jsou zaměřeny na úspěšnost hledání. Zkoumány jsou rozdíly mezi jednotlivými hudebními databázemi, které byly pro vyhledání použity. Výsledky jsou zprůměrovány z desítky hledání, které byly prováděny s různými parametry nastavení. Konkrétně tedy s výchozím nastavením, kdy byla hranice pro vyhodnocení shody nastavena na hodnotu 75 a byly hledány nahrávky co nejstarší plus jeden rok. Pak byly vyhledány stejné skladby a interpreti, ale s nastavenou přísnější, či mírnější hranicí pro porovnávání shody. V poslední řadě bylo provedeno hledání s větším rozsahem let.

Nejdříve byl zkoumán počet nalezených výsledků, které odpovídaly všem kritériím hledání a čas strávený hledáním. Výsledky tohoto testu jsou zobrazeny v tabulce 6.1. Z výsledků lze jednoznačně usoudit, že nejvíce záznamů bylo nalezeno v hudební databázi Discogs. Tento výsledek je očekávaný, jelikož v porovnání s ostatními zdroji obsahuje mnohonásobně více záznamů. Vyšší čas nutný pro vyhledání, byl také očekáván, především kvůli limitaci Discogs API, které umožňuje maximálně 60 dotazů za minutu. S přihlédnutím k tomuto omezení, potom počet nalezených shod za minutu, není tak špatným výsledkem. Jako nečekaný výsledek by se dala označit úspěšnost v případě databáze MusicBrainz. V porovnání s databázemi 45cat a 45worlds nabízí mnohem více záznamů. O to více, když je brán v potaz způsob implementace, kdy je prohledáváno více interpretů, než by pravděpodobně bylo nutné. Nejlépe v poměru nalezených výsledků k době hledání dopadla databáze 45cat. Což oproti původnímu očekávání ukazuje, že se jedná o zdroj, který není zanedbatelný. U databáze 45worlds jsou značné rozdíly v jejich jednotlivých sekcích. V sekci pro dvanácti palcové desky, byla ve všech hledáních nalezena pouze jedna nahrávka, což z celkového počtu vytvářelo velice nízké číslo, proto je v tabulce uvedena hodnota 0. Výsledky tohoto testování, mohou být zkráceny výběrem interpretů a skladeb. Je dost možné, že při jiném výběru by byly výsledky lehce odlišné.

Zdroj	Počet shod	Čas (minuty)	Shody za minutu
Discogs	56,9	10:49	5,3
MusicBrainz	6,2	4:11	1,5
45cat	7,9	0:52	9,2
45worlds – alba	10,6	4:31	2,3
45worlds – 12"	0	0:13	0
45worlds – 78 RPM	0,7	0:15	2,8

Tabulka 6.1: Výsledky testování v závislosti na zdroji. Hodnoty v tabulce jsou průměrem ze všech prováděných vyhledávání.

## Porovnání výsledků při změnách nastavení

Pro vhodné zvolení výchozích hodnot nastavení, bylo nutné provést řadu testů s různě nastavenými hodnotami a posléze vyhodnotit, ve kterých případech je dosaženo nejlepších

výsledků. Hlavním parametrem, který podstatně ovlivňuje jak rychlost, tak počet nalezených shod, je hranice pro vyhodnocení shody hledaného výrazu se zadaným.

Testování probíhalo na menším vzorku dat (10 různých skladeb a interpretů), kdy nejprve byla hodnota pro vyhodnocení shody nastavena na hodnotu 50 a při každé další iteraci testů zvětšena o 5. U jednotlivých testů potom bylo porovnáváno, zda jsou data relevantní k zadanému výrazu, doba trvání vyhledávání, počet nalezených výsledků.

Jak lze očekávat, při nižších hodnotách bylo nalezeno o něco více výsledků. Ovšem velká většina z nich, nesouvisela s hledaným výrazem. Mimo se například u databáze Discogs mnohonásobně zvedla doba nutná pro dokončení vyhledání, především z důvodu, že byly prohledávány nahrávky více interpretů, u kterých nebyla nalezena žádná shoda v názvu skladby.

Při zvyšování hodnoty, začínaly být výsledky přesnější. Ovšem při hodnotách 85 a více, se při vyhledávání objevilo několik případů, kdy v internetové databázi bylo více interpretů se stejným názvem a pro jejich odlišení byla použita číslovka. V takovém případě u interpreta, jehož jméno obsahovalo číslovku, nebyla vyhodnocena shoda. Pokud právě interpret s číslovkou ve jméně, měl být ten, který obsahoval hledané nahrávky, celé vyhledávání bylo neúspěšné a nebyli nalezeny žádné vyhovující nahrávky.

Tímto způsobem se došlo k závěru, že právě hodnota 75 je zlatým středem, kdy jsou nacházeny relevantní výsledky a zároveň je stále prostor pro vyhodnocení shody, pokud je název v databázi odlišen nějakým identifikátorem.

## 6.2 Návrh možných rozšíření

V průběhu implementace a testování byla odhalena řada věcí, které by aplikaci mohli v mnoha směrech vylepšit, nebo přinejmenším usnadnit uživateli práci s aplikací.

Jednou z takových věcí je ještě více optimalizovat vyhledávání. A to především u databáze Discogs. K implementaci vyhledávání byla použita pomocná knihovna. Pomocí ní sice bylo možné dosáhnout poměrně uspokojivých výsledků. I přes to se domnívám, že pro konkrétní potřeby aplikace VinylFinder, by při vlastní implementaci získávání dat z Discogs API bylo možné omezit počet zasílaných dotazů a tím dosáhnout vyšší rychlosti.

Dalším možným vylepšením by mohlo být vyhledávání nahrávek podle dalších vlastností, než pouze název interpreta a skladby. Zde by bylo zapotřebí vyřešit, několik dalších problémů, především při zjišťování shody s hledaným výrazem. Vinylové desky, konkrétně tedy singly obsahují většinou dvě skladby, kdy název těchto skladeb je pak uveden v názvu nahrávky. U některých databázích je pak v názvu nahrávky uvedena například i země, kde deska byla vydána apod. Bylo by zapotřebí blíže prozkoumat v jakém tvaru, se může název nahrávky v jednotlivých databázích objevit a podle toho dále zpracovat.

Následující funkce by byla určena především pro uživatele, kteří se rozhodli digitalizovat si svoji vinylovou sbírku. Aplikace VinylFinder by pak nejdříve dohledala informace o nahrávkách. Poté by si uživatel vybral dohledaný záznam a soubor s obsahující digitalizovanou nahrávku a informace o nahrávce by se vložily do metadat hudebního souboru.

V poslední řadě by byla vhodná možnost porovnání jednotlivých dohledaných záznamů o nahrávkách. Stejně jako je možné v některých e-shopech porovnávat parametry zboží bok po boku, by uživatel měl možnost porovnat informace o jednotlivých nahrávkách.

# Kapitola 7

## Závěr

Cílem této práce bylo vytvořit desktopovou aplikaci, která v hudebních databázích vyhledává informace o původních nahrávkách vinylových desek. Ty jsou dohledávány na základě názvu interpreta a skladby, které jsou zadány uživatelem aplikace. Na základě těchto informací by se poté uživatel měl snáze rozhodnout o originalitě nahrávky.

Na začátku práce bylo potřeba prozkoumat internetové databáze, které poskytují metadata k hudebním souborům a zjistit jakým způsobem z nich mohou být data získány. Dalším krokem bylo zjistit, jaká metadata jsou relevantní pro nahrávky na vinylových deskách. Především tedy z hlediska určení originality nahrávky.

Na základě získaných poznatků byl pak proveden návrh aplikace. Pro dohledávání informací bylo využito databází Discogs, MusicBrainz, 45cat a 45worlds. První dvě zmíněné databáze zpřístupňují svá data skrze API, které umožňují vyhledávání dle různých parametrů a lze pomocí nich vyhledat interpreta a poté procházet jeho nahrávky. Naopak 45cat a 45worlds žádné API nemají. Pro získávání dat tedy bylo využito techniky zvané web scraping. Což ve zkratce znamená zpracovávání zdrojového kódu webových stránek a vybírání z něj užitečných dat. Informace o nahrávkách u kterých byla vyhodnocena shoda s hledaným výrazem jsou následně ukládány do lokální databáze.

Posléze byla aplikace podrobena rozsáhlému testování, kdy byla zkoumána především úspěšnost a rychlost vyhledávání, nebo také relevantnost vůči zadaným výrazům, pro hledání. Díky testům byl objeven prostor pro optimalizaci vyhledávání v databázi Discogs. V jednom z případů pomohla tato optimalizace zkrátit čas hledání téměř o hodinu. Ve většině ostatních případech, byla změna řádově v jednotkách minut. Ačkoliv i tak tuto skutečnost hodnotím jako úspěch.

I přes tento úspěch se ovšem domnívám, že by bylo možné vyhledávání ještě více optimalizovat. Nejrozumnějším řešením by bylo místo pomocné knihovny pro komunikaci s Discogs API, implementovat vlastní způsob získávání dat. Vzhledem ke způsobu, jakým jsou data v aplikaci zpracovávána, si myslím, že by bylo možné snížit počet dotazů na službu. Rychlost vyhledávání je nyní brzděna právě kvůli limitaci počtu zpracovaných dotazů na 60 za minutu.

# Literatura

- [1] BAYER, M. SQLAlchemy. In: BROWN, A. a WILSON, G., ed. *The Architecture of Open Source Applications Volume II: Structure, Scale, and a Few More Fearless Hacks* [online]. Aosabook.org, 2012 [cit. 2021-01-10]. Dostupné z: <http://aosabook.org/en/sqlalchemy.html>.
- [2] BLACK, P. E. *Algorithms and Theory of Computation Handbook* [online]. 15. vyd. duben 2019 [cit. 2021-01-15]. Dostupné z: <https://www.nist.gov/dads/HTML/Levenshtein.html>.
- [3] BOUVIGNE, G. Frame header. *MP3' Tech* [online]. 2001 [cit. 7. dubna 2020]. Dostupné z: [http://www.mp3-tech.org/programmer/frame\\_header.html](http://www.mp3-tech.org/programmer/frame_header.html).
- [4] DISCOGS. *How To Find Information On A Vinyl Record* [online], 30. srpna 2018 [cit. 25. dubna 2020]. Dostupné z: <https://support.discogs.com/hc/en-us/articles/360008602254-How-To-Find-Information-On-A-Vinyl-Record>.
- [5] FOUNDATION, M. *How To Find Information On A Vinyl Record* [online]. 2020 [cit. 29. dubna 2020]. Dostupné z: [https://musicbrainz.org/doc/Introduction\\_to\\_Voting](https://musicbrainz.org/doc/Introduction_to_Voting).
- [6] HACKER, S. *MP3: The Definitive Guide*. 1. vyd. O'Reilly, březen 2000 [cit. 2020-04-05]. ISBN 1-56592-661-7.
- [7] ID3.ORG. mp3Frame. *Id3.org* [online]. 2012. Aktualizováno 8. 10. 2012 [cit. 7. dubna 2020]. Dostupné z: <https://id3.org/mp3Frame>.
- [8] JAROSLAV REICHL, M. V. Kvantování signálu. *Digital Sound & Music* [online]. 2011. Aktualizováno 27. 12. 2011 [cit. 22. března 2020]. Dostupné z: <http://fyzika.jreichl.com/main.article/view/1358-kodovani-signalu>.
- [9] JOSH COALSON, X. F. documentation. *FLAC* [online]. 2014 [cit. 16. dubna 2020]. Dostupné z: <https://xiph.org/flac/documentation.html>.
- [10] MARTIN NILSSON, e. a. *Id3.org* [online]. 2012 [cit. 15. dubna 2020]. Dostupné z: <https://id3.org/ID3v2Easy>.
- [11] ROMNEY, J. 5.1.2 Digitization. *Digital Sound & Music* [online]. 2014. Aktualizováno 23. 1. 2018 [cit. 22. března 2020]. Dostupné z: <http://digitalsoundandmusic.com/5-1-2-digitization>.