

ŠIAULIŲ UNIVERSITETAS

Kompiuterių sistemų katedra

Robertas Pocius

**PSEUDO ATSITIKTINIS GROJARAŠČIŲ  
SUDARYMO ALGORITMAS IR JO TYRIMAS**

Magistro baigiamasis darbas

Vadovė dr. S. Ramanauskaitė

Šiauliai, 2015

ŠIAULIŲ UNIVERSITETAS

Kompiuterių sistemų katedra

TVIRTINU

Kompiuterių sistemų katedros vedėjas dr. E. Paliulis  
2015-06-01

## **PSEUDO ATSITIKTINIS GROJARAŠČIŲ SUDARYMO ALGORITMAS IR JO TYRIMAS**

Informatikos inžinerijos magistro baigiamasis darbas

### **Vadovė**

KS katedros docentė  
2015 m. birželio 1 d.

dr. S. Ramanauskaitė

### **Recenzantai**

KS katedros docentas  
2015 m. birželio \_\_ d.

dr. E. Paliulis

KS katedros lektorius  
2015 m. birželio \_\_ d.

dr. R. Zemblys

### **Atliko**

ITM-13 gr. studentas  
2015 m. birželio 1 d.

R. Pocius

Šiauliai, 2015

## **SANTRAUKA**

### **Pseudo atsitiktinis grojaraščių sudarymo algoritmas ir jo tyrimas**

Dabartiniu metu sparčiai didėjant informacinių technologijų skvarbai pasaulyje, atsiveria naujos informacijos valdymo galimybės. Greitas ypatingai spartaus ryšio, interneto bei išmaniųjų įrenginių populiarėjimas lemia, kad informacija tampa pasiekiamą bet kur ir bet kada, o kurti, keisti, kaupti ar naudoti jos turinį gali kiekvienas interneto vartotojas. Taigi, šiuo metu vis labiau populiarėjant internetui, internetinėms muzikos parduotuvėms, pažengę ir skaitmeninio garso suspaudimo technologijos, kurios muzikos klausytojams leidžia kaupti didžiules muzikos bibliotekas. Atsirado automatiniai grojaraščių sudarymo algoritmai bei jais paremtos muzikos rekomendavimo sistemos, kurios padeda vartotojams klausytis tik jiems patinkančios muzikos ir taupo jų laiką, nes nebereikia kurti grojaraščių patiems vartotojams.

Šiame darbe pateikiamas suprojektuotas pseudo atsitiktinis grojaraščių sudarymo algoritmas, atsižvelgiantis į muzikos klausymo istoriją. Šis algoritmas yra orientuotas į kuo mažesnę papildomą vartotojo veiksmų įsikišimą.

Pagal pasiūlytą algoritmą realizuotas prototipas. Jo pagalba įvertinta, kad siūlomas algoritmas grojaraščius generuoja adaptyviau ir labiau prisitaiko prie vartotojo muzikos klausymo poreikių, nei kiti egzistuojantys sprendimai.

## **SUMMARY**

### **The Design and Research of Music Shuffle Algorithm**

Nowadays new information management is revealing due to the rapid growth of information technology in the world. Quick especially high-speed Internet and the popularity of smart devices help to improve the accessibility of information everywhere and anytime. Creation, editing, storing or using its content become capable for each internet user. As a result of growing popularity of internet, online music stores, and progress in digital audio compression technology allowed music listeners to create massive music libraries. Appeared automatic playlists algorithms and music recommender systems helps users to listen to the music they want only and saves them time, because there is no need for consumers to create their own playlists.

In the thesis music shuffle algorithm based on user's music play history is designed. The algorithm is oriented to minimize the additional user action intervention in order to identify suitable music for the listener at the moment.

Under the proposed algorithm a prototype is realized. Comparison of the prototype and other existing music players with shuffle function showed the proposed algorithm is more accurate and adaptive to user's music listening needs than other existing solutions.

## TERMINŲ IR SANTRUMPŲ ŽODYNĖLIS

- AIMP** „AIMP“ muzikos grotuvas
- BPM** Dūžiai per minutę (*angl.* beats per minute) yra vienetas, paprastai naudojamas kaip matas muzikos ritme.
- Java** Objektiškai orientuota programavimo kalba.
- Meta-duomenys** Struktūriškai apibrėžta informacija, kuri apibūdina, paaiškina dokumentą ar informacijos šteklių, nurodo jo buvimo vietą arba kitokiu būdu palengvina jo suradimą, naudojimą arba valdymą.
- MP3** Srautinis formatas, skirtas skaitmeninėje formoje saugoti suspaustą audio signalą bei jį perduoti.
- Multimedija** Daugialypė terpė, dažniausiai reiškia informacijos pateikimą skaitmeniniame formate.
- PRO** Realizuotas pasiūlyto algoritmo prototipas.
- Pseudo atsitiktinis** Ne tikrai atsitiktinis, bet toks kurį tam tikromis sąlygomis galima laikyti atsitiktiniu.
- RAM** Operatyvinė atmintis.
- Shuffle** Muzikos klausymo režimas, kai grojaraštyje esančios dainos yra grojamos ne iš eilės, o pseudo atsitiktine tvarka.
- WMP** „Windows Media Player“ muzikos grotuvas

## PAVEIKSLĖLIŲ SĄRAŠAS

<b>1 pav.</b>	„Spotify“ muzikos rekomendavimo sistema .....	10
<b>2 pav.</b>	„iTunes Genius“ muzikos rekomendavimo sistema.....	11
<b>3 pav.</b>	„Pandora“ muzikos rekomendavimo sistema.....	12
<b>4 pav.</b>	„Last.fm“ muzikos rekomendavimo sistema .....	13
<b>5 pav.</b>	Siūlomo algoritmo bazinė schema .....	17
<b>6 pav.</b>	Savybių svarbos koeficientų sąrašo generavimo diagrama .....	19
<b>7 pav.</b>	Savybių svarbos koeficientų sąrašo kitimas, kai klausoma daina nepatiko .....	20
<b>8 pav.</b>	Savybių svarbos koeficientų sąrašo kitimas, kai klausoma daina patiko .....	21
<b>9 pav.</b>	Dainų koeficientų sąrašo generavimo diagrama .....	22
<b>10 pav.</b>	Dainų koeficientų sąrašo kitimas .....	23
<b>11 pav.</b>	Sekančios grojaraščio dainos parinkimo diagrama .....	24
<b>12 pav.</b>	Algoritmo realizacija.....	25
<b>13 pav.</b>	Tyrime naudojamų muzikos grotuvų aplinkos.....	26
<b>14 pav.</b>	Metų pasiskirstymas .....	27
<b>15 pav.</b>	Ritmų pasiskirstymas .....	27
<b>16 pav.</b>	Žanrų pasiskirstymas.....	28
<b>17 pav.</b>	Dainų klausymo sesijos su visais muzikos grotuvais .....	29
<b>18 pav.</b>	Vidutinis visų respondentų pastovus dainų išklausymas be perjunginėjimų .....	30
<b>19 pav.</b>	Dainų išklausymas mažiau nei 50% dainos .....	30
<b>20 pav.</b>	Pilnas dainų išklausymas.....	31
<b>21 pav.</b>	Vidutinis dainų išklausymas pagal muzikos grotuvą .....	31
<b>22 pav.</b>	Žanrų kitimas .....	32
<b>23 pav.</b>	Ritmų kitimas .....	32
<b>24 pav.</b>	Dainų išklausymo histograma .....	33
<b>25 pav.</b>	Savybių svarbos koeficientų sąrašo generavimo našumas .....	34
<b>26 pav.</b>	Muzikos įrašų koeficientų sąrašo generavimo našumas.....	35
<b>27 pav.</b>	Sekančios dainos parinkimo našumas .....	36
<b>28 pav.</b>	RAM naudojimas parenkant sekančią grojaraščio dainą .....	36

## TURINYS

ĮVADAS.....	8
1. ATSITIKTINIŲ GROJARAŠČIŲ SUDARYMO ALGORITMAI.....	9
1.1 Muzikos grojimo režimas „Shuffle“.....	9
1.2 Muzikos rekomendavimo sistemos.....	9
1.3 Muzikos rekomendavimo metodai.....	11
1.3.1 Automatinis grojaraščio sudarymo metodas.....	11
1.3.2 Dinaminis grojaraščio sudarymo metodas.....	12
1.3.3 Bendro filtravimo grojaraščio sudarymo metodas.....	13
1.3.4 Dainos turiniu pagrįstas grojaraščio sudarymo metodas.....	13
1.3.5 Dainos meta-duomenimis paremtas grojaraščio sudarymo metodas.....	14
1.3.6 Kontekstu paremtas grojaraščio sudarymo metodas.....	14
1.3.7 Hibridinis grojaraščio sudarymo metodas.....	14
2. PSEUDO ATSITIKTINIS GROJARAŠČIŲ SUDARYMO ALGORITMAS, ATSIŽVELGIANTIS Į MUZIKOS KLAUSYMO ISTORIJĄ.....	16
2.1 Siūlomo algoritmo principai.....	16
2.1.1 Dainos išklausymas.....	17
2.1.2 Savybių svarbos koeficientų sąrašas.....	18
2.1.3 Dainų koeficientų sąrašas.....	21
2.1.4 Sekančios dainos parinkimas.....	23
2.2 Siūlomo algoritmo realizacija.....	25
3. PASIŪLYTO ALGORITMO TYRIMAS.....	26
3.1 Tyrimo aplinka, duomenys, eiga ir sąlygos.....	26
3.1.1 Tyrimo duomenys.....	27
3.1.2 Tyrimo eiga.....	28
3.2 Tyrimo rezultatai.....	29
3.3 Pasiūlyto algoritmo našumo testavimas.....	34
3.3.1 Testavimo resursai.....	34
3.3.2 Testavimo rezultatai.....	34
LITERATŪRA.....	39

## IVADAS

Pastaraisiais metais populiarėjant internetui, internetinėms muzikos parduotuvėms, stipriai pažengė ir skaitmeninio garso suspaudimo technologijos, kurios muzikos klausytojams leidžia visą savo muzikos biblioteką nešiotis tiesiog kišenėje. Klausytojų muzikos bibliotekos išaugo nuo kelių dešimčių iki kelių tūkstančių muzikos įrašų, ko pasekoje susiduriama su problema, kad klausytojui sunku išsirinkti, ką klausytis iš tokios gausos muzikos įrašų, o kai kurios dainos gali būti klausomos tik kartą arba iš viso nei karto. Minėtai problemai spręsti buvo sukurti įvairūs dainų parinkimo, grojaraščių sudarymo algoritmai bei muzikos rekomendavimo sistemos, kurių pagalba siekiama kuo labiau patenkinti klausytojo muzikos klausymo poreikius.

Šio darbo **tyrimo objektas** – pseudo atsitiktinių grojaraščių sudarymo algoritmai, jų principai ir pritaikymo galimybės.

Šio darbo **tikslas** – suprojektuoti pseudo atsitiktinio grojaraščio sudarymo algoritmą, kuris sekančią dainą vartotojui parinktų, atsižvelgdamas į prieš tai klausytų dainų sąrašą ir būtų orientuotas į kuo mažesnę papildomą vartotojo veiksmų įsikišimą.

Darbe keliami šie uždaviniai:

1. Išanalizuoti egzistuojančius atsitiktinių grojaraščių sudarymo sprendimus.
2. Suprojektuoti naują pseudo atsitiktinio grojaraščio sudarymo algoritmą, kuris atrinkdamas sekančią grojamą dainą, remtųsi prieš tai vartotojui grotų muzikos kūrinų sąrašu.
3. Realizuoti suprojektuoto algoritmo prototipą.
4. Ištirti pasiūlyto algoritmo gebėjimą adaptuotis prie vartotojo muzikinio skonio.

Šis darbas svarbus praktine prasme, nes naujai pasiūlytas algoritmas adaptuosis prie vartotojo muzikinio skonio ir adaptyviai generuos muzikos grojaraštį, kuris turėtų labiau patenkinti klausytojo muzikos klausymo poreikius, todėl turėtų sumažėti vartotojo įsikišimas perjunginėjant nepatikusias grojaraščio dainas.



# 1. ATSITIKTINIŲ GROJARAŠČIŲ SUDARYMO ALGORITMAI

## 1.1 Muzikos grojimo režimas „Shuffle“

Pastaraisiais metais internetas tapo pagrindiniu šaltiniu, ieškant multimedijos, pavyzdžiui: vaizdo įrašų, knygų, muzikos ir t.t. Daugumai žmonių muzika yra neatsiejamas gyvenimo dalykas, todėl vis daugiau žmonių klausosi muzikos. Ankstesni tyrimai parodė, kad tyrimo dalyviai klausėsi muzikos daug dažniau nei užsiiminėjo kita veikla [19] (t.y. žiūrėjo televizorių, skaitė knygas ir žiūrėjo filmus). Stipriai pažengė ir skaitmeninio garso suspaudimo technologijos, kurios muzikos klausytojams leidžia gana lengvai kaupti didelį kiekį įvairiausių muzikos įrašų. Muzikos paieška ir saugojimas tapo toks paprastas, kad įsigyti naują muzikos albumą ar dainą užtenka vos kelių pelės paspaudimų. Tačiau dabar atsirado problema, kaip vartotojams organizuoti ir valdyti didžiules muzikos įrašų kolekcijas. Turint nemažą muzikos įrašų kolekciją, kartais sunku surasti ką norėtūsi paklausti. Tokiu atveju galima susikurti savo grojaraštį, bet tai gali užimti nemažai laiko, todėl dauguma klausytojų, negaišdami brangaus laiko, klausosi muzikos maišymo (*angl.* shuffle) režimu.

Muzikos maišymo (*angl.* shuffle) režimas – tai muzikos grojimo režimas, kai klausytojo grojaraštyje esančios dainos grojamos ne paeiliui, bet sumaišomos pagal atitinkamą muzikos grojaraščio sudarymo algoritmą ir dainos grojamos ta tvarka, kurią sugeneravo algoritmas [15]. Kitaip tariant, šį muzikos klausymo režimą galima traktuoti, kaip grojaraščio sudarymo algoritmą, kuris nustato, kokia tvarka bus grojamos dainos.

Grojaraščių sudarymo algoritmai dažniausiai sutinkami įvairiausiose programose, kaip maišymo (*angl.* shuffle) funkcija, kurios skirtos muzikos įrašų klausymui. Tokią funkciją turi: „iTunes“, „Winamp“, „AIMP“, „VLC“, „Windows Media Player“, mobilieji įrenginiai bei MP3 grotuvai. Šią funkciją palaiko praktiškai kiekvienas šiuolaikinis muziką galintis atkurti įrenginys. 2005 metais kompanija „Apple“ sėkmingai pristatė MP3 muzikos grotuvą „iPod Shuffle“, kuris skirtas klausyti muzikos tik maišymo (*angl.* shuffle) režimu. Šio MP3 grotuvo pardavimai buvo tokie geri, kad per pirmus metus jam priklausė 43 % visos MP3 grotuvų pasaulio rinkos [1].

Taip pat atsirado internetinės muzikos rekomendavimo sistemos, kurių veikimas paremtas, būtent, grojaraščių sudarymo algoritmais. Populiariausios internetinės muzikos rekomendavimo sistemos yra: „Last.fm“, „Pandora“, „Allmusic“, „Musicoverly“, „Spotify“, „iTunes Genius“, „Mufin“. Jos pritraukė milijonus muzikos klausytojų ir toliau pasižymi intensyviu vystymusi ir plėtra muzikos kūrinių pasirinkimo ir klausymosi režimų srityje [5, 6].

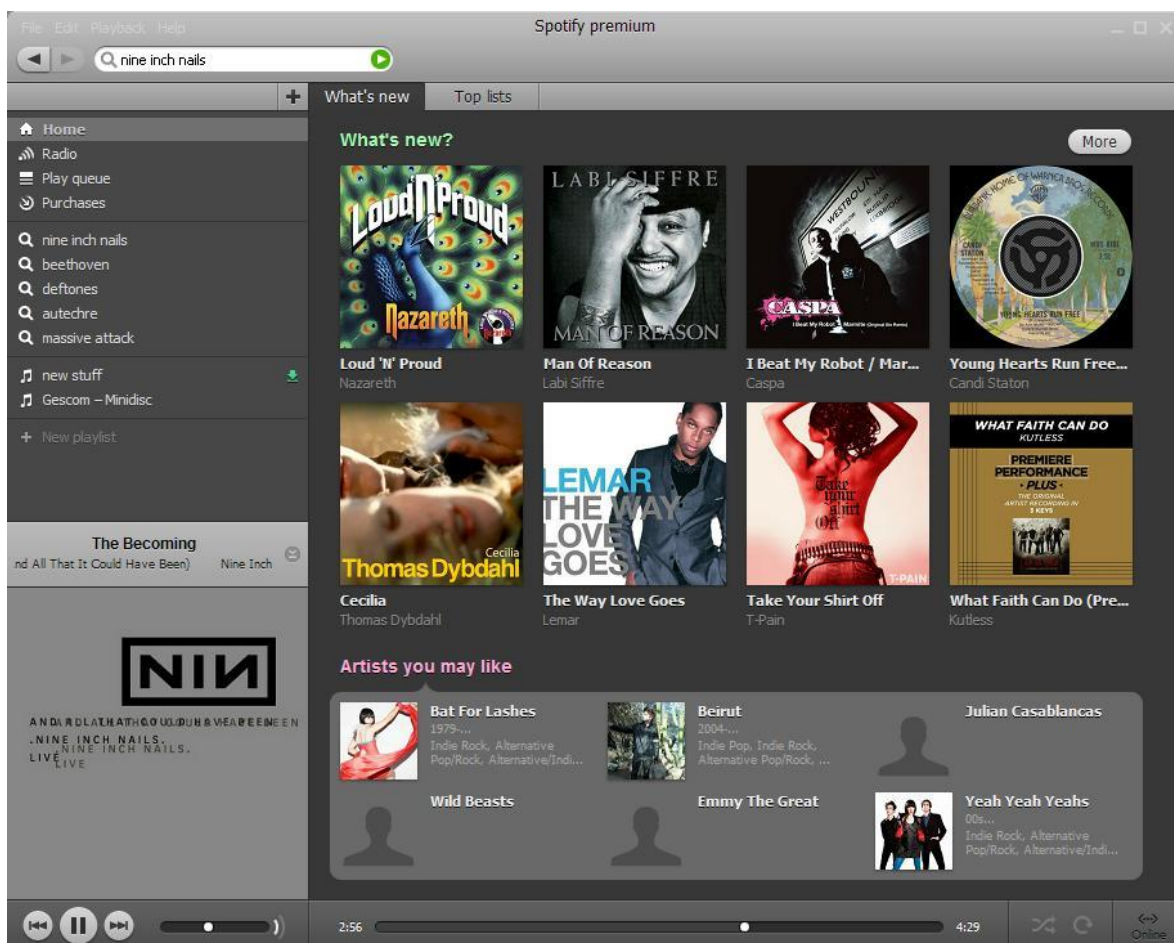
## 1.2 Muzikos rekomendavimo sistemos

Muzikos rekomendavimo sistema – tai sistema, kuri turi galėti automatiškai rekomenduoti muzikos įrašus (sudaryti muzikos grojaraštį), atsižvelgiant į kiekvieno klausytojo individualius muzikos klausymo poreikius bei įpročius [14]. Tokių muzikos rekomendavimo sistemų šiai dienai egzistuoja nemažai. Populiariausios muzikos rekomendavimo sistemos:

- „Last.fm“ – tai nemokama muzikos rekomendavimo sistema, įkurta Jungtinėje Karalystėje 2002 metais. Ši sistema turi daugiau nei 30 milijonų aktyvių vartotojų.

„Last.fm<sup>1</sup>“ sistema bene geriausiai įgyvendino bendro filtravimo muzikos rekomendavimo metodą.

- „Pandora“ – muzikos transliavimo ir automatinė muzikos rekomendavimo sistema, paremta „Music Genome<sup>2</sup>“ projektu, kuriame apmokyti „Pandora“ darbuotojai klasifikuoja kiekvieną muzikos įrašą pagal kelis šimtus kintamųjų. „Pandora“ yra prieinama tik Jungtinėse Amerikos Valstijose, Australijoje ir Naujojoje Zelandijoje. Ši sistema turi daugiau nei 80 milijonų aktyvių vartotojų [18].
- „Spotify“ – tai komercinė muzikos transliavimo ir rekomendavimo sistema (žr. 1. 1. 11 lentelė. 1 pav. 1 pav. ), įkurta 2008 metais Švedijoje. Taip pat yra ir nemokama sistemos versija. Ši sistema turi daugiau nei 60 milijonų vartotojų, iš kurių 15 milijonų naudojami komercine sistemos versija [29]. „Spotify“ muzikos rekomendavimo sistema pagrįsta muzikos įrašų atlikėjų duomenimis. Pirmiausia yra atsižvelgiama, kokių atlikėjų klausosi vartotojas, o tada, naudojant duomenis apie panašius atlikėjus iš „All Music Guide<sup>3</sup>“ duomenų bazės, yra rekomenduojamos sekančios dainos vartotojui [2].



1 pav. „Spotify“ muzikos rekomendavimo sistema

<sup>1</sup> <http://www.last.fm/>

<sup>2</sup> <https://www.pandora.com/about/mgp>

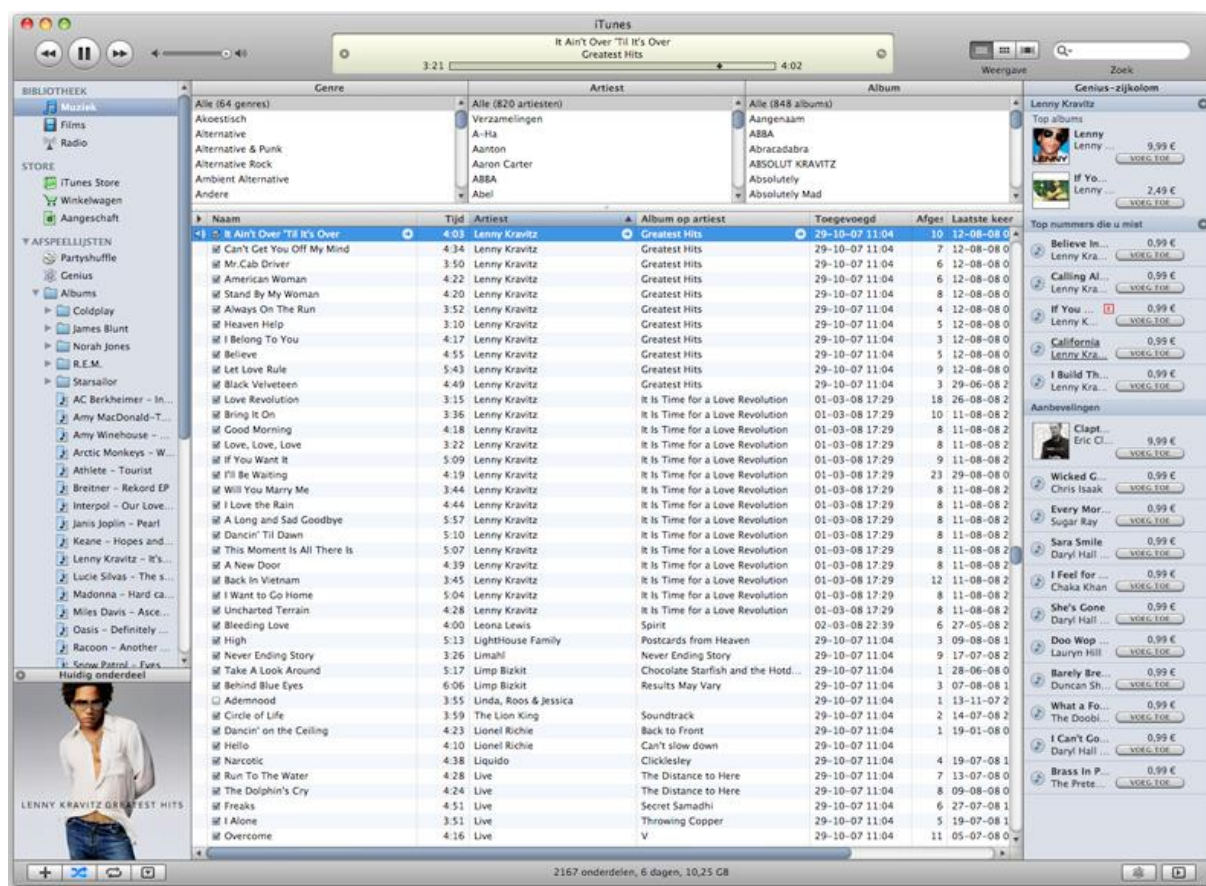
<sup>3</sup> <http://www.allmusic.com/>

### 1.3 Muzikos rekomendavimo metodai

Dauguma kompanijų savo produktuose, tokiuose kaip, pvz. „Apple iTunes“, „Microsoft Windows Media Player“, naudoja grojaraščių sudarymo algoritmus, bet tie produktai yra komerciniai ir juose taikomi algoritmai yra neatskleidžiami. Kadangi nėra galimybės tiksliai sužinoti, koku principu jie veikia, negalima sužinoti į ką šiuose algoritmuose yra atsižvelgiama (pvz. dainos metaduomenis, klausytojo veiksmus) generuojant grojaraštį. Pavyzdžiui „VLC“ muzikos grotuve [21] tiesiog naudojama atsitiktinių skaičių (*angl.* random) generavimo funkcija, o į turimos dainų bibliotekos metaduomenis visiškai nekreipiama dėmesio. Tačiau šiandienai yra sukurta įvairių muzikos rekomendavimo metodų, kurie sugeba parinkti labiau vartotojams turinčius patikti muzikos kūrinius. Šiuos metodus galima suskirstyti į kelias grupes.

#### 1.3.1 Automatinis grojaraščio sudarymo metodas

Automatinio grojaraščio sudarymo metode svarbiausias dėmesys yra sutelktas į tai, kokią dainą dabar klausosi vartotojas. Pagal tai yra nustatoma, kokias dainas įtraukti į grojaraštį. Ragno [20] pateikė būdą – grojaraštį sudaryti tik iš tokių dainų, kurios yra panašios į dabar klausomą dainą. Tarkim, jeigu dabar klausoma daina yra POP žanro, tai pagal Ragno pasiūlymą, grojaraštis bus sudarytas iš tokio pat žanro dainų. Taip pat Flexer [8] pateikė būdą, kai vartotojas parenka pirmą ir paskutinę grojaraščio dainą. Pagal tai grojaraščio pradžia yra sudaroma iš dainų, panašių į vartotojo pasirinktą pirmąją grojaraščio dainą, o grojaraščio pabaiga sudaroma iš dainų, panašių į vartotojo pasirinktą paskutinę grojaraščio dainą.



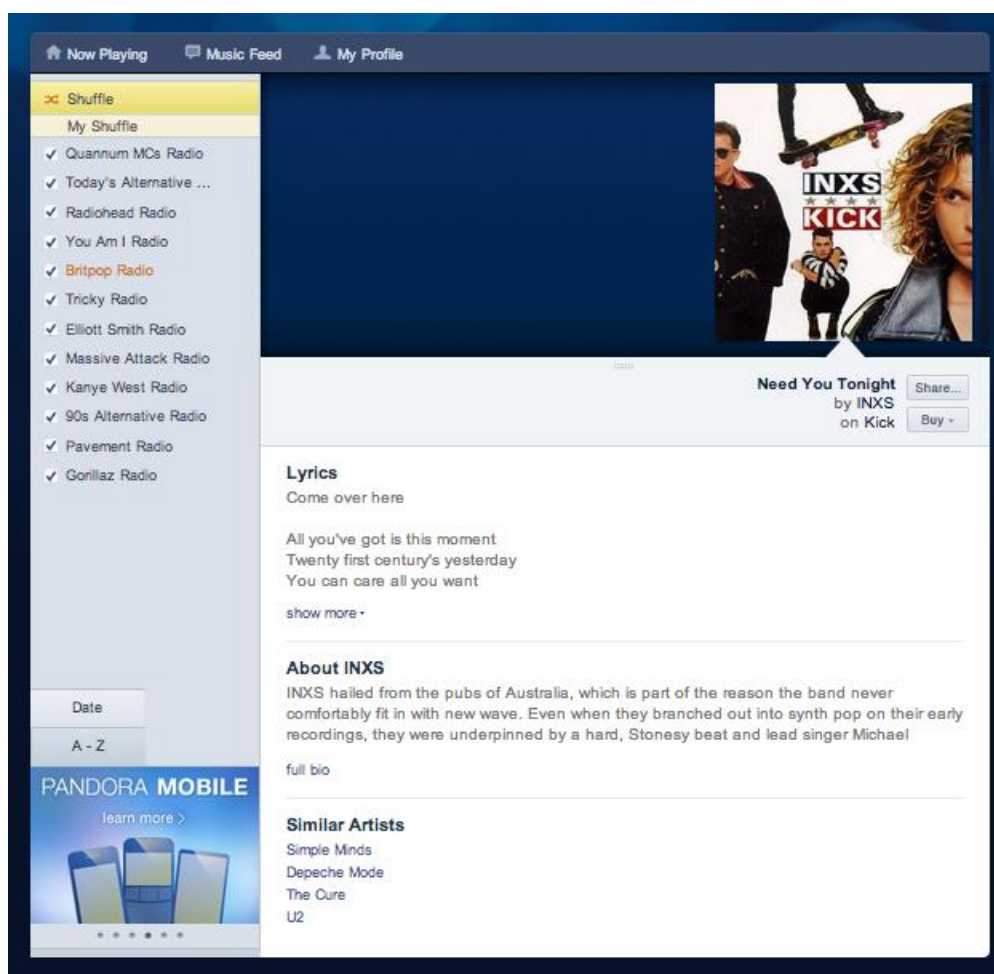
2 pav. „iTunes Genius“ muzikos rekomendavimo sistema

Tačiau pagrindinė šitų metodų problema yra ta, kad pasirenkant vieną ar kelias dainas grojaraščio generavimui, grojaraštis tampa gan monotoniškas, nes visos dainos grojaraštyje parenkamos labai panašios į pasirinktąsias. Toks grojaraštis gali greitai atsibosti klausytojui dėl įvairovės nebuvimo. Taip pat šie metodai ignoruoja klausytojo pastabas, tokias kaip dainų prasukinėjimas, dainų reitingavimas ir remiasi tik pradiniu vartotojų pasirinkimu.

„iTunes Genius“ muzikos rekomendavimo sistema dirba panašiais metodais – generuoja grojaraštį pagal dabar klausomą dainą (žr. 2 pav. ). Barrington darbe užsiminta, kad „iTunes Genius“ taip pat naudoja ir dainų meta-duomenis generuojant grojaraštį [13]. Meta-duomenys šio tipo algoritmuose būtini, norint įvertinti dainų panašumus.

### 1. 3. 2 Dinaminis grojaraščio sudarymo metodas

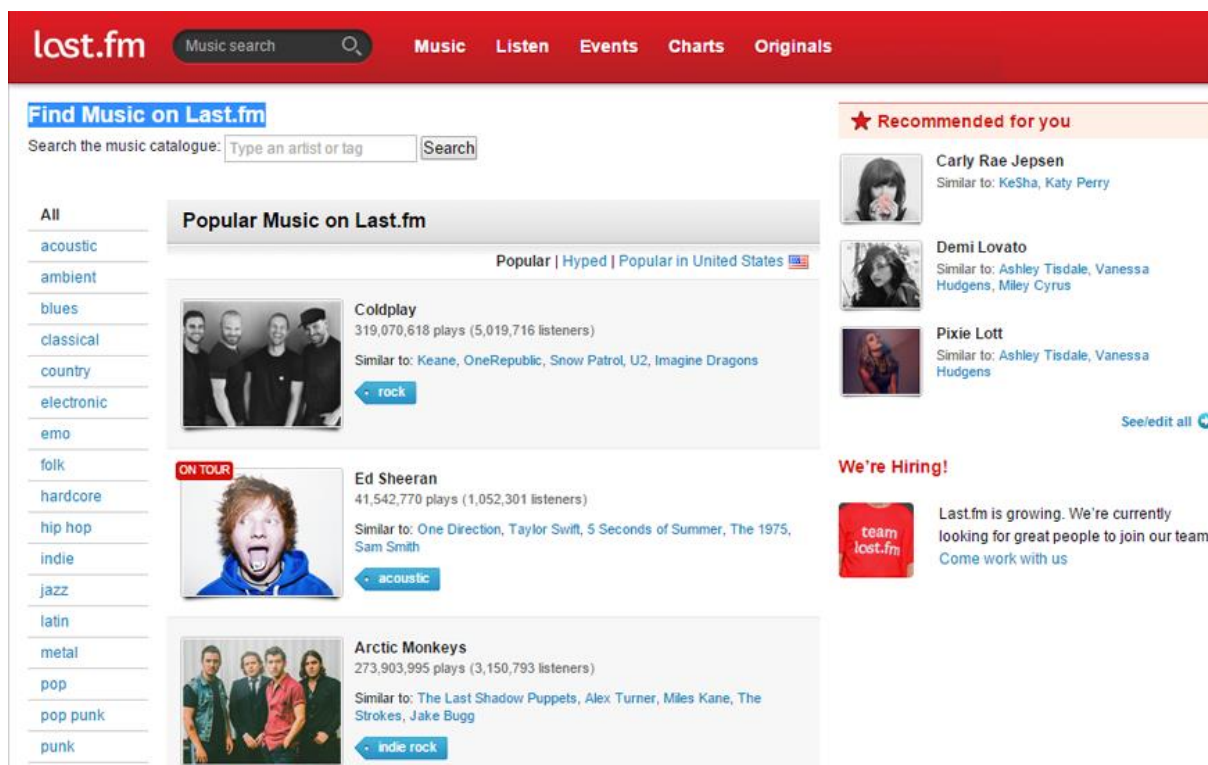
Dinaminis grojaraščio sudarymo metodas – tai pagerintas automatinis grojaraščio sudarymo metodas, nes yra atsižvelgiama į vartotojo veiksmus (pvz. dainos prasukinėjimas, dainos reitingavimas). Pampalk [16] siūlomame metode grojaraščio generavimas pradamas kurti pradėjus groti dainai. Sekanti grojaraščio daina yra parenkama, atsižvelgiant į tai, ar vartotojas išklausė dainą iki galo, ar perjungė dainą, neišklausęs jos iki galo. Visos grojaraščio dainos, kurios panašios į prasuktas dainas, yra pašalinamos iš grojaraščio, o paliekamos tik tos, kurios panašios į išklaustas dainas. Tokiu būdu sugeneruojamas grojaraštis labiau tenkins klausytojo muzikos klausymo poreikius, nes yra atsižvelgiama į klausytojo veiksmus. Panašus metodas yra taikomas ir „Pandora“ muzikos rekomendavimo sistemoje (žr. 3 pav. ) [17].



3 pav. „Pandora“ muzikos rekomendavimo sistema

### 1. 3. 3 Bendro filtravimo grojaraščio sudarymo metodas

Bendro filtravimo grojaraščio sudarymo metodu vartotojo grojaraštis generuojamas, atsižvelgiant į vartotojo klausomas dainas ir į tai, kaip tos dainos buvo įvertintos kitų vartotojų [7]. Tačiau šis metodas reikalauja labai daug vartotojų ir daug dainų, kurios būtų reitinguotos kitų vartotojų. Taip pat toks metodas negali muzikos klausytojui pasiūlyti dainos, jeigu ji nebuvo nei karto reitinguota kitų klausytojų, nors daina ir atitinka klausytojo muzikos žanrą. Tai ypač apsunkina naujų dainų įtraukimą į vartotojams sudarytus grojaraščius. Šis principas dažniausiai naudojamas įvairiuose socialiniuose tinklapiuose, kurie turi didelį aktyvių vartotojų ratą. Tokios sistemos tipinis pavyzdys – „Last.fm“ sistema (žr. 4 pav. ).

The image is a screenshot of the Last.fm website. At the top, there is a red navigation bar with the Last.fm logo and a search bar. Below the navigation bar, there is a section titled "Find Music on Last.fm" with a search input field. To the left of the main content area is a vertical list of music genres. The main content area is titled "Popular Music on Last.fm" and features three artist profiles: Coldplay, Ed Sheeran, and Arctic Monkeys. Each profile includes a photo, the artist's name, the number of plays, the number of listeners, and a list of similar artists. To the right of the main content area, there is a "Recommended for you" section with three artist profiles: Carly Rae Jepsen, Demi Lovato, and Pixie Lott. Below this is a "We're Hiring!" section with a red button and text.

4 pav. „Last.fm“ muzikos rekomendavimo sistema

### 1. 3. 4 Dainos turiniu pagrįstas grojaraščio sudarymo metodas

Dažniausiai grojaraštis yra generuojamas, apskaičiuojant panašumą tarp dainų. Tačiau įvertinti panašumą tarp dviejų dainų gali būti pakankamai sudėtinga. Cano [6] pasiūlė muzikos kūrinų palyginimo metodą, kuriame išgaunamos akustinės dainos savybės, tokias kaip tempas, tembras, metras, ritmas. Pirmiausia yra išgaunamos minėtos dainos savybės ir žiūrima, kokios dainos klausosi vartotojas. Toliau yra ieškoma daina su panašiausiomis dainos savybėmis į vartotojo klausomas dainos savybes. Minėtam metodui nereikia, kad vartotojo muzikos įrašai būtų reitinguoti kitų vartotojų, kaip kad bendro filtravimo metode. Tuo tarpu Cai [4] atliktame darbe siūloma muzikos grojaraštį generuoti, atsižvelgiant į dainų emocijų panašumą. Tokiu atveju būtų remiamasi ne pirmine muzikos kūrinio informacija, bet jos interpretacija, atitinkančia tam tikrą emociją.

Nepaisant minėtų privalumų, dainos turiniu pagrįstas grojaraščio sudarymo metodas turi ir trūkumų. Kadangi šiuo metodu yra skaičiuojamas panašumas tarp dainų grojaraštyje, gali susidaryti tokia situacija, kai tik kelios dainos bus panašios į vartotojo klausomą dainą. To pasekoje grojaraštyje bus grojamos pastoviai tos pačios dainos. Taip pat turint didelį kiekį muzikos kūrinų, grojaraštis gali

būti lėtai generuojamas, nes reikės iš visų grojaraščio dainų išgauti dainos savybes ir dar surasti panašumą tarp dabar klausomos vartotojo dainos, o tai dėl savo sudėtingumo gali užtrukti.

#### *1. 3. 5 Dainos meta-duomenimis paremtas grojaraščio sudarymo metodas*

Muzikos kūrinio meta-duomenimis paremtas grojaraščio sudarymo metodas yra vienas pagrindinių ir lengviausių būdų generuoti muzikos grojaraštį. Jame ieškant muzikos įrašų yra naudojami dainos meta-duomenys, tokie kaip: atlikėjas, dainos sukūrimo metai, žanras, ritmas, dainų žodžiai, albumo pavadinimas, nuotaika ir t.t. [10]. Visi šie duomenys yra lengvai gaunami iš muzikos kūrinio aprašymo ir nereikalauja akustinės ar emocinės analizės.

Gulik [22] atliktame darbe grojaraštis generuojamas, atsižvelgiant į tokius dainos meta duomenis, kaip nuotaika, dainos sukūrimo metai, ritmas, žanras. Yu [9] savo darbe grojaraštį generuoja, atsižvelgiant į dainos žanrą bei dainos sukūrimo metus. Nors šie grojaraščio generavimo metodai atsižvelgia į dainos meta-duomenis ir dėl to yra greiti ir gan tikslūs, tačiau turi ir trūkumų. Į grojaraštį galima įtraukti tik tuos muzikos kūrinius, kurie turi meta-duomenis. Dažnai būna, kad dainos neturi aprašytų meta-duomenų, jie nėra išsaugoti muzikos kūrinio faile ar duomenų bazėje. Tokiu atveju tokios dainos nebus įtrauktos į generuojamą grojaraštį, nes nebus galima įvertinti jų panašumo su kitais muzikos kūriniais. Taip pat toks metodas atsižvelgia tik į dainos meta-duomenis ir ignoruoja klausytojo veiksmus, tokius kaip: dainos prasukinėjimas, nepatikusios dainos perjungimas, dainos įvertinimas.

#### *1. 3. 6 Kontekstu paremtas grojaraščio sudarymo metodas*

Konteksto informacija yra apibrėžiama, kaip aplinka aplink muziką besiklausantį vartotoją. Ji apima fizinę vietą, aktyvumą, dabartinę nuotaiką ir t.t. Su ir Yeh [26] patobulino bendro filtravimo metodą, naudojant vartotojų grupavimą pagal konteksto informaciją. Šie autoriai atsižvelgia į vartotojo vietas, judesio, aplinkos, nuotaikos informaciją. Gauta konteksto informacija padeda tiksliau parinkti grojaraščio dainas klausytojui, nes, pavyzdžiui žinant klausytojo nuotaiką galima į ją atsižvelgti parinkinėjant grojaraščio dainas. Kai kurie mokslininkai teigė, kad konteksto informacijos naudojimas aplenkė dainos turiniu pagrįstą modelį [25, 28]. Tačiau šis metodas turi tas pačias problemas, kaip ir bendro filtravimo metodas.

#### *1. 3. 7 Hibridinis grojaraščio sudarymo metodas*

Hibridiniu metodu siekiama geresnių rezultatų, apjungiant kelis muzikos rekomendavimo metodus. Burke [3] nurodė keletą metodų, kuriuos galima naudoti kuriant hibridinį muzikos rekomendavimo metodą: svertinis, perjungimo, maišymo, savybių kombinavimo ir pakopinis metodas. Shao [23] savo darbe apjungė dainos turiniu grindžiamą metodą su dinaminiu grojaraščio sudarymo metodu. Tokiu būdu grojaraštis generuojamas, atsižvelgiant ir į dainos turinio savybes, ir į klausytojo veiksmus. Nėra jokių abejonių, kad tinkamas hibridinis metodas yra pranašesnis už kitus metodus, nes gali įtraukti privalumus keleto metodų, nepaveldint jų trūkumų [11, 12, 24]. Tačiau aprašytuose moksliniuose darbuose pateikiamos tik bendros grojaraščių sudarymo tendencijos, kurių nepakanka pilnai funkcionuojančio prototipo ar sistemos sukūrimui.

Apibendrinant matoma, kad dauguma kompanijų naudoja automatinius grojaraščių sudarymo algoritmus, tačiau jų neatskleidžia, nes dauguma jų produktų bei sistemų laikosi minėtų algoritmų veikimo principų. Apžvelgus muzikos rekomendavimo metodus bei sistemas, kuriomis naudojasi milijonai vartotojų, galima teigti, kad ateityje tokie grojaraščių sudarymo (muzikos rekomendavimo) metodai tik dar labiau tobulės ir tokių sistemų daugės, nes jos padeda muzikos klausytojams klausytis tik jiems patinkančios muzikos ir taupo jų brangų laiką, nes vis rečiau reikia prasukinėti nepatinkančias dainas ar kurti grojaraščius individualiai.

## **2. PSEUDO ATSITIKTINIS GROJARAŠČIŲ SUDARYMO ALGORITMAS, ATSIŽVELGIANTIS Į MUZIKOS KLAUSYMO ISTORIJĄ**

Algoritmų analizė parodė, kad dauguma kompanijų savo produktuose naudoja automatinius grojaraščių sudarymo algoritmus, tačiau tie algoritmai yra komerciniai ir jie nėra viešai atskleidžiami, todėl nėra galimybės jų išanalizuoti ir nustatyti veikimo principus. Tik keleto mokslininkų tyrimuose galima rasti jų siūlomų grojaraščių sudarymo metodų aprašymus ir savybių, į kurias atsižvelgiama sudarant grojaraščius sąrašą. Tačiau patys algoritmai dažnai nėra detalizuojami, o tik glaustai aprašoma, į kokias savybes atsižvelgiama generuojant grojaraštį.

Remiantis dinaminio grojaraščio sudarymu, atsižvelgiant į vartotojo elgseną klausantis muzikos, bei grojaraščio sudarymu, atsižvelgiant į muzikos įrašų meta duomenis, siūloma apjungti šiuos metodus. Papildomai siūloma atsižvelgti ne tik į vartotojo elgseną, klausantis muzikos kūrinio, meta-duomenis, bet ir į muzikos klausymo istoriją. Tai svarbu, nes kaupiant informaciją apie jau klausytus muzikos kūrinius ir tinkamai ją panaudojant, tikėtina, kad bus galima tiksliau rekomenduoti sekančias grojaraščio dainas. Tai būtų naujas hibridinis muzikos rekomendavimo metodas, kuriame keli metodai yra apjungiami į vieną (žr. 1. 3. 7 skyrių).

### **2.1 Siūlomo algoritmo principai**

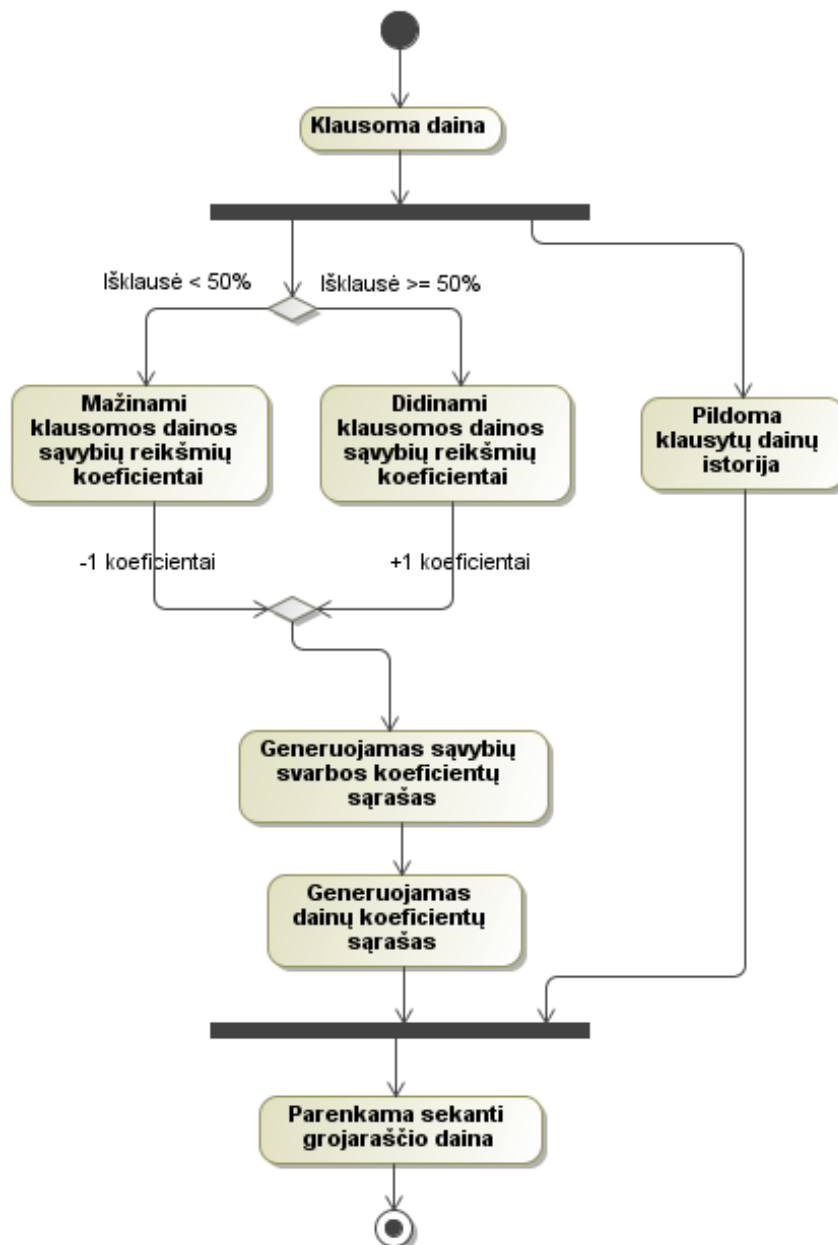
Pagrindinė pseudo atsitiktinio grojaraščių sudarymo, atsižvelgiant į muzikos klausymo istoriją, algoritmo idėja – sumažinti vartotojo įsikišimą, klausantis sugeneruoto grojaraščio ir tuo pat metu padidinti vartotojo pasitenkinimą, parenkant tokias dainas, kurios klausytojui patiktų ir jis nenorėtų jų perjunginėti ar prasukinėti. Šis algoritmas laikomas pseudo atsitiktiniu, nes pačiam vartotojui turi atrodyti, kad jam parenkamos dainos visiškai atsitiktinai, nesikartoja tas pats dainų eiliškumas klausantis. Tuo tarpu iš tikro daina parenkama pagal konkretų algoritmą, kuris atsitiktinai parenka dainą tik tada, kai atrinkus kelis galimus tolesnės dainos variantus, nėra aišku kurią dainą groti pirmiau.

Siekama, kad algoritmas veiktų taip, kad vartotojas kuo rečiau norėtų perjunginėti dainas, o pačiam vartotojui nereikėtų, kaip kituose nagrinėtuose metoduose, dar papildomai pačiam įvertinti dainas reitingais ar dar kokio kito papildomo įsikišimo.

Siūlomo algoritmo veikimo schema pateikta 5 pav. Pirmiausia yra fiksuojama, kokią dainą klausosi vartotojas ir ją algoritmas iš karto įrašo į klausytų dainų istoriją. Tada yra fiksuojami klausomos dainos meta-duomenys. Taip pat fiksuojamas pilnas bei nepilnas dainos išklausymas (žr. 2. 1. 1 skyrių). Toliau yra generuojamas savybių svarbos koeficientų sąrašas (žr. 2. 1. 2 skyrių), kuriame priklausomai nuo dainos išklausymo yra didinami arba mažinami dainos savybių reikšmių koeficientai. To pasekoje yra gaunamas dainų savybių reikšmių koeficientų sąrašas. Po to yra generuojamas dainų koeficientų sąrašas (žr. 2. 1. 3 skyrių). Pirmiausia yra tikrinama, ar dainos savybių reikšmės egzistuoja savybių svarbos koeficientų sąrašė. Jeigu randamos egzistuojančios reikšmės, tuomet tų rastų reikšmių koeficientai iš savybių svarbos koeficientų sąrašo yra sumuojami ir taip gaunamas dainos koeficientas. Tokiu principu visos grojaraštyje esančios dainos yra patikrinamos ir susumuojami jų koeficientai, kurie pastoviai kinta, klausantis muzikos kūrinių. Galiausiai sekanti



daina parenkama, atsižvelgiant į dainą su didžiausiu koeficientu iš dainų koeficientų sąrašo, bei į tai, ar šios dainos nėra dainų klausymo istorijoje (žr. 2. 1. 4 skyrių).



5 pav. Siūlomo algoritmo bazinė schema

Nors šioje schemoje pateikiama dainos išklausymo riba lygi 50%, tačiau šią ribą galima keisti ir taip adaptuoti algoritmą skirtingo tipo vartotojams.

### 2. 1. 1 Dainos išklausymas

Hu [9] nagrinėtame metode pilnas bei nepilnas dainos išklausymas yra traktuojamas taip, kad jeigu klausytojas klausosi dainos ir ją perjungia neišklausęs iki galo, reiškia, kad daina klausytojui nepatiko, o jeigu dainą išklausė nuo pradžios iki galo, reiškia, kad jam daina patiko. Tačiau šitoks dainų išklausymo traktavimas nėra labai geras, nes visų pirma nėra atsižvelgiama į tai, kad pvz. vartotojas klausosi grojamos dainos ir išklausius kokius 90 % dainos, jis perjungia kitą, nes pati dainos pabaiga jam nepatiko, nors visa kita jam patiko, tokiu atveju dainą būtų galima traktuoti, kaip patiko. Kitas atvejis, kai pvz. vartotojas nenori klausytis pačios dainos pradžios, tad kokį 10 % dainos praleidžia ir toliau išklausia dainą iki galo, nes dažnai būna, kad dainų pradžios būna nuobodžios arba

labai ištemptos. Tokiu atveju daina vėl bus traktuojama, kaip nepatikusi, nors ji bus išklaudyta iki galo. Siūlomame pseudo atsitiktinio grojaraščių sudarymo, atsižvelgiant į muzikos klausymo istoriją, algoritme, dainos išklaudytas traktuojamas taip:

Pirmiausia yra pasirinkta optimali dainos išklaudyto riba 50 %, tačiau svarbu paminėti, kad norint šią ribą galima keisti, atsižvelgiant į pačio vartotojo ar taikymo srities reikalavimus. Tad šiuo atveju jei klausytojas dainą išklaudė mažiau nei 50 %, tuomet traktuojama, kad daina nepatiko, o jeigu klausytojas išklaudė 50 % dainos ar daugiau, tuomet traktuojama, kad daina patiko. Taip pat svarbu pažymėti, kad atsižvelgiama ir į tai, ar klausoma daina buvo prasukinėjama. Jeigu pvz. klausytojas prasuko dainos pradžią kokį 20 %, po to klausėsi iki dainos vidurio, t.y. 50 %, tai dar nereiškia, kad daina išklaudyta 50 %. Šiuo atveju daina išklaudyta tik 30 % ir jeigu vartotojas toliau klausysis dainos, pvz. bent 20 %, tada jau bus iš viso išklaudęs 50 % dainos – daina bus traktuojama kaip patikus.

Apibendrinant, siūlomame pseudo atsitiktinio grojaraščių sudarymo, atsižvelgiant į muzikos klausymo istoriją, algoritme yra fiksuojama, kiek tiksliai procentų klausytojas išklaudė dainos, atsižvelgiant, kad daina gali būti prasukinėjama. Tokiu būdu tiksliau galima fiksuoti, ar klausoma daina patiko klausytojui, ar ne.

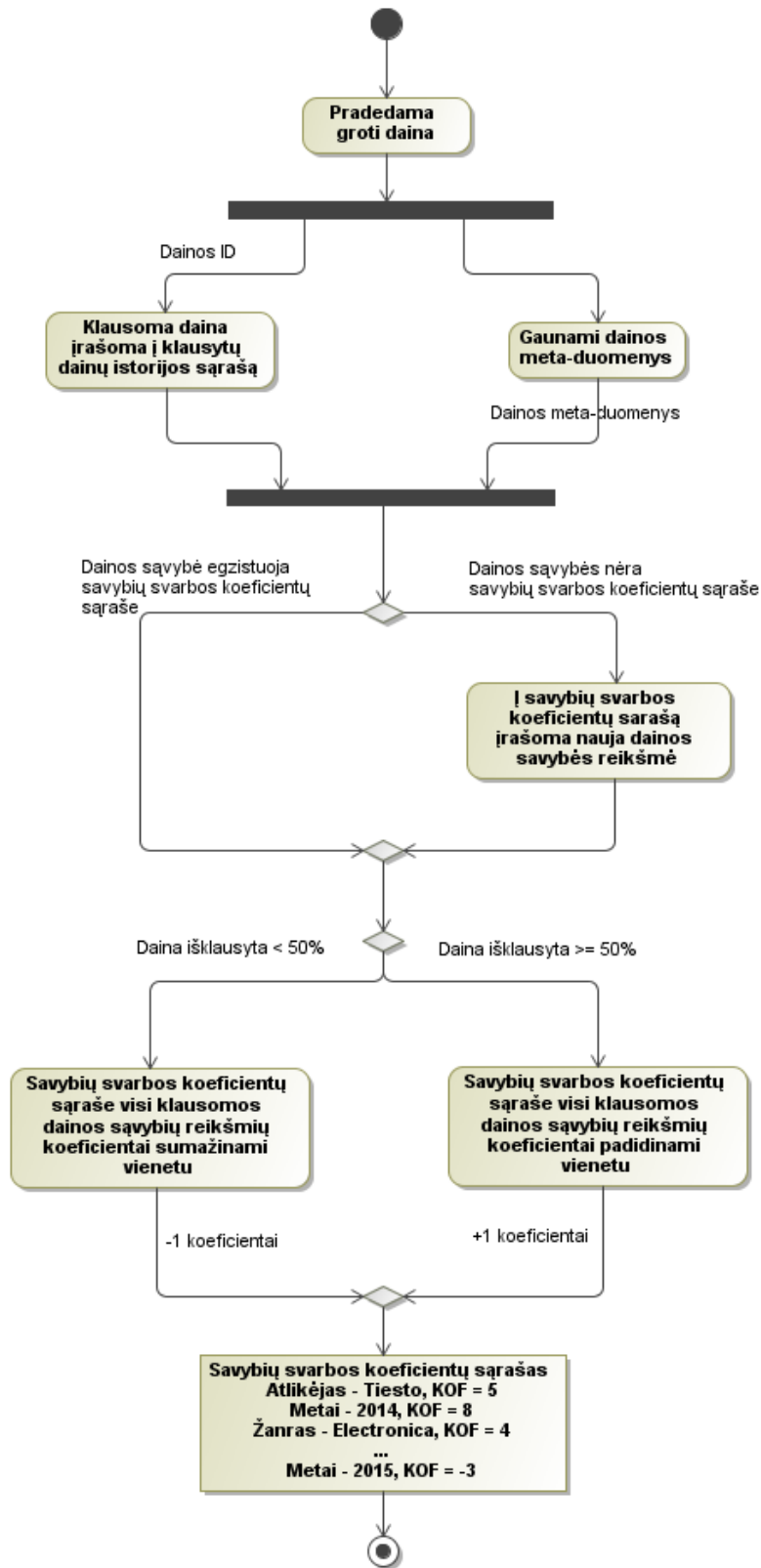
#### *2. 1. 2 Savybių svarbos koeficientų sąrašas*

Vartotojui tik pradėjus klausytis dainos, yra generuojamas savybių svarbos koeficientų sąrašas (žr. 6 pav. ). Pirmiausia tik pradėjus groti dainai, nuskaitomi jos meta-duomenys ir ji yra įrašoma į klausytų dainų istorijos sąrašą. Tada tikrinama, kiek procentų dainos yra išklaudyta. Jeigu išklaudyta mažiau kaip pusę dainos, reiškia, kad daina nepatiko. Tuomet yra tikrinama, ar savybių svarbos koeficientų sąraše egzistuoja klausomos dainos savybių reikšmės. Kad būtų aiškiau, pateikiamas pavyzdys:

Klausoma daina turi keturias savybes – tai atlikėjas, metai, žanras ir ritmas. Kiekviena savybė turi reikšmes, pvz.:

- Atlikėjas: Tiesto;
- Metai: 2014;
- Žanras: Electronica;
- Ritmas: Greitas.

Tai reiškia, kad bus tikrinama, ar savybių svarbos koeficientų sąraše yra tokia savybės reikšmė, kurią turi klausoma daina. Jeigu tokios savybės reikšmės nėra savybių svarbos koeficientų sąraše, tuomet ta savybės reikšmė yra įrašoma į savybių svarbos koeficientų sąrašą. Kai patikrinamos, ar egzistuoja dainos savybių reikšmės savybių svarbos koeficientų sąraše, toliau, jeigu klausoma daina traktuojama kaip nepatinkanti, visų savybių svarbos koeficientų sąraše sutampančių savybių reikšmių su klausomos dainos savybių reikšmėmis koeficientai yra mažinami vienetu. 7 pav. pateikta, kaip keičiasi savybių svarbos koeficientų sąrašas, kai klausoma daina nepatiko.



6 pav. Savybių svarbos koeficientų sąrašo generavimo diagrama

Pateiktoje schemoje aiškiai matosi, kad ne visos klausomos dainos savybių reikšmės buvo savybių svarbos koeficientų sąrašė, todėl sąrašas papildomas naujomis reikšmėmis. Šiuo atveju tai atlikėjas „Tiesto“ ir žanras „Trance“ (7 pav. pažymėta raudonai). Taip pat, kadangi klausoma daina nepatiko, sutampančių savybių reikšmių koeficientai sumažinami vienetu. Kadangi sutampančios savybių reikšmės yra „2014“, „Greitas“, „Tiesto“, „Trance“ (7 pav. pažymėta geltonai), šių reikšmių koeficientai sumažinami vienetu.

Savybių svarbos koeficientų sąrašas prieš klausantis dainos		
Savybė	Reikšmė	Koeficientas
Atlikėjas	Michael Jackson	1
Metai	2014	2
Žanras	POP	1
Ritmas	Greitas	4



Dainos meta-duomenys	
Savybė	Reikšmė
Atlikėjas	Tiesto
Metai	2014
Žanras	Trance
Ritmas	Greitas

Savybių svarbos koeficientų sąrašas kai klausoma daina nepatiko		
Savybė	Reikšmė	Koeficientas
Atlikėjas	Michael Jackson	1
Metai	2014	<b>1</b>
Žanras	POP	1
Ritmas	Greitas	<b>3</b>
<b>Atlikėjas</b>	<b>Tiesto</b>	<b>-1</b>
<b>Žanras</b>	<b>Trance</b>	<b>-1</b>

7 pav. Savybių svarbos koeficientų sąrašo kitimas, kai klausoma daina nepatiko

Jeigu yra išklaudyta pusė dainos ar daugiau, reiškia, kad daina patiko. Tuomet vėl yra tikrinama, ar savybių svarbos koeficientų sąrašė egzistuoja klausomos dainos savybių reikšmės ir atitinkamai yra sąrašas pildomas naujomis savybių reikšmėmis, o savybių reikšmių koeficientai savybių svarbos koeficientų sąrašė yra didinami vienetu. 8 pav. pateikta, kaip keičiasi savybių svarbos koeficientų sąrašas, kai klausoma daina patiko.

Pateiktoje scheme matosi, kokių dainos savybių reikšmių nebuvo savybių svarbos koeficientų sąrašė (8 pav. pažymėta raudonai). Taip pat matosi sutampančios klausomos dainos ir savybių svarbos koeficientų sąrašo savybių reikšmės (8 pav. pažymėta baltai). Tai „2014“, „Greitas“, „Tiesto“, „Trance“, todėl šių savybių reikšmių koeficientai yra padidinami vienetu, nes klausoma daina patiko klausytojui (8 pav. pažymėta geltonai).

Taip pat svarbu pažymėti, kad yra nemažai muzikos kūrinių, kurie yra atliekami keleto atlikėjų (pvz. duetai) bei kurių žanrai ne visada priklauso tik vienam žanrui. Tokiu atveju iš dainos meta-duomenų išgaunami visi muzikos kūrinių atlikėjai, žanrai bei kitos savybės. Ko pasekoje gaunamas didesnis ir tikslesnis savybių svarbos koeficientų sąrašas, kuris padės tiksliau parinkti sekančią grojaraščio dainą, kuri turėtų patikti klausytojui ir jis nenorėtų jos perjungti.

Savybių svarbos koeficientų sąrašas prieš klausantis dainos		
Savybė	Reikšmė	Koeficientas
Atlikėjas	Michael Jackson	1
Metai	2014	2
Žanras	POP	1
Ritmas	Greitas	4



Dainos meta-duomenys	
Savybė	Reikšmė
Atlikėjas	Tiesto
Metai	2014
Žanras	Trance
Ritmas	Greitas

Savybių svarbos koeficientų sąrašas kai klausoma daina patiko		
Savybė	Reikšmė	Koeficientas
Atlikėjas	Michael Jackson	1
Metai	2014	3
Žanras	POP	1
Ritmas	Greitas	5
<b>Atlikėjas</b>	<b>Tiesto</b>	<b>1</b>
<b>Žanras</b>	<b>Trance</b>	<b>1</b>

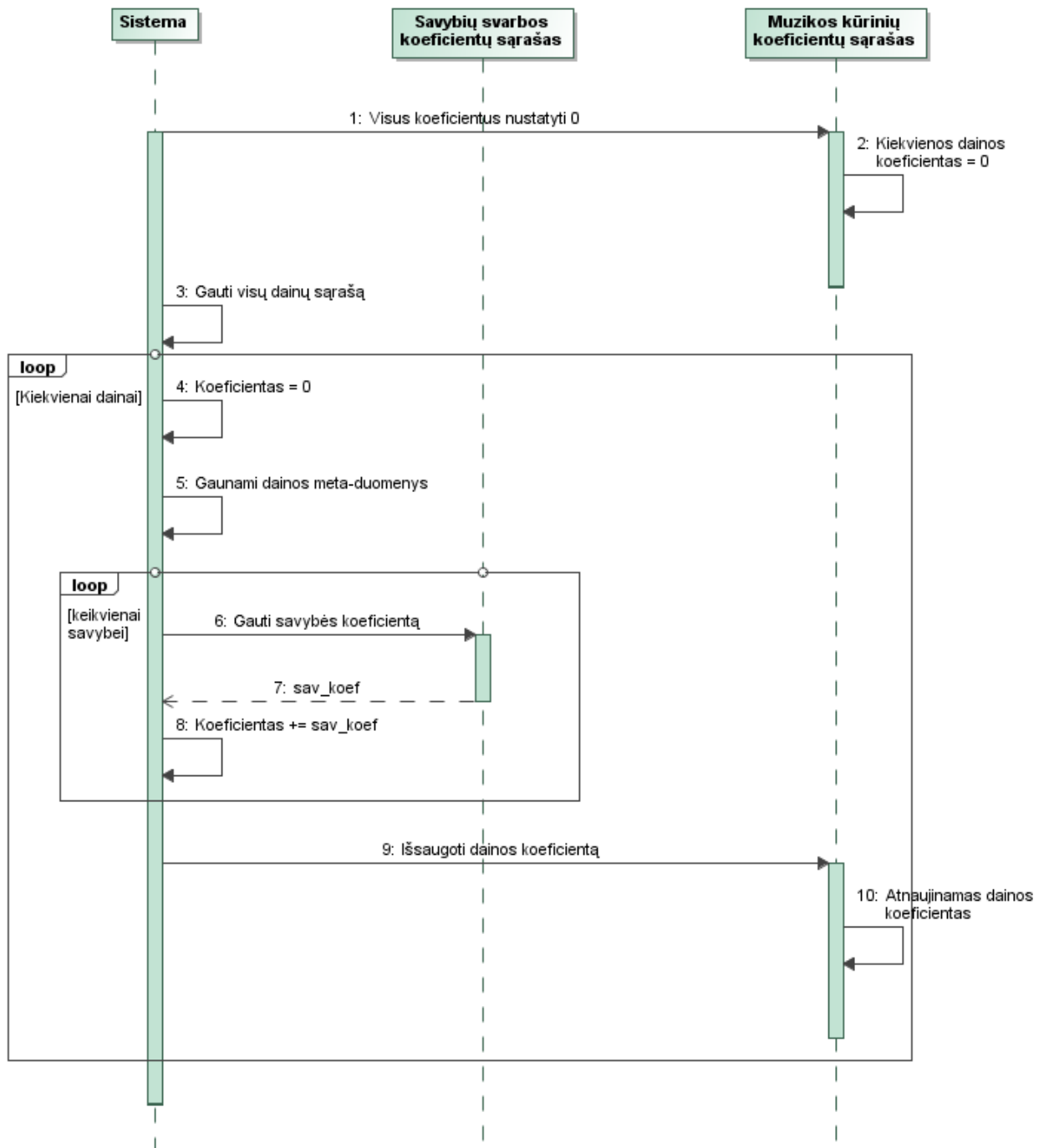
8 pav. Savybių svarbos koeficientų sąrašo kitimas, kai klausoma daina patiko

Svarbu pastebėti, kad kai pradeda klausyti bet kuri daina iš grojaraščio, siūlomas algoritmas visada dainą traktuoja kaip nepatinkančią, nes dainą tik pradėjus klausytis, dar ji nėra išklausyta iki 50 % ir kai ši riba yra pasiekama ar viršijama, tik tada yra daina traktuojama kaip patikusi ir visi klausomos dainos savybių reikšmių koeficientai yra padidinami.

### 2. 1. 3 Dainų koeficientų sąrašas

Baigus generuoti savybių svarbos koeficientų sąrašą, toliau yra generuojamas dainų koeficientų sąrašas (žr. 9 pav. ).

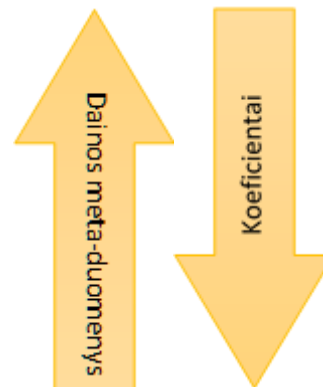
Iš pateiktos diagramos matosi, kad siūlomas algoritmas pirmiausiai visų dainų koeficientus nustato nuliais, esančius muzikos kūrinių koeficientų sąrašė. Tuomet yra gaunamas visų grojaraštyje esančių dainų sąrašas. Toliau yra nuskaitomi kiekvienos dainos meta-duomenys ir tikrinama, ar tos dainos savybių reikšmės egzistuoja savybių svarbos koeficientų sąrašė. Radus egzistuojančias reikšmes savybių svarbos koeficientų sąrašė, paimamas rastos savybės reikšmės koeficientas ir sumuoja jį prie dainos koeficiento, dainos koeficientų sąrašė. Baigus sumuoti visus dainos koeficientus, gautas koeficientas yra atnaujinamas muzikos kūrinių koeficientų sąrašė. 10 pav. pateikta, kaip keičiasi dainos koeficientų sąrašas, kai randamos egzistuojančios reikšmės savybių svarbos koeficientų sąrašė.



9 pav. Dainų koeficientų sąrašo generavimo diagrama

Pateiktoje schemoje matosi, kaip kinta dainų koeficientų sąrašas. Pvz. daina, kurios id = 1, turi reikšmes „Tiesto, 2012, trance, greitas“. Patikrinus, ar savybių svarbos koeficientų sąrašė egzistuoja tokios reikšmės, paaiškėja, kad egzistuoja reikšmė „Tiesto“, kurios koeficientas 2, reikšmė „Trance“, kurios koeficientas 2 bei reikšmė „Greitas“, kurios koeficientas 2. Tad susumavus visų egzistuojančių reikšmių koeficientus, gauname, kad dainos, kurios id = 1, koeficientas lygus 6. Tokiu principu visoms dainoms yra suskaičiuojami dainų koeficientai.

Savybių svarbos koeficientų sąrašas		
Sąlybė	Reikšmė	Koeficientas
Atlikėjas	Michael Jackson	3
Metai	2014	4
Žanras	POP	3
Ritmas	Greitas	2
Atlikėjas	Tiesto	2
Metai	2013	2
Žanras	Trance	2
Atlikėjas	Eminem	-2
Metai	2011	-1
Žanras	Hip-hop	-2



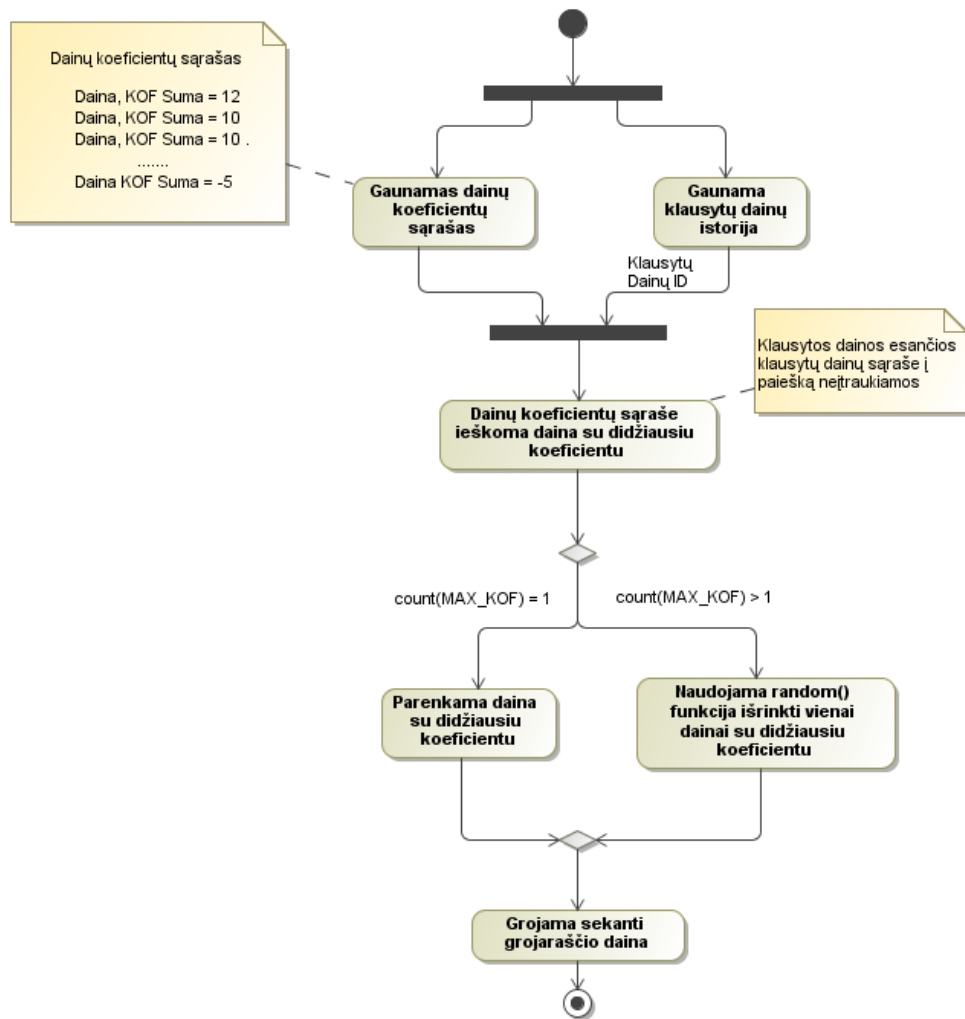
Dainų koeficientų sąrašas					
Dainos id	Dainos meta-duomenys				Koeficientas
	Atlikėjas	Metai	Žanras	Ritmas	
1	Tiesto	2012	Trance	Greitas	<b>6</b>
2	Tiesto	2014	Trance	Greitas	<b>10</b>
3	Michael Jackson	2014	POP	Vidutinis	<b>10</b>
4	Linkin Park	2012	Rock	Vidutinis	<b>0</b>
5	Rihanna	2013	R&B	Vidutinis	<b>2</b>
6	Katy Perry	2014	POP	Vidutinis	<b>7</b>
...	...	...	...	...	...
100	Eminem	2011	Hip-hop	Vidutinis	<b>-5</b>

*10 pav. Dainų koeficientų sąrašo kitimas*

Pateiktoje schemoje matosi, kaip kinta dainų koeficientų sąrašas. Pvz. daina, kurios id = 1, turi reikšmes „Tiesto, 2012, trance, greitas“. Patikrinus, ar savybių svarbos koeficientų sąrašė egzistuoja tokios reikšmės, paaiškėja, kad egzistuoja reikšmė „Tiesto“, kurios koeficientas 2, reikšmė „Trance“, kurios koeficientas 2 bei reikšmė „Greitas“, kurios koeficientas 2. Tad susumavus visų egzistuojančių reikšmių koeficientus, gauname, kad dainos, kurios id = 1, koeficientas lygus 6. Tokiu principu visoms dainoms yra suskaičiuojami dainų koeficientai.

#### *2. 1. 4 Sekančios dainos parinkimas*

Galiausiai, baigus generuoti dainų koeficientų sąrašą, renkama sekanti grojaraščio daina (žr. 11 pav. ). Pirmiausia dainų koeficientų sąrašė yra ieškoma daina su didžiausiu koeficientu, nes kuo didesnis koeficientas, reiškia, tuo labiau ši daina turėtų patikti klausytojui.



**11 pav.** Sekančios grojaraščio dainos parinkimo diagrama

Atliekant dainos paiešką, nėra įtraukiamos dainos, esančios klausytų dainų istorijoje, nes tada būtų užburtas ratas, jeigu kelios dainos iš eilės patiko ir jų koeficientai stipriai išauga, tuomet pastoviai tas dainas algoritmas parinkinėtų kaip sekancias, o norint sumažinti koeficientus, tektų prasukinėti labai daug dainų, tad toks algoritmo veikimas būtų neteisingas. Šiuo atveju svarbu paminėti, kad pačiam vartotojui leidžiama nusistatyti, kokį kiekį dainų saugoti klausymo istorijoje. Tai vartotojui leidžia prisitaikyti algoritmą pagal tai, po kelių dainų jis sutiktų dar kartą perklausyti tą pačią dainą. Todėl šioje vietoje pačiam klausytojui paliekama laisvė rinktis. Tokiu būdu algoritmas geriau prisitaiko prie klausytojų poreikių tenkinimo.

Suradus dainą su didžiausiu koeficientu, nustatoma, kad ta daina grojaraštyje ir gros sekanti. Tačiau gali susidaryti tokia situacija, kad kelios dainos gali turėti didžiausią vienodą koeficientą. Tokiu atveju yra atliekamas papildomas tikrinimas, kad jeigu surandama daugiau, nei viena daina, turinti didžiausią koeficientą ir jie yra lygūs, tuomet iš tų dainų atsitiktinai yra išrenkama viena daina, nes tų dainų tikimybė, kad patiks klausytojui, yra vienoda, nes koeficientai visų lygūs.

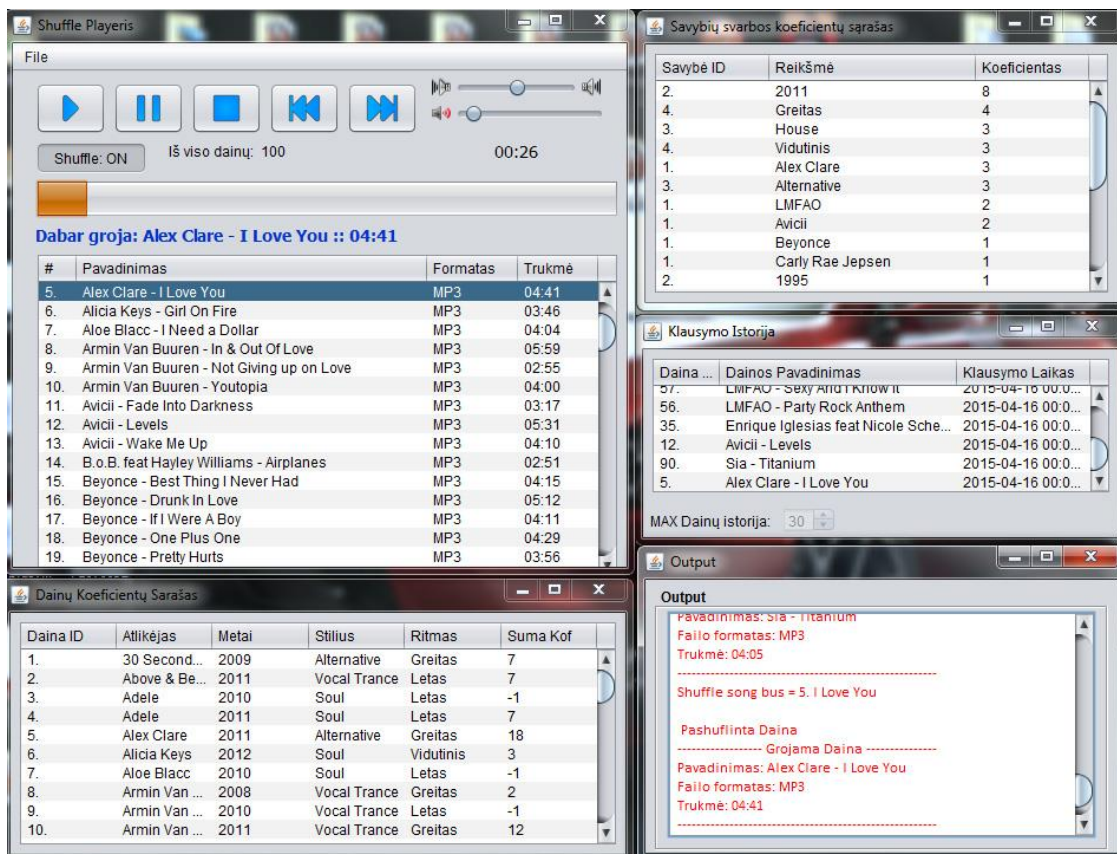
Apibendrinant siūlomo algoritmo veikimo principą, matoma, kad sekantios dainos grojaraštyje yra parinkinėjamos pagrinde tik pagal didžiausią dainos koeficientą, tačiau, tas koeficientas labai priklauso nuo to, kokias dainas prieš tai klausėsi klausytojas ir ar jam jos patiko (išklausė bent 50 % dainos) ar nepatiko (neišklausė 50 % dainos). Todėl šis algoritmas skirtas



adaptyviai prisitaikyti ir prie klausytojo kintančios nuotaikos, pvz. jeigu klausytojas kelias dainas pasiklausė POP stiliaus ir paskui užsinerėjo kitokio stiliaus, tai tik kelias dainas perjungus, dainų koeficientai pasikeičia kardinaliai ir sekančios dainos jau bus parinkinėjamos kito stiliaus nei POP, nes tų dainų koeficientai bus daug didesni.

## 2.2 Siūlomo algoritmo realizacija

Siūlomo pseudo atsitiktinio grojaraščių sudarymo, atsižvelgiantį į muzikos klausymo istoriją algoritmo teisingumui ir tinkamumui vertinti, buvo sukurtas prototipas (žr. 12 pav. ), kurio pagalba, patikrinta, jog siūlomas algoritmas veikia teisingai, pagal visus jo principus.



12 pav. Algoritmo realizacija

Algoritmo prototipas sukurtas, naudojant Java programavimo kalbą. Ši programavimo kalba pasirinkta, nes yra nepriklausoma nuo platformos (tai leis sukurtą realizacijos prototipą pritaikyti skirtingo tipo įrenginių ir operacinių sistemų palaikymui) ir yra objektiškai orientuota (todėl padidėja jos komponentų panaudojamumas, tobulinant esamą realizaciją). Taip pat svarbu pažymėti, kad naudojant šį prototipą, bus atliekamas algoritmo tyrimas bei eksperimentiškas palyginimas su kitais egzistuojančiais sprendimais, todėl šis prototipas turi pagrindines muzikos grotuvo funkcijas, kaip ir kiti muzikos grotuvai.

Taip pat svarbu paminėti, kad prototipo vartotojo sąsaja kurta kuo panašesnė į kitų muzikos grotuvų, nes atliekant tyrimą norima, kad jis būtų kuo objektyvesnis, todėl ir prototipas turi kuo mažiau išsiskirti iš kitų tiriamų egzistuojančių sprendimų.

### 3. PASIŪLYTO ALGORITMO TYRIMAS

#### 3.1 Tyrimo aplinka, duomenys, eiga ir sąlygos

Siekiant įvertinti pseudo atsitiktinį grojaraščių sudarymo algoritmą, atsižvelgiantį į muzikos klausymo istoriją, tinkamumą, reikia jį palyginti su kitais egzistuojančiais sprendimais ir ištirti, kaip kiekvienas algoritmas prisitaiko prie klausytojo muzikos klausymo poreikių.

Iš daugumos įvairiausių muzikos grotuvų lyginimui pasirinkti gan populiarūs ir gerai žinomi muzikos grotuvai „AIMP3 (AIMP)“, „Windows Media Player“ (WMP), kurie turi maišymo (*angl.* shuffle) funkciją, o tai reiškia, kad yra naudojami grojaraščių sudarymo algoritmai. Pagrindinis pasirinktų muzikos grotuvų motyvas toks, kad norima neblaškyti klausytojo ir siekiama objektyvumo nesusiejant su muzikos grotuvo išvaizda, o tik grojamomis dainomis. Pasirinktų grotuvų aplinkos yra panašios. Nėra perkrautos galybė įvairiausių funkcijų bei papildoma informacija, kas gali blaškyti klausytoją, kaip kitų muzikos grotuvų pvz. „iTunes“ ar „Spotify“. Juose aplinkos ženkliai skiriasi, pilna įvairiausių papildomų funkcijų, informacijos ir pan. Pvz. „Spotify“ muzikos grotuve net negalima paslėpti dainų sąrašo, kas šiame tyrime yra būtina siekiant gauti objektyvius tyrimo rezultatus.

Tyrimo objektyvumui užtikrinti, minėtuose muzikos grotuvuose, o taip pat ir prototipe, yra paslėpiamas dainų sąrašas, kad respondentas nežinotų, kokios muzikos kūriniai yra sąrašė, nes priešingu atveju, respondentas gali susidaryti išankstinę nuomonę apie grojaraštyje esančias dainas, ko pasekoje dainų klausymas bei tyrimo rezultatai nebūtų objektyvūs. Taip pat svarbu pažymėti, kad tyrimui naudojami muzikos grotuvai buvo pasirinkti atsižvelgiant ir į tai, kad jų visų aplinkos būtų panašios ir tyrimo rezultatams neturėtų įtakos. Siūlomo algoritmo prototipas, taip pat buvo kuriamas taip, kad būtų panašus į kitus tyrime naudojamus muzikos grotuvus. 13 pav. pateiktos tyrime naudojamų muzikos grotuvų aplinkos, kurias mato respondentas.

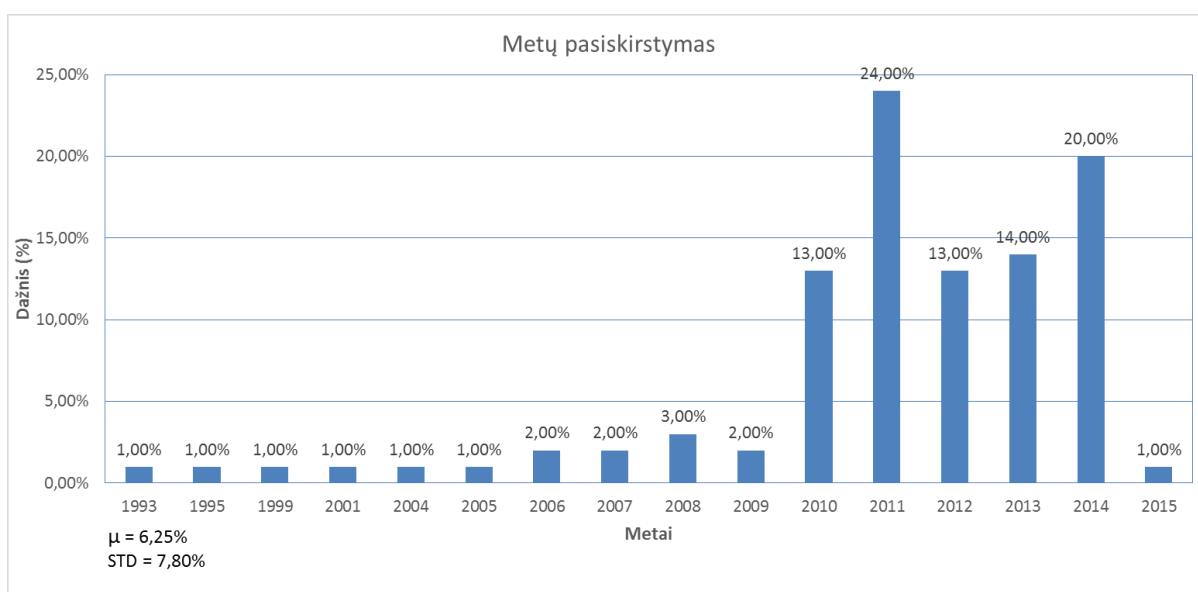


13 pav. Tyrime naudojamų muzikos grotuvų aplinkos

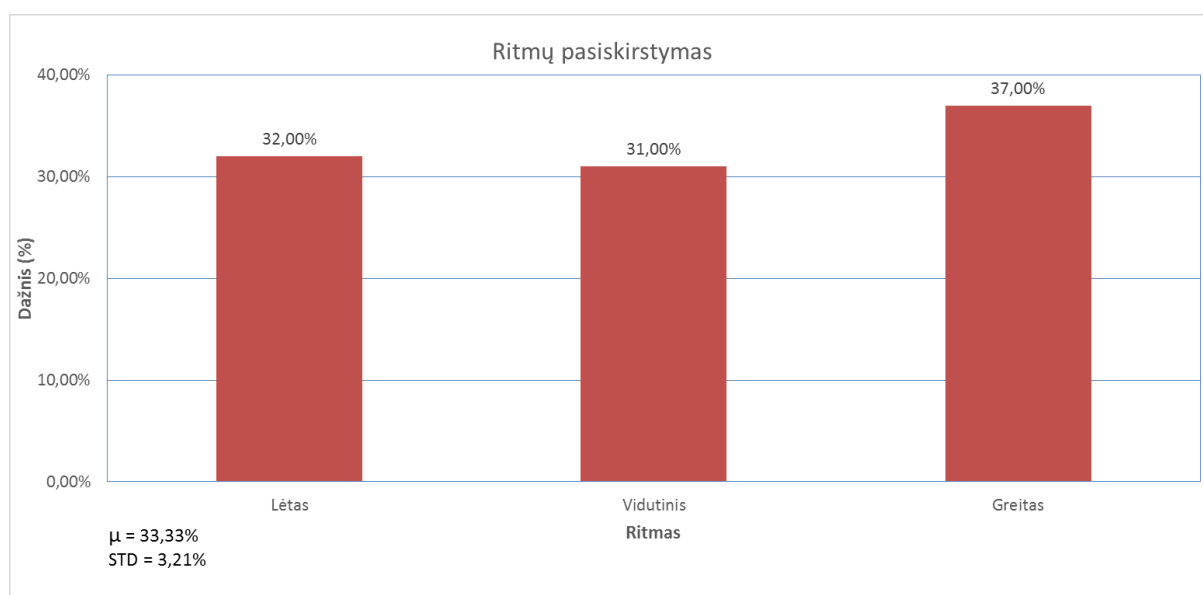
### 3. 1. 1 Tyrimo duomenys

Tyrimui atlikti buvo parinkta 100 skirtingų dainų. Visos dainos buvo užpildytos meta-duomenimis pasinaudojant „MusicBrainz“<sup>4</sup> dainų meta-duomenų baze, kur kiekvienai dainai buvo ieškomi tai dainai priklausantys meta-duomenys. Buvo nuspręsta naudoti tokius meta-duomenis kaip atlikėjas, metai, žanras bei ritmas, nes algoritmų analizė parodė, kad kiti tyrėjai dažniausiai naudojo minėtus meta-duomenis savo siūlytuose sprendimuose [9, 13, 22].

Svarbu pažymėti, kad sužinoti dainos ritmui, neužteko minėtos „MusicBrainz“ duomenų bazės, nes joje ritmai ne visiems muzikos įrašams yra pateikiami, todėl „BPM Counter“<sup>5</sup> programinės įrangos pagalba, kiekvienai dainai buvo randamas BPM (*angl.* beats per minute), pagal kurį, atžvelgiant į ritmų klasifikaciją pagal BPM [27], ritmai buvo suklasifikuoti į tris grupes: lėtas ritmas, vidutinis ritmas ir greitas ritmas. Tyrime naudotų meta-duomenų reikšmės ir jų pasiskirstymas, pateiktas **Error! Reference source not found.**, 14 pav. , 15 pav. paveikslėliuose.



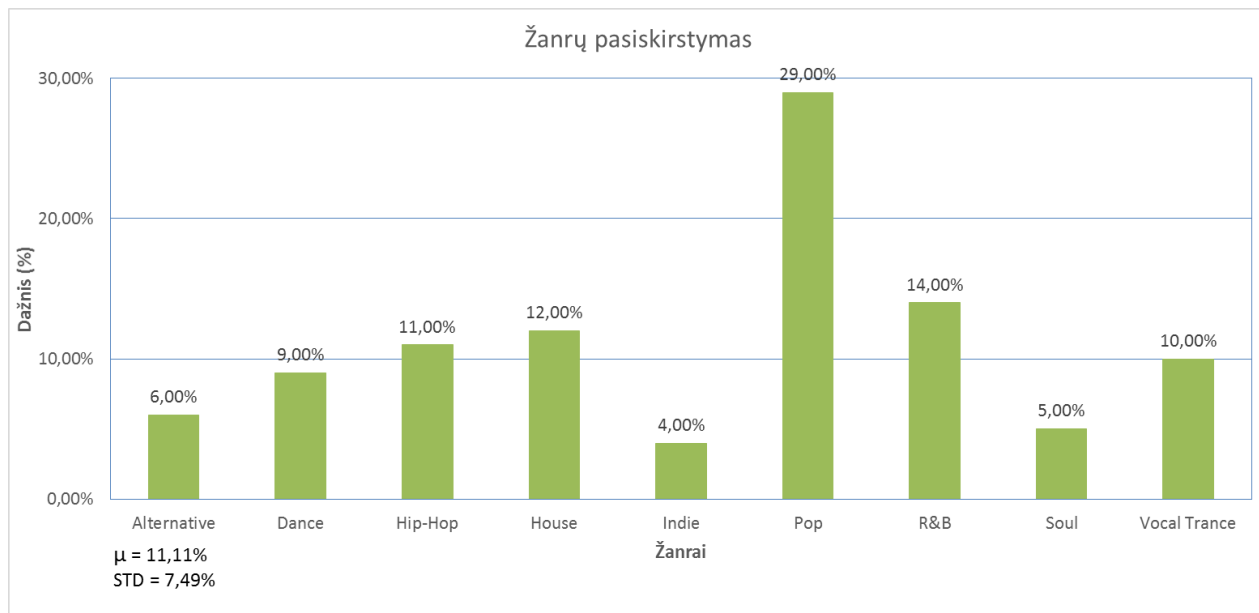
14 pav. Metų pasiskirstymas



15 pav. Ritmų pasiskirstymas

<sup>4</sup> <https://musicbrainz.org>

<sup>5</sup> <http://www.abysmedia.com/bpmcounter>



**16 pav.** Žanrų pasiskirstymas

### 3. 1. 2 Tyrimo eiga

Norint įvertinti pseudo atsitiktinių grojaraščių sudarymo algoritmo, atsižvelgiančio į muzikos klausymo istoriją, tinkamumą, atliekamas eksperimentas su respondentų grupe (10 respondentų). Pastariesiems duodama užduotis klausytis muzikos vieną valandą su kiekvienu tyrime naudojamu muzikos grotuvu, t.y. su „AIMP“, su prototipu (PRO) ir su „Windows Media Player“ (WMP). Respondentai eksperimento metu nežino kas bus vertinama, kokie parametrai bus fiksuojami ir t.t., jiems tiesiog nurodyta po valandą klausytis muzikos su skirtingais muzikos grotuvais. Respondentams leidžiama naudotis visomis pagrindinėmis muzikos grotuvo funkcijomis, tokiomis kaip: dainos prasukinėjimas, nepatikusios dainos perjungimas, dainos sustabdymas ir t.t.

Visos respondentų muzikos klausymo sesijos, su visais muzikos grotuvais, įrašinėjamos. Tokiu būdu gaunami respondentų klausymo sesijų vaizdo įrašai, kuriuos peržiūrėjus fiksuojama, kiek respondentas išklausė dainų iki galo, kiek respondentas išklausė dainų mažiau negu 50 %, fiksuojama, kokias dainas kiekvienos sesijos metu klausėsi respondentas, ko pasekoje stebima, kaip kito dainų išklausymas kiekvienos klausymo sesijos metu, iš ko turėtų matytis ar buvo perjunginėjamos nepatikusios dainos. Taip pat stebimas dainų meta-duomenų reikšmių kitimas klausymo metu, ko pasekoje turėtų paaiškėti kokie meta-duomenys turi įtakos grojaraščio generavimui.

Svarbu pažymėti, kad respondentams leista patiems rinktis, kada klausytis muzikos su kiekvienu muzikos grotuvu. Pavyzdžiui, jeigu būtų nurodyta klausytis muzikos iš eilės su visais muzikos grotuvais be jokių pertraukų, tai reikštų, kad būtų klausoma net tris valandas muzikos, ko pasekoje, respondentui greičiausiai nusibostų klausytis muzikos ir tarkim jau klausantis 3 valandą su paskutiniu tyrimo naudojamu muzikos grotuvu, respondentas jau daug tų pačių dainų būtų išgirdęs ir jam jos jau nepatiktų ir jis tik perjunginėtų vieną dainą po kitos, o kadangi tyrime naudojama 100 dainų, tai gali būti, kad patinkančių dainų ir neberastų. Tokiu atveju tyrimo rezultatai jau nebūtų objektyvūs, todėl respondentas pats sprendžia kada nori klausytis muzikos. Siekiant, kad tyrimo rezultatai būtų kuo objektyvesni, respondentams duodama klausytis muzikos su atsitiktinai parenkamu muzikos grotuvu, t.y. vienas respondentas klausosi muzikos tokia muzikos grotuvų išdėstymo tvarka:

1. „AIMP“, 2. „PRO“, 3. „WMP“, o kitas: 1. „WMP“, 2. „AIMP“, 3. „PRO“ ir t.t. Kadangi klausantis muzikos naudojamas maišymo (*angl.* shuffle) režimas, tai muzikos klausymo sesijos pradedamos atsitiktinai parenkama daina iš duoto 100 muzikos kūrinių grojaraščio.

### 3.2 Tyrimo rezultatai

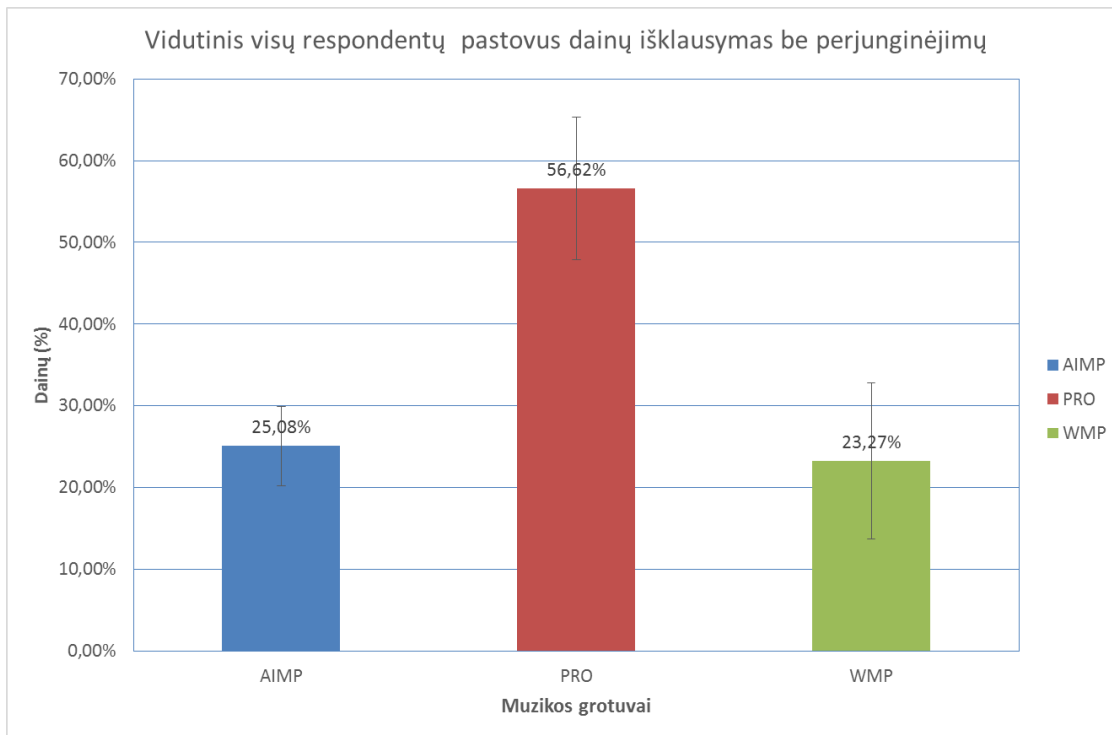
Grafike (žr. 17 pav. ) pateikta vieno respondento muzikos klausymo sesijos su visais tyrime naudojamais muzikos grotuvais. Ji parodo kiek procentų išklausė kiekvienos dainos, ko pasekoje matosi kiek dainų iki galo buvo išklausyta nuosekliai.



17 pav. Dainų klausymo sesijos su visais muzikos grotuvais

X ašyje vaizduojamas dainos klausymo eilės numeris, Y ašyje vaizduojamas dainos išklausymas procentais. Iš pateiktos diagramos matyti, kad su „AIMP“ ir „WMP“ muzikos grotuvais, išklausius vieną ar kelias dainas iki galo, sekančios viena ar kelios dainos neįklausomos iki galo. Tai parodo, kad gan trumpą laiko tarpą respondentas klauso dainas pastoviai iki galo. Tuo tarpu klausantis dainų naudojant prototipą, aiškiai matoma, kad tam tikrą laiko tarpą respondentas išklausoma dainas

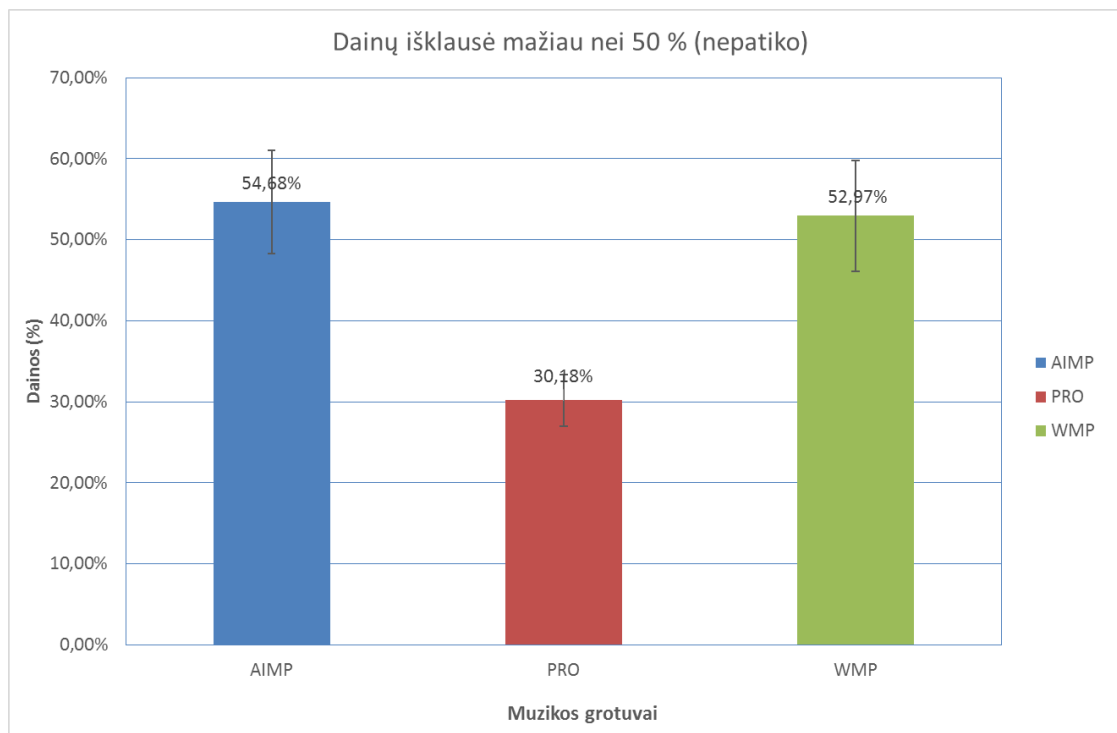
pastoviai iki galo jų neperjunginėdamas. 18 pav. paveikslėlyje pateikta, kiek vidutiniškai dainų yra pastoviai išklausoma iki galo jų neperjungiant, su kiekvienu muzikos grotuvu.



18 pav. Vidutinis visų respondentų pastovus dainų išklausymas be perjunginėjimų

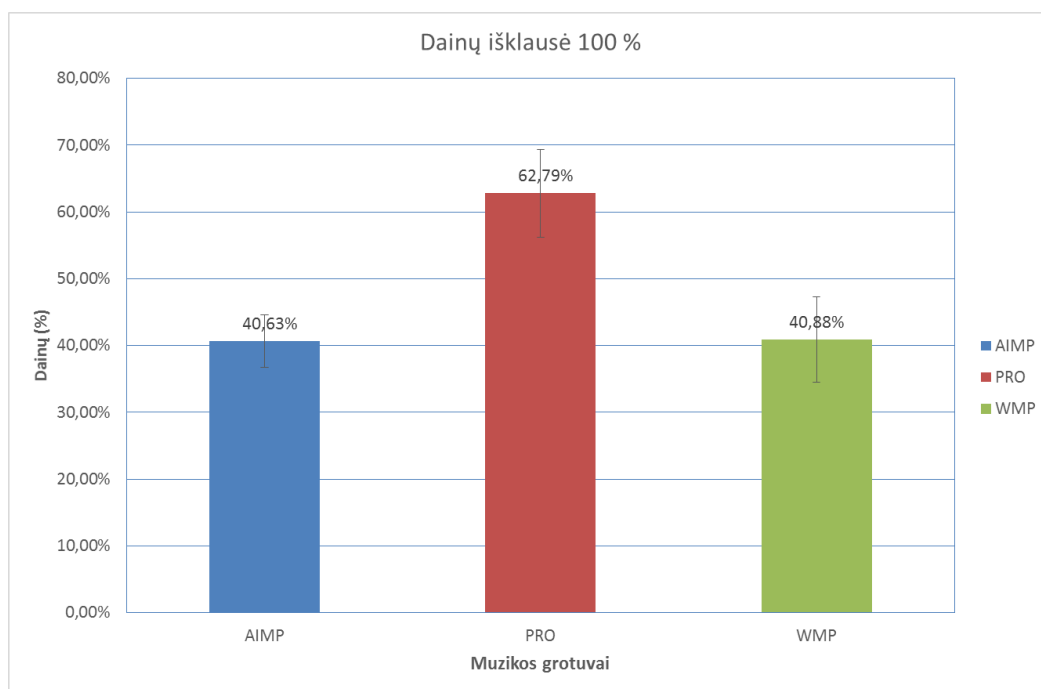
Iš pateikto grafiko matyti, kad su prototipu vidutiniškai pastoviai išklausoma 56,62% dainų jų neperjunginėjant, tuo tarpu su „AIMP“ ir „WMP“ muzikos grotuvais, vidutiniškai pastoviai išklausoma daug mažiau dainų (25,08% ir 23,27%) jų neperjunginėjant.

Tyrimas parodė, kad mažiausiai nepatikusių dainų, kurios buvo išklausytos mažiau nei 50%, buvo naudojant siūlomo algoritmo prototipą (30,18%) (žr. 19 pav. ), o naudojant „AIMP“ ir „WMP“ muzikos grotuvus, nepatikusių dainų buvo parenkama daug daugiau (54,68% ir 52,97%).



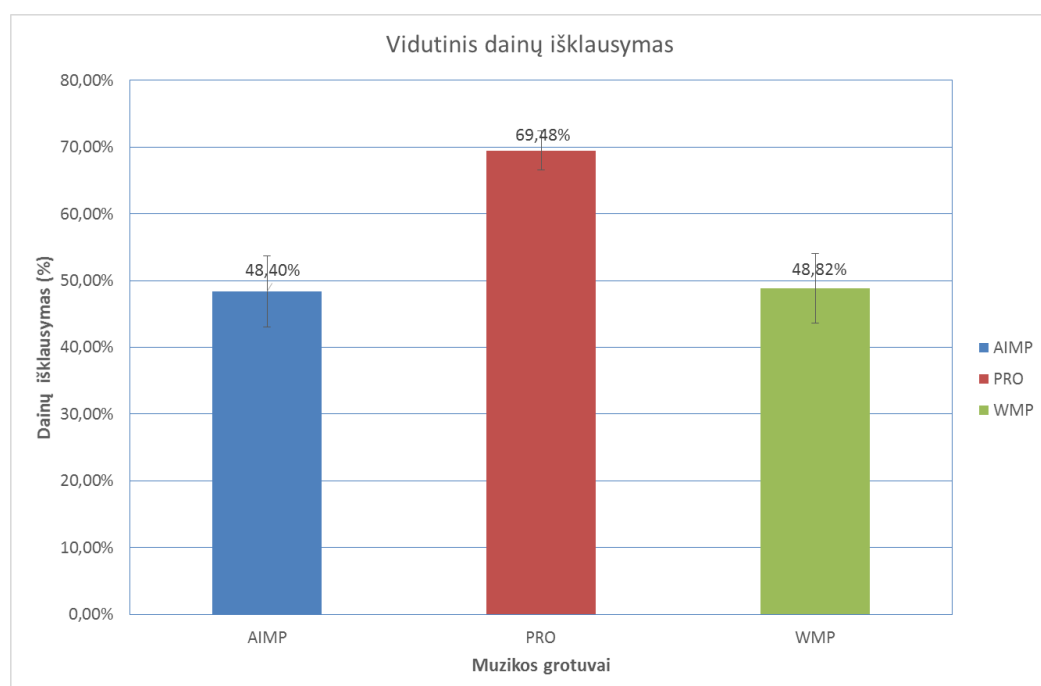
19 pav. Dainų išklausymas mažiau nei 50% dainos

Taip pat paaiškėjo, kad daugiausiai pilnai išklausytų (nuo pradžios iki galo) dainų buvo išklausa su siūlomo algoritmo prototipu (62,79%). Kaip iš 20 pav. paveikslėlyje pateikto grafiko matoma, kad su „AIMP“ ir „WMP“ muzikos grotuvais pilnai išklausytų dainų buvo mažiau (40,63% ir 40,88%) nei su prototipu. Iš to galima teigti, kad daugiau respondentui patinkančių dainų buvo pasiūlyta naudojant siūlomo algoritmo prototipą.



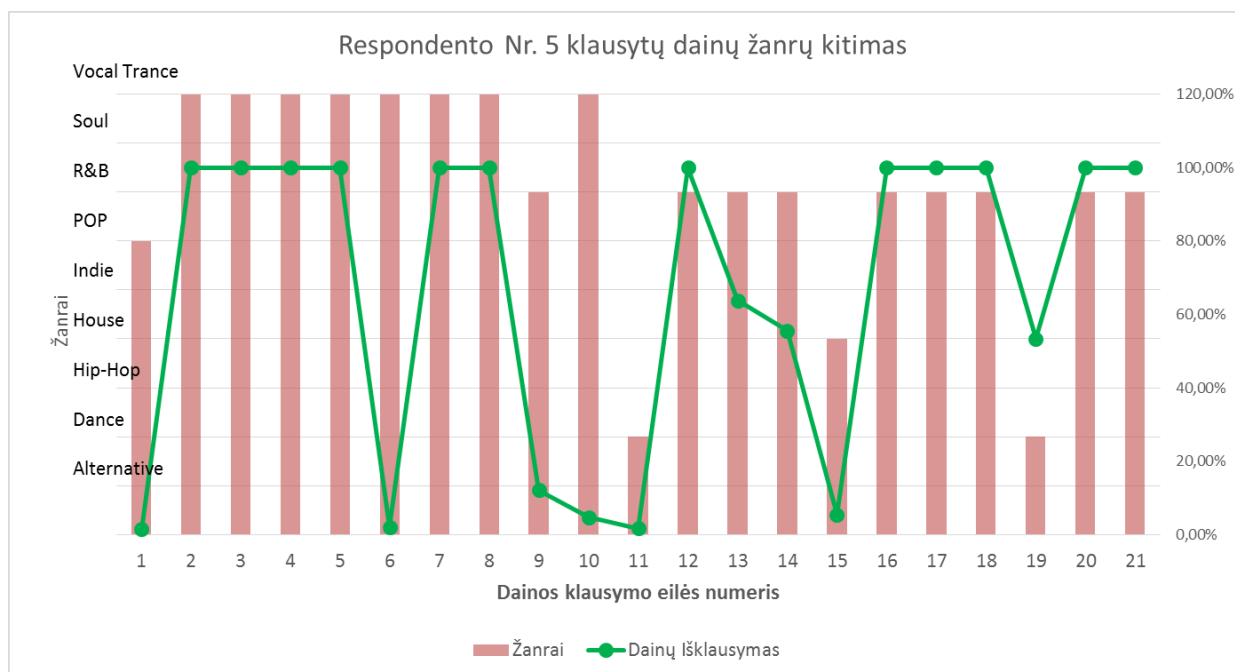
20 pav. Pilnas dainų išklausymas

Iš pateikto grafiko (žr. 21 pav. ) paaiškėjo, kad respondentai vidutiniškai daugiausiai dainų išklauso su siūlomo algoritmo prototipu (69,48%), tuo tarpu su kitais muzikos grotuvais vidutiniškai išklauso mažiau dainų (48,40% ir 48,82%). Iš to galima teigti, kad siūlomo algoritmo prototipas klausytojui parenka labiau patinkančias grojaraščio dainas nei kiti muzikos grotuvai, todėl jas rečiau perjunginėja ir dažniau išklauso iki galo.

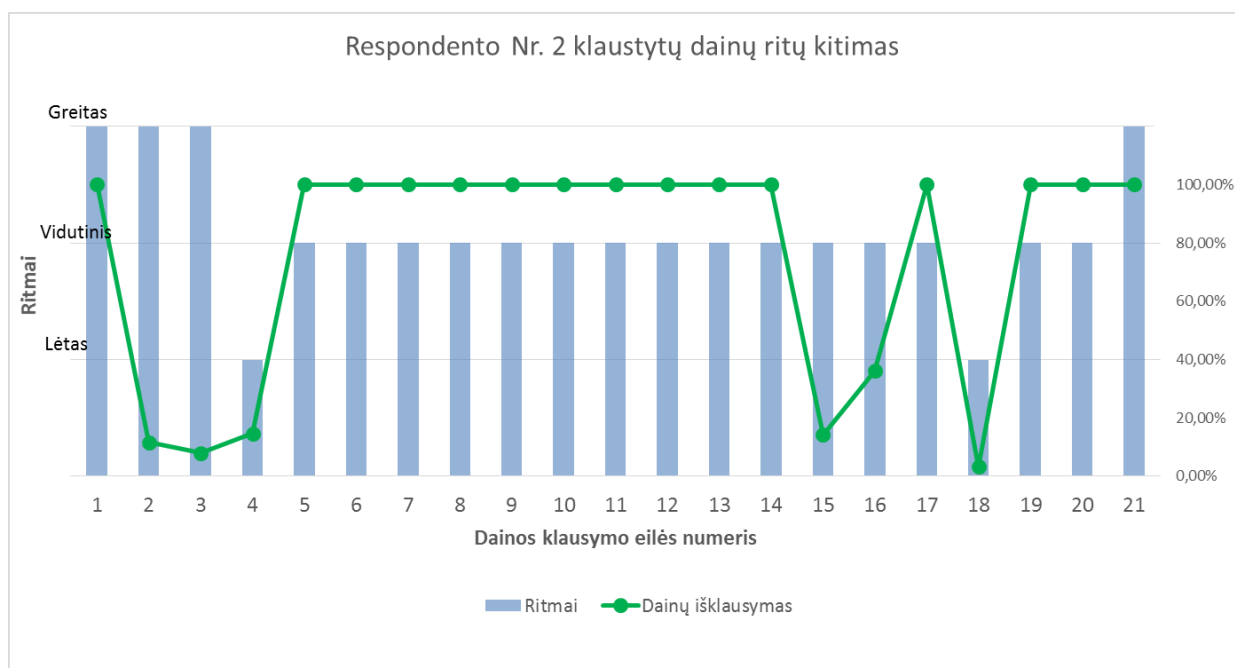


21 pav. Vidutinis dainų išklausymas pagal muzikos grotuvą

Siekiant nustatyti, kurie dainų meta-duomenys turėjo daugiausiai įtakos siūlomo algoritmo grojaraščio sudarymui, buvo atliktas tyrimas, kurio metu fiksuojamas meta-duomenų reikšmių kitimas respondentams klausantis muzikos įrašų su siūlomo algoritmo prototipu.



22 pav. Žanrų kitimas



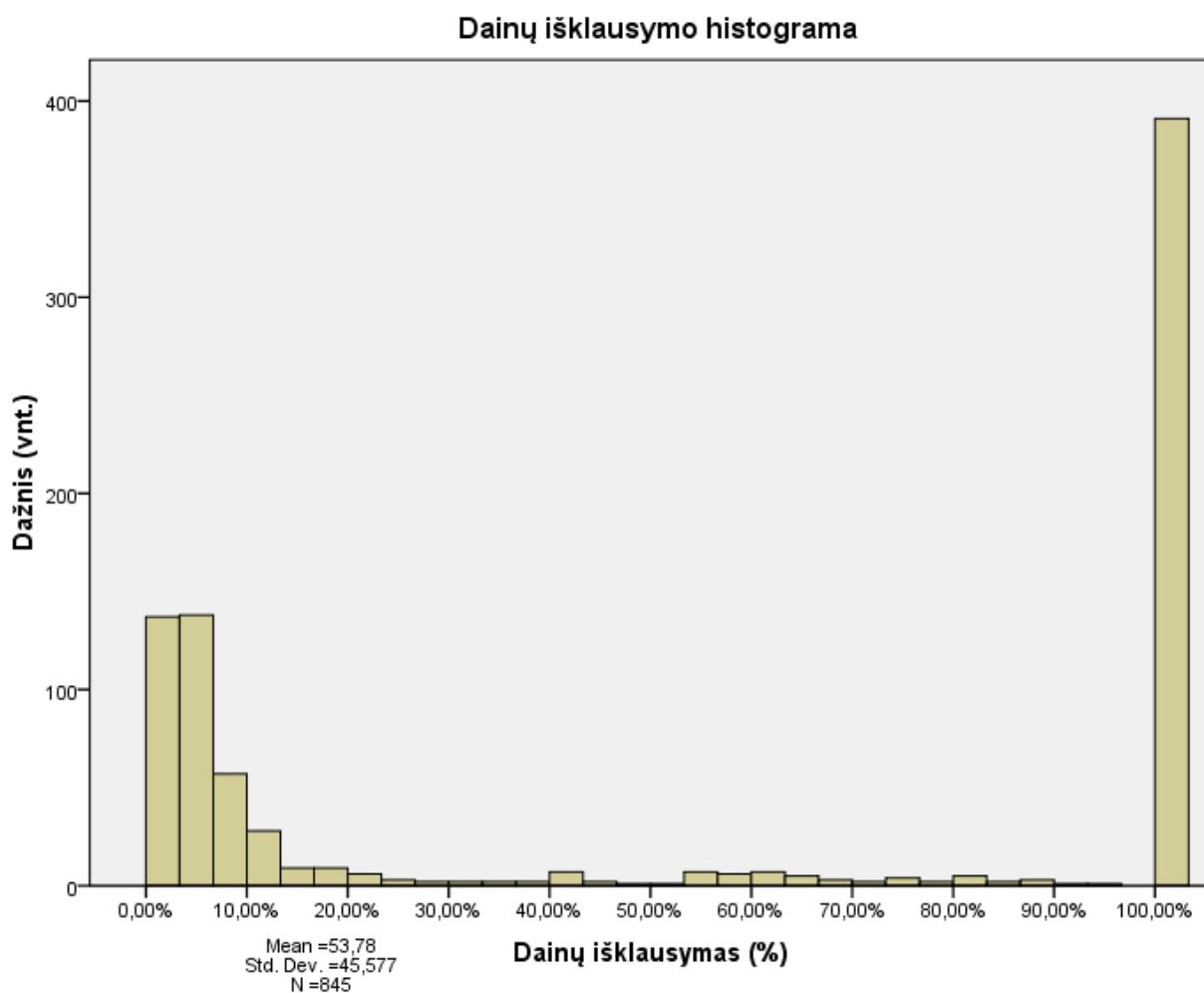
23 pav. Ritmų kitimas

Tyrimas atskleidė, kad vieniems respondentams didelę įtaką grojaraščio sudarymui turėjo dainos žanras (žr. 22 pav. ), o kitiems respondentams – dainos ritmas (žr. 23 pav. ). Iš pateiktų grafikų aiškiai matomas pastovumas tiek dainos ritmo, tiek jos žanro. Vienam iš respondentų beveik visas grojaraštis sudarytas iš vidutinio ritmo dainų, o kitam iš Vocal Trance bei R&B žanro dainų. Taip pat matomos ir sąsajos su dainos išklausimu. Nusistovėjęs pastoviam ritmui ar žanrui dažniausiai dainos išklausomos iki galo. Iš to galima teigti, kad siūlomo algoritmo grojaraščių sudarymui didelę įtaką turi ir klausomos dainos ritmas ir jos žanras, tuo tarpu kiti meta-duomenys, kaip dainos metai bei atlikėjas, neturėjo



didelės įtakos grojaraščio sudarymui, nes reikšmės buvo labai išsibarsčiusios ir nebuvo matoma jokio pastovumo tarp jų.

Taip pat buvo atliktas tyrimas siekiant išsiaiškinti kada vartotojai dažniausiai perjunginėja nepatikusias dainas. Ar jas perjunginėja iš karto tik pradėjus klausytis dainos ar išsiklausius ilgiau. Iš pateikto grafiko (žr. 24 pav. ) pastebima, kad daugiausiai dainų perjunginėjama dainą išklaudus iki ~13%. Taip pat matoma, kad daug dainų yra išklaudoma iki galo, ir yra tik dvi vietos, kur beveik dainos yra neperjunginėjamos., t.y. ties maždaug 50% ir 98%. Tad norint palikti vietas mažiems dainos prasukimams ir pan. naudojamas tas viduriukas ties 50%, dainų dalinimui į dvi kategorijas – patiko ir nepatiko. Tai patvirtina, kad tyrime pasirinkta dainų traktavimo į patinkančias ir nepatinkančias, 50% riba yra panaši į vartotojų elgseną.



*24 pav. Dainų išklaudymo histograma*

Atlikus siūlomo pseudo atsitiktinį grojaraščių sudarymo, atsižvelgiantį į muzikos klausymo istoriją, algoritmo tyrimą ir apžvelgus gautus tyrimo rezultatus, galime sakyti, kad naujas algoritmas turi pranašumą prieš kitus egzistuojančius sprendimus, nes naudojant minėtą algoritmą, daug mažiau dainų buvo prasukinėjama, daug daugiau dainų buvo pastoviai išklaudyta iki galo, mažiausiai buvo parenkama nepatikusių dainų nei su kitais sprendimais, o tai reiškia, kad naujasis algoritmas geriau prisitaiko prie klausytojo muzikos klausymo poreikių, nes klausytojui vis rečiau reikia perjunginėti nepatikusias dainas, ko pasekoje yra labiau tenkinami klausytojų poreikiai.

### 3.3 Pasiūlyto algoritmo našumo testavimas

Norint taikyti pasiūlytą algoritmą, svarbu išsiaiškinti jo veikimo spartą bei kokie kriterijai tai labiausiai įtakoja. Našumo testavimo metu testuojamos pagrindinės pasiūlyto algoritmo funkcijos: savybių svarbos koeficientų sąrašo generavimo našumas, muzikos kūrinų koeficientų sąrašo generavimo našumas, sekančios grojaraščio dainos parinkimo našumas. Taip pat testuojama ir RAM naudojimas parinkinėjant sekančią grojaraščio dainą. Našumas testuojamas į programinį kodą kiekvienos testuojamos funkcijos pradžioje ir pabaigoje, įterpiant laiko funkcijas, kurios fiksuoja, kada funkcija prasidėjo ir kada baigėsi. Tada žiūrimas funkcijos pradžios ir pabaigos laikų skirtumas ir sužinoma funkcijos įvykdymo trukmė. Siekiant sužinoti RAM naudojimą, naudojamas Java programinis kodas.

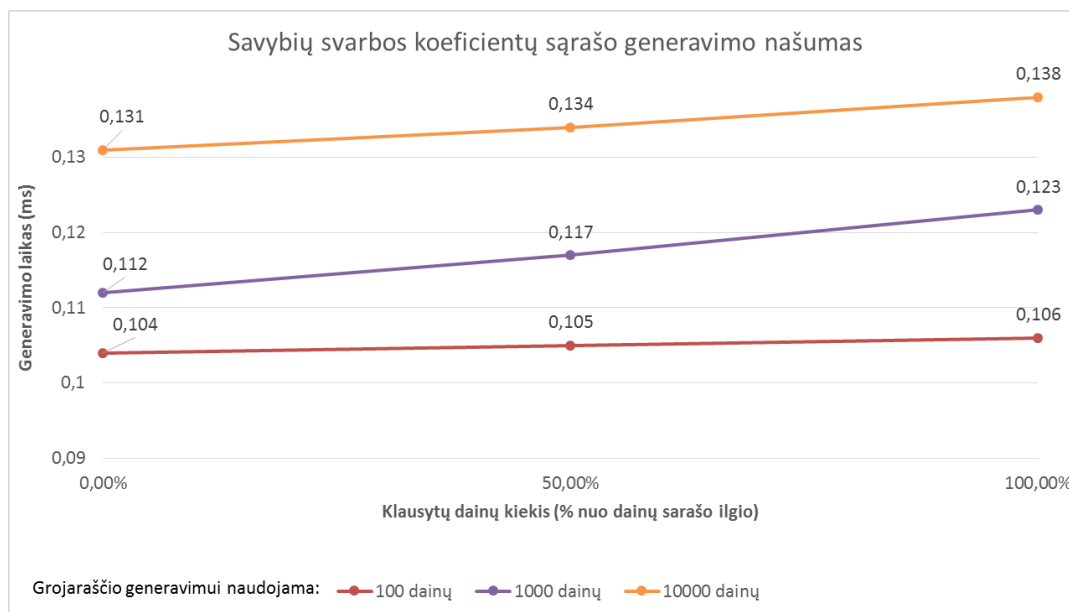
Svarbu pažymėti, kad pagrindiniai kintamieji testuojant našumą – grojaraščio generavimui naudojamas dainų kiekis (100 dainų, 1000 dainų ir 10000 dainų) ir klausytų dainų kiekis (0% grojaraščio generavimui naudojamų dainų kiekio, 50% ir 100% grojaraščio generavimui naudojamų dainų kiekio). Pirmiausia testuojama pasiūlyto algoritmo funkcija su 100 muzikos įrašų ir 0% klausytų dainų kiekiu. Paskui didinamas klausytų dainų kiekis atitinkamai iki 50% ir 100%. Po to didinamas muzikos įrašų kiekis iki 1000 bei 10000, o klausytų dainų istorija vėl didinama nuo 0% iki 50% ir 100% ir t.t. Tokiu būdu ištestuojamos visos pagrindinės pasiūlyto algoritmo funkcijos.

#### 3.3.1 Testavimo resursai

- Procesorius – Intel Core 2 Duo T6500 2.1 GHz;
- Operatyvioji atmintis – 4,00 GB;
- Vaizdo plokštė – ATI Mobility Radeon HD 4650 1 GB;
- Operacinė sistema – Microsoft Windows 7 Ultimate SP1 x64;
- Java – 1.70.40.

#### 3.3.2 Testavimo rezultatai

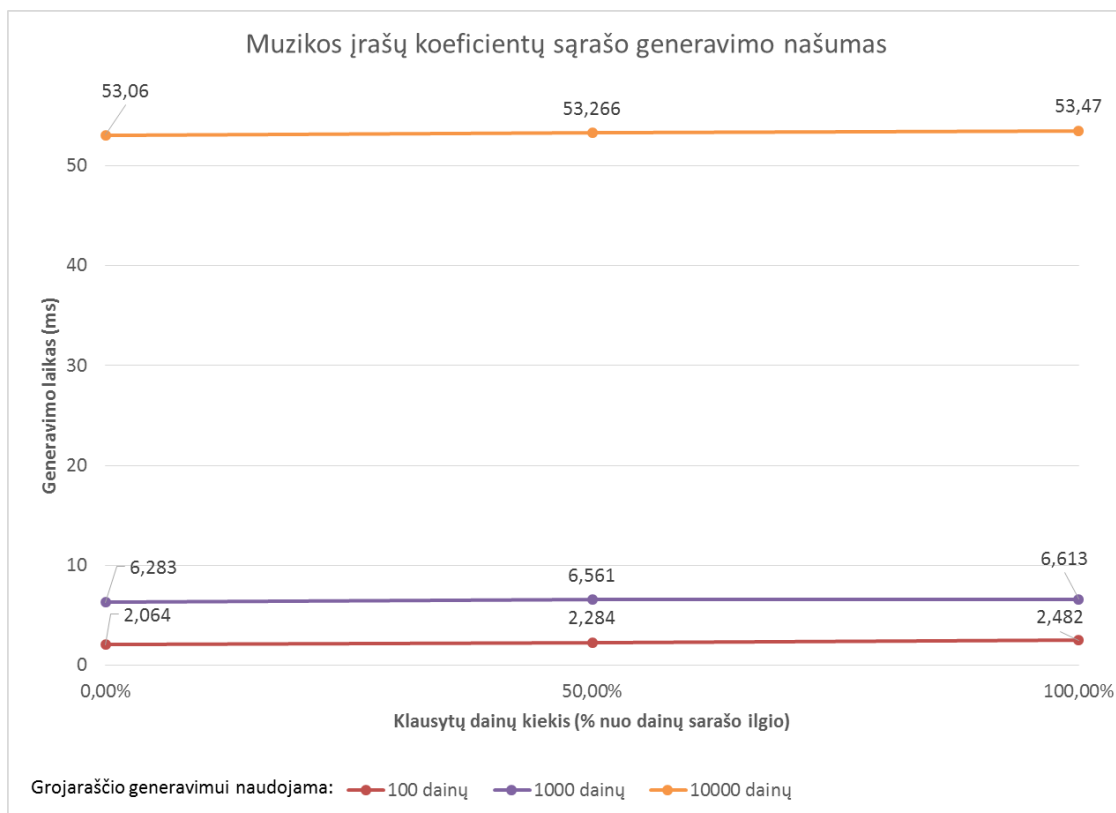
Grafike (žr. 25 pav.) pateiktas savybių svarbos koeficientų sąrašo generavimo našumas.



25 pav. Savybių svarbos koeficientų sąrašo generavimo našumas

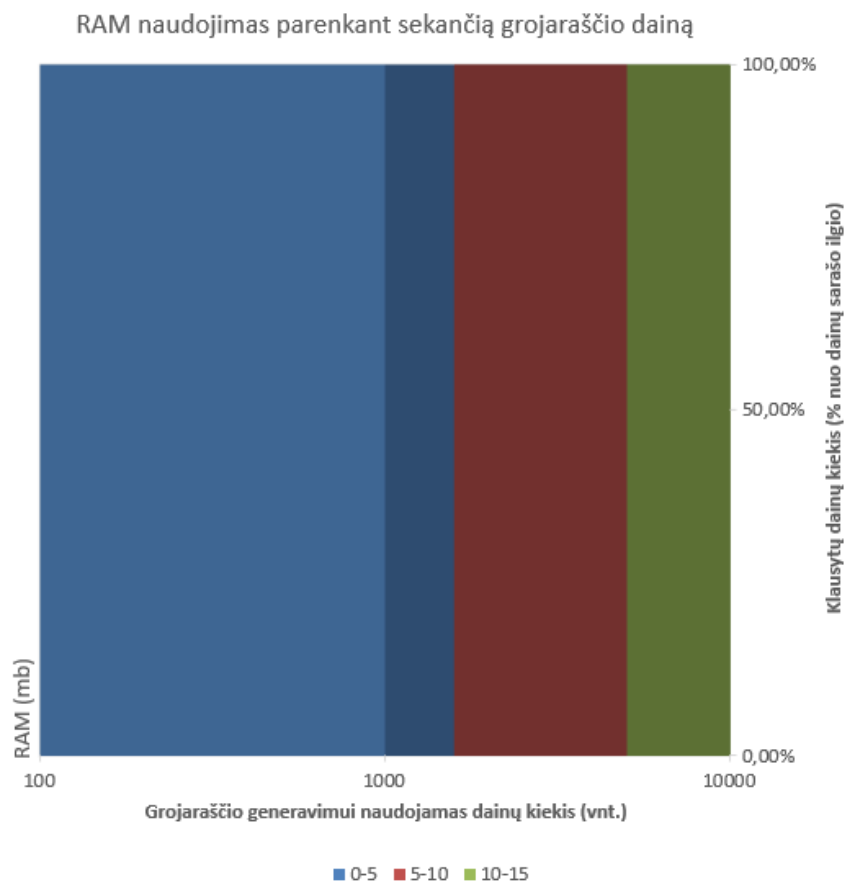
X ašyje vaizduojamas klausytų dainų kiekis (% nuo dainų sąrašo ilgio), Y ašyje vaizduojamas savybių svarbos koeficientų sąrašo generavimo laikas (ms). Iš pateikto grafiko matyti, kad šio sąrašo generavimui didesnę įtaką turi grojaraščio generavimui naudojamų dainų kiekis, negu klausytų dainų kiekis. Tačiau abu kriterijai nedaug įtakoja generavimo laiką, nes didinant grojaraščio generavimui naudojamų dainų kiekį ir klausytų dainų kiekį, generavimo laikas pailgėjo tik keliolika mili sekundžių.

Atlikus muzikos įrašų koeficientų sąrašo generavimo našumą (žr. 26 pav. ), paaiškėjo, kad nemažą įtaką sąrašo generavimui turi grojaraščio generavimui naudojamų muzikos įrašų kiekis. Tuo tarpu klausytų dainų kiekis beveik įtakos neturi, nes nuo jo sąrašo generavimo laikas užtrunka tik keliomis mili sekundėmis ilgiau, o pvz., padidinus grojaraščio generavimui naudojamų dainų kiekį nuo 1000 iki 10000 dainų, generavimo laikas ženkliai skiriasi (nuo 6,283 iki 53,06 ms). Taip yra todėl, kad muzikos įrašų koeficientų sąrašo generavimui visoms dainoms yra suskaičiuojamas koeficientas priklausomai, kaip buvo klausoma daina. Todėl kuo didenis dainų kiekis, tuo ilgiau trunka visoms dainoms suskaičiuoti koeficientus.

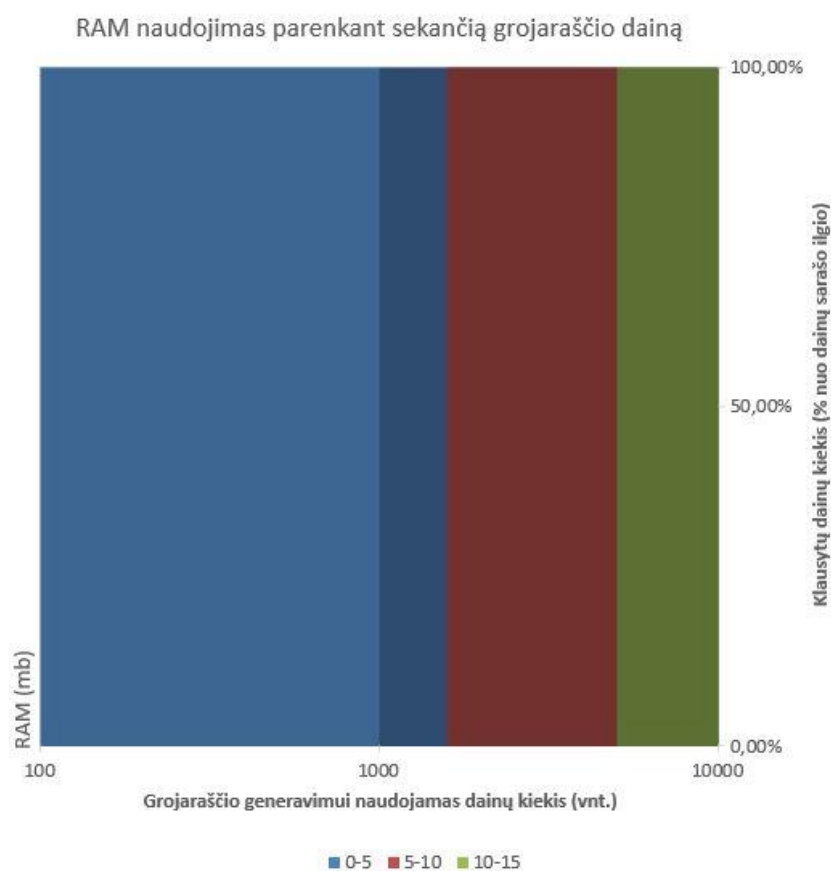


26 pav. Muzikos įrašų koeficientų sąrašo generavimo našumas

Grafike (žr. 27 pav. ) pateiktas sekančios dainos parinkimo našumas. Iš pateikto grafiko matyti, kad sekančios dainos parinkimui turi įtakos ir grojaraščio generavimui naudojamų muzikos įrašų kiekis ir klausytų dainų kiekis. Tačiau klausytų dainų kiekis labiau įtakoja sekančios dainos parinkimą nei grojaraščio generavimui naudojamų dainų kiekis. Pvz., esant grojaraštyje 10000 dainų, didinant klausytų dainų kiekį, sekančios dainos parinkimas užtrunka nuo ~50 ms iki ~250ms. Taip yra todėl, kad sekančios dainos parinkimui yra naudojama viena iš funkcijų, kuri atsakinga už tai, kad sekanti daina nebūtų parinkta iš jau klausytų dainų, esančių klausytų dainų istorijoje. Todėl kuo didesnis klausytų dainų kiekis, tuo ilgiau užtrunkama atrinkti, kurios dainos jau buvo klausytos.



27 pav. Sekančios dainos parinkimo našumas



28 pav. RAM naudojimas parenkant sekančią grojaraščio dainą

Atlikus sekančios dainos parinkimo našumo testavimą, paaiškėjo, kad operatyvinės atminties buvo naudojama nedaug (iki 15 MB) (žr. 28 pav. ). Taip pat svarbu pažymėti, kad klausytos dainos

neturėjo įtakos RAM naudojimui. RAM naudojimas priklausė tik nuo grojaraščio generavimui naudojamų dainų kiekio.

Apibendrinant galima teigti, kad pseudo atsitiktinis grojaraščių sudarymo algoritmas yra tinkamas taikyti įvairiose sistemose bei įrenginiuose, kadangi atlikus šio algoritmo našumo testavimą, matyti, kad algoritmas veikia sparčiai net ir su 10000 muzikos įrašų grojaraščiu, o operatyvinės atminties naudoja ganėtinai mažai.

## IŠVADOS

1. Išanalizavus egzistuojančias atsitiktinių grojaraščių sudarymo metodų kategorijas ir konkrečius pavyzdžius pastebėta, kad hibridiniai grojaraščių sudarymo metodai leidžia labiau išnaudoti skirtingų metodų kategorijų privalumus. Todėl siūlant naują muzikos grojaraščių sudarymo algoritmą reikėtų taikyti kelių tipų metodų kombinaciją ir taip padidinti naujo metodo naudą vartotojų poreikių tenkinimui.
2. Sukurtas pseudo atsitiktinis grojaraščių sudarymo algoritmas, kuris atrinkdamas sekančią grojamą grojaraščio dainą, remiasi ne tik vartotojų atliekamais veiksmais, kaip dainos prasukimas, pilnas bei nepilnas dainos išklausymas, bet ir vartotojui grotų muzikos kūrinų sąrašą. Tai leidžia sukurtam algoritmui adaptuotis prie skirtingų ir kintančių vartotojų muzikinių skonių.
3. Šiuo metu egzistuojantys muzikos grojaraščių algoritmai nėra pilnai detalizuoti moksliniuose darbuose ar internete, bet yra įgyvendinami esamuose muzikos grotuvuose. Šiame darbe buvo sukurtas muzikos grotuvo prototipas, kuriame realizuotas naujai siūlomas pseudo atsitiktinis grojaraščių generavimo algoritmas. Tai leidžia juodos dėžės principu lyginti siūlomą algoritmą su kituose muzikos grotuvuose taikomais algoritmais.
4. Išanalizavus tyrimo rezultatus paaiškėjo, kad pasiūlytas pseudo atsitiktinis grojaraščių sudarymo, atsižvelgiantis į muzikos klausymo istoriją, algoritmo prototipas yra ženkliai pranašesnis už kitus tirtus egzistuojančius sprendimus – naudojant sukurtą prototipą, dainos vidutiniškai buvo išklausomos 69,48%, tuo tarpu su „AIMP“ grotuvu vidutiniškai išklausoma tik 48,40% dainos, o su „WMP“ grotuvu – 48,82% dainos. Tai leidžia teigti, kad sukurtas algoritmas ir jo pagrindu realizuotas prototipas tinkamai geba adaptuotis prie vartotojo muzikinio skonio. Taip pat, atlikus šio algoritmo našumo testavimą, matyti, kad algoritmas veikia sparčiai net ir su 10000 muzikos įrašų grojaraščiu, o operatyvinės atminties naudoja ganėtinai mažai, todėl galima teigti, kad pseudo atsitiktinis grojaraščių sudarymo algoritmas yra tinkamas taikyti įvairiose sistemose bei įrenginiuose.

## LITERATŪRA

1. Apple introduces iPod Shuffle. [interaktyvus], [žiūrėta 2015-04-26]. Prieiga per internetą: <<http://www.apple.com/pr/library/2005/jan/11shuffle.htm>>
2. Bernhardsson E. Implementing a scalable music recommender system (2009). [interaktyvus], [žiūrėta 2015-04-26]. Prieiga per internetą: <[https://www.nada.kth.se/utbildning/grukth/exjobb/rapportlistor/2009/rapporter09/bernhardsson\\_erik\\_09071.pdf](https://www.nada.kth.se/utbildning/grukth/exjobb/rapportlistor/2009/rapporter09/bernhardsson_erik_09071.pdf)>
3. Burke R. Hybrid Recommender Systems: Survey and Experiments. User Modeling and User-Adapted Interaction (2002). [interaktyvus], [žiūrėta 2015-04-26]. Prieiga per internetą: <[http://kslab.kaist.ac.kr/kse643/Burke\\_umuai02.pdf](http://kslab.kaist.ac.kr/kse643/Burke_umuai02.pdf)>
4. Cai, R., Zhang, C., Wang, C., Zhang, L., Ma, W. Musicsense: Contextual music recommendation using emotional allocation modeling (2007). [interaktyvus], [žiūrėta 2015-04-26]. Prieiga per internetą: <<http://www.msr-waypoint.net/en-us/um/people/leizhang/paper/acmmm07-cai-poster.pdf>>
5. Cano P., Koppenberger M., Wack N. Content-based Music Audio Recommendation (2005). [interaktyvus], [žiūrėta 2015-04-26]. Prieiga per internetą: <<http://s3.amazonaws.com/academia.edu.documents/33739575/a4924b-ACMMM05-Cano-Koppenberger-Wack.pdf?AWSAccessKeyId=AKIAJ56TQJRTWSMTNPEA&Expires=1430000679&Signature=ATEoYG9tcmt5g%2FVNok5kjuMKw0Q%3D>>
6. Cano, P., Koppenberger, M., Wack, N. An industrial-strength content-based music recommendation system (2005). [interaktyvus], [žiūrėta 2015-04-26]. Prieiga per internetą: <<http://www.mtg.upf.edu/files/publications/3ac0d3-SIGIR05-pcano.pdf>>
7. Cohen, W. W. ir Fan, W. Web-collaborative filtering: Recommending music by crawling the web (2000). [interaktyvus], [žiūrėta 2015-04-26]. Prieiga per internetą: <<http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=CA8021552DA22FF1C7EFBF666F04F7B5?doi=10.1.1.218.8476&rep=rep1&type=pdf>>
8. Flexer, A., Schnitzer, D., Gasser, M., G., W. Playlist generation using start and end songs (2008). [interaktyvus], [žiūrėta 2015-04-26]. Prieiga per internetą: <<http://www.ofai.at/cgi-bin/get-tr?download=1&paper=oefai-tr-2008-03.pdf>>
9. Hu Y. Ogihara M. Nexttone Player: A Music Recommendation System Based on User Behavior (2011). [interaktyvus], [žiūrėta 2015-04-26]. Prieiga per internetą: <<http://ismir2011.ismir.net/papers/PS1-11.pdf>>
10. J. Stephen Downie. Music Information Retrieval. (2003). [interaktyvus], [žiūrėta 2015-04-26]. Prieiga per internetą: <[http://www.music.mcgill.ca/~ich/classes/mumt611\\_08/downie\\_mir\\_arist37.pdf](http://www.music.mcgill.ca/~ich/classes/mumt611_08/downie_mir_arist37.pdf)>

11. Kazuyoshi Y., Masataka G., Kazunori K., Tetsuya O., Hiroshi O. Hybrid Collaborative and Content-based Music Recommendation Using Probabilistic Model with Latent User Preferences. (2006). [interaktyvus], [žiūrėta 2015-04-26]. Prieiga per internetą: <[http://ismir2006.ismir.net/PAPERS/ISMIR0647\\_Paper.pdf](http://ismir2006.ismir.net/PAPERS/ISMIR0647_Paper.pdf)>
12. Kazuyoshi Y., Masataka G., Kazunori K., Tetsuya O., Hiroshi O. Improving Efficiency and Scalability of Model-Based Music Recommender System Based on Incremental Training (2007). [interaktyvus], [žiūrėta 2015-04-26]. Prieiga per internetą: <<http://130.54.20.150/~yoshii/papers/ismir-2007-yoshii.pdf>>
13. L. Barrington, R. Oda, G. Lanckriet. Smarter than genius? human evaluation of music recommender systems (2009). [interaktyvus], [žiūrėta 2015-04-26]. Prieiga per internetą: <[http://www.cs.princeton.edu/~roda/pubs/Barrington-Oda-Lanckriet\\_ISMIR09.pdf](http://www.cs.princeton.edu/~roda/pubs/Barrington-Oda-Lanckriet_ISMIR09.pdf)>
14. Lamere P. Social Tagging and Music Information Retrieval (2008). [interaktyvus], [žiūrėta 2015-04-26]. Prieiga per internetą: <[https://musicmachinery.files.wordpress.com/2009/05/v2\\_jnmr\\_social\\_tagging\\_intro.pdf](https://musicmachinery.files.wordpress.com/2009/05/v2_jnmr_social_tagging_intro.pdf)>
15. M. García Quiñones. Listening in shuffle mode (2007). [interaktyvus], [žiūrėta 2015-04-26]. Prieiga per internetą: <[http://www.jstor.org/stable/30043759?seq=1#page\\_scan\\_tab\\_contents](http://www.jstor.org/stable/30043759?seq=1#page_scan_tab_contents)>
16. Pampalk, E., Pohle, T., Widmer, G. Dynamicplaylist generation based on skipping behavior (2005). [interaktyvus], [žiūrėta 2015-04-26]. Prieiga per internetą: <<http://www.ofai.at/cgi-bin/get-tr?paper=ofai-tr-2005-23.pdf>>
17. Pandora [interaktyvus], [žiūrėta 2015-04-26]. Prieiga per internetą: <<http://www.pandora.com>>
18. Pandora statistics 2015 [interaktyvus], [žiūrėta 2015-04-26]. Prieiga per internetą: <<http://expandedramblings.com/index.php/pandora-statistics/>>
19. Peter J. Rentfrow, Samuel D. Gosling. The Do Re Mi's of Everyday Life: The structure and personality correlates of music preferences (2003). [interaktyvus], [žiūrėta 2015-04-26]. Prieiga per internetą: <<http://www.stat.ufl.edu/STA%204702/Spring%2007/Rentfrow&Gosling.pdf>>
20. Ragno, R., Burges, C., Herley, C. Inferring similarity between music objects with application to playlist generation (2005). [interaktyvus], [žiūrėta 2015-04-26]. Prieiga per internetą: <<http://research.microsoft.com/pubs/69373/mir2005.pdf>>
21. Random playlists not random. [interaktyvus], [žiūrėta 2015-04-26]. Prieiga per internetą: <<https://forum.videolan.org/viewtopic.php?t=90559>>
22. Rob van Gulik, Vignoli F. Visual Playlist Generation on the Artist Map (2005) [interaktyvus], [žiūrėta 2015-04-26]. Prieiga per internetą: <<http://www.ismir2005.ismir.net/proceedings/2011.pdf>>
23. Shao, B., Wang, D., Li, T., Ogihara, M. Musicrecommndation based on acoustic features and user access patterns (2009). [interaktyvus], [žiūrėta 2015-04-26]. Prieiga per internetą: <<https://users.cs.fiu.edu/~taoli/pub/music-TASLP.pdf>>



24. Shoham Y. Balabannovic M. Content-Based, Collaborative Recommendation. Communications of the (1997). [interaktyvus], [žiūrėta 2015-04-26]. Prieiga per internetą: <[https://www.ischool.utexas.edu/~i385q/readings/Balabanovic\\_Shoham-1997-Fab.pdf](https://www.ischool.utexas.edu/~i385q/readings/Balabanovic_Shoham-1997-Fab.pdf)>
25. Symeonidis P., Ruxanda M., Nanopoulos A., Manolopoulos Y. Ternary Semantic Analysis of Social Tags for Personalized Music Recommendation (2008). [interaktyvus], [žiūrėta 2015-04-26]. Prieiga per internetą: <<http://delab.csd.auth.gr/papers/ISMIR08srnm.pdf>>
26. Su, J., Yeh, H. Music recommendation using content and context information mining (2010). [interaktyvus], [žiūrėta 2015-04-26]. Prieiga per internetą: [http://vigliensoni.com/BUP/DropBox\\_10\\_12\\_09/1\\_PHD\\_APPLICATION/Proposal/su10EEEE.pdfd/su10EEEE.pdf](http://vigliensoni.com/BUP/DropBox_10_12_09/1_PHD_APPLICATION/Proposal/su10EEEE.pdfd/su10EEEE.pdf)
27. Tempo [interaktyvus], [žiūrėta 2015-04-26]. Prieiga per internetą: <<http://en.wikipedia.org/wiki/Tempo>>
28. Wang D., Li T., Ogihara M. Tags Better Than Audio Features? The Effect of Joint use of Tags and Audio Content Features for Artistic Style Clustering (2010). [interaktyvus], [žiūrėta 2015-04-26]. Prieiga per internetą: <<http://www.ismir2010.ismir.net/proceedings/ismir2010-12.pdf>>
29. Wood R., H. Interview Insider: How to Get Hired at Spotify (2015). [interaktyvus], [žiūrėta 2015-04-26]. Prieiga per internetą: <<http://www.cosmopolitan.com/career/interviews/a37865/interview-insider-spotify-career-jobs>>