Izrada Android mobilne aplikacije za ugađanje tona akustične gitare

Vukadin, Domagoj

Undergraduate thesis / Završni rad

2020

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:200:508693

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-04-19**

Repository / Repozitorij:

Faculty of Electrical Engineering, Computer Science and Information Technology Osijek

SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKUFAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA

Sveučilišni studij

IZRADA ANDROID APLIKACIJE ZA UGAĐANJE TONA AKUSTIČNE GITARE

Završni rad

Domagoj Vukadin

Osijek, 2020.

FERIT
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA OSIJEK

Obrazac Z1P - Obrazac za ocjenu završnog rada na preddiplomskom sveučilišnom studiju

Osijek, 14.09.2020.

Odboru za završne i diplomske ispite

Prijedlog ocjene završnog rada na preddiplomskom sveučilišnom studiju

Ime i prezime studenta:	Domagoj Vukadin
Studij, smjer:	Preddiplomski sveučilišni studij Računarstvo
Mat. br. studenta, godina upisa:	R4006, 24.09.2019.
OIB studenta:	01537520837
Mentor:	Izv. prof. dr. sc. Damir Blažević
Sumentor:	
Sumentor iz tvrtke:	
Naslov završnog rada:	Izrada Android mobilne aplikacije za ugađanje tona akustične gitare
Znanstvena grana rada:	**Procesno računarstvo (zn. polje računarstvo)**
Predložena ocjena završnog rada:	Izvrstan (5)
Kratko obrazloženje ocjene prema Kriterijima za ocjenjivanje završnih i diplomskih radova:	Primjena znanja stečenih na fakultetu: 3 bod/boda Postignuti rezultati u odnosu na složenost zadatka: 3 bod/boda Jasnoća pismenog izražavanja: 3 bod/boda Razina samostalnosti: 3 razina
Datum prijedloga ocjene mentora:	14.09.2020.
Datum potvrde ocjene Odbora:	23.09.2020.

Potpis mentora za predaju konačne verzije rada u Studentsku službu pri završetku studija:	Potpis:
	Datum:

FERIT
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA
I INFORMACIJSKIH TEHNOLOGIJA **OSIJEK**

IZJAVA O ORIGINALNOSTI RADA

Osijek, 28.09.2020.

Ime i prezime studenta:	Domagoj Vukadin
Studij:	Preddiplomski sveučilišni studij Računarstvo
Mat. br. studenta, godina upisa:	R4006, 24.09.2019.
Turnitin podudaranje [%]:	2

Ovom izjavom izjavljujem da je rad pod nazivom**: Izrada Android mobilne aplikacije za ugađanje tona akustične gitare**

izrađen pod vodstvom mentora Izv. prof. dr. sc. Damir Blažević

i sumentora

moj vlastiti rad i prema mom najboljem znanju ne sadrži prethodno objavljene ili neobjavljene pisane materijale drugih osoba, osim onih koji su izričito priznati navođenjem literature i drugih izvora informacija.
Izjavljujem da je intelektualni sadržaj navedenog rada proizvod mog vlastitog rada, osim u onom dijelu za koji mi je bila potrebna pomoć mentora, sumentora i drugih osoba, a što je izričito navedeno u radu.

Potpis studenta:

SADRŽAJ

1. UVOD 1

2. AKUSTIKA, ZVUK I TON 2

2.1. Općenito o akustici i zvuku 2

2.3. Stojni (stacionarni) val 2

2.2. Ton 3

2.3. Akustična gitara 3

3. POSTOJEĆA RJEŠENJA 5

3.1. GuitarTuna 5

3.2. Tuner – gStrings Free 6

3.3. DaTuner 6

4. OPIS PRIMJENJENIH TEHNOLOGIJA 8

4.1. Android studio 8

4.2. XML opisni jezik 8

4.3. Java programski jezik 9

4.4. TarsosDSP biblioteka 9

5. REALIZACIJA MOBILNE APLIKACIJE 10

5.1. Kreiranje projekta u Android Studiju 10

5.2. Dodavanje dozvola 11

5.3. Dodavanje biblioteke (Tarsos DSP) 11

5.4. Izgled (XML) 13

5.5. Funkcionalnost (Java) 14

6. TESTIRANJE APLIKACIJE I ANALIZIRANJE REZULTATA 17

6.1. Testiranje preciznosti aplikacije pri ugađanju tona 17

6.2. Testiranje pomoću digitalnoga klavira 17

6.3. Usporedba preciznosti aplikacije sa TC Electronic UniTune-om 19

7. ZAKLJUČAK 20
LITERATURA 21
SAŽETAK 23
ABSTRACT 24
ŽIVOTOPIS 25

1. UVOD

Zadatak ovoga završnoga rada je realizacija Android mobilne aplikacije koja glazbenicima, bili oni amateri ili profesionalci, omogućuje ugađanje tona akustične gitare putem tvornički ugrađenoga mikrofona u mobilnome uređaju.

Aplikacija je nastala unutar Android studio programskoga okruženja. Dizajn i izgled definirani su u XML opisnome jeziku. Funkcionalnost aplikacije napisana je u Java jeziku pri čemu je korištena vanjska biblioteka Tarsos DSP.

Na samome početku iza uvoda nalazi se drugo poglavlje o području fizike koju nazivamo akustika. Grana fizike koja se bavi pojmovima vezanim uz nastanak i prijenos zvuka. Također u poglavlju se nalazi teorijski dio o stojnome valu, tonu i akustičnoj gitari koji pobliže objašnjava pojmove koji su ključni za razumijevanje konačne svrhe ove aplikacije. U trećem poglavlju prikazana su neka od trenutnih postojećih rješenja. Objašnjeno je kako se koje postojeće rješenje koristi pomoću slika zaslona sa mobilnoga uređaja. Četvrto poglavlje sadrži teorijsku podlogu o tehnologijama pomoću kojih je stvorena aplikacija. Peto poglavlje bavi se realizacijom aplikacije unutar Android studio programskoga okruženja. Prikazani su najbitniji koraci stvaranja aplikacije. Šesto poglavlje bavi se testiranjem rada ove aplikacije naspram postojećih rješenja te digitalnoga uređaja za ugađanje tona. Analiza se temelji na rezultatima mjerenja prikazanim grafovima. Na kraju stoji poglavlje koje sadržava zaključak donesen na temelju cjelokupnoga rada.

1.1. Zadatak završnoga rada

Zadatak završnoga rada je napraviti Android mobilnu aplikaciju unutar Android studio programskog okruženja čija je glavna funkcionalnost ugađanje tona akustične gitare. Treba objasniti najbitnije korake stvaranja aplikacije, kao i tehnologije koje se koriste u pojedinome koraku. Mobilnu aplikaciju treba testirati i prikazati rezultate mjerenja u odnosu na postojeća rješenja.

2. AKUSTIKA, ZVUK I TON

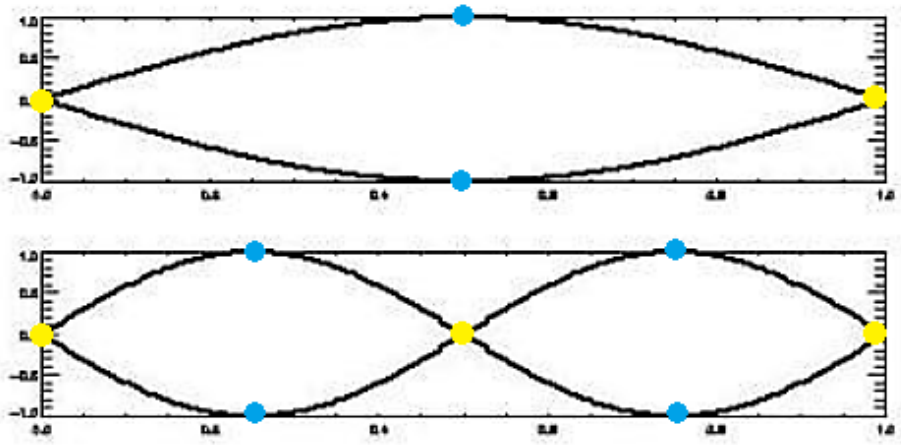
Ovo poglavlje završnoga rada je sažeta teorijska podloga o akustici kao znanosti, budući da se bavi proučavanjem zvuka općenito. Radi boljega razumijevanja svrhe ove aplikacije u nastavku teksta objašnjeno je što to povezuje zvuk, ton i akustičnu gitaru.

2.1. Općenito o akustici i zvuku

Prema [1], akustika je znanost, zasebna grana fizike, koja se bavi zvukom. Proučava nastanak, prenošenje (širenje) i osjet (prihvaćanje) zvuka. Dalje, prema [2], Zvuk se definira kao mehanički val čiji je raspon između 16 Hz i 20 000 Hz. Ujedno taj raspon odgovara rasponu koji ljudsko uho čuje. Zvuk nastaje kada izvor zvuka periodički titra te u svom neposrednom okruženju prouzrokuje promjenu tlaka sredstva u kojemu se širi. Zvukom se prenose energija i impuls sile, ali se pri njegovom širenju ne odvija prijenos mase.

2.3. Stojni (stacionarni) val

Stojnim valom nazivamo val koji je nastao zbrajanjem (superpozicijom) dva vala čije su valne duljine i amplitude jednake. Mjesta u kojima je amplituda jednaka nuli nazivamo čvorovima, a mjesta na kojima je amplituda najveća trbusima. Na slici 2.1. prvi val predstavlja stacionarni val napete niti koja je učvršćena na krajevima označenim žutom bojom, ujedno oni predstavljaju čvorove vala, dok plave točke predstavljaju trbuhe stojnoga vala. Ako bismo naslonili prst na sredinu niti i djelovali silom koja bi uzrokovala titranje tada bi dobili prvi harmonik, donji val na slici 2.1, odnosno val sa tri čvora i dvostruko većom frekvencijom titranja.



Slika 2.1. Stojni val [3]

2.2. Ton

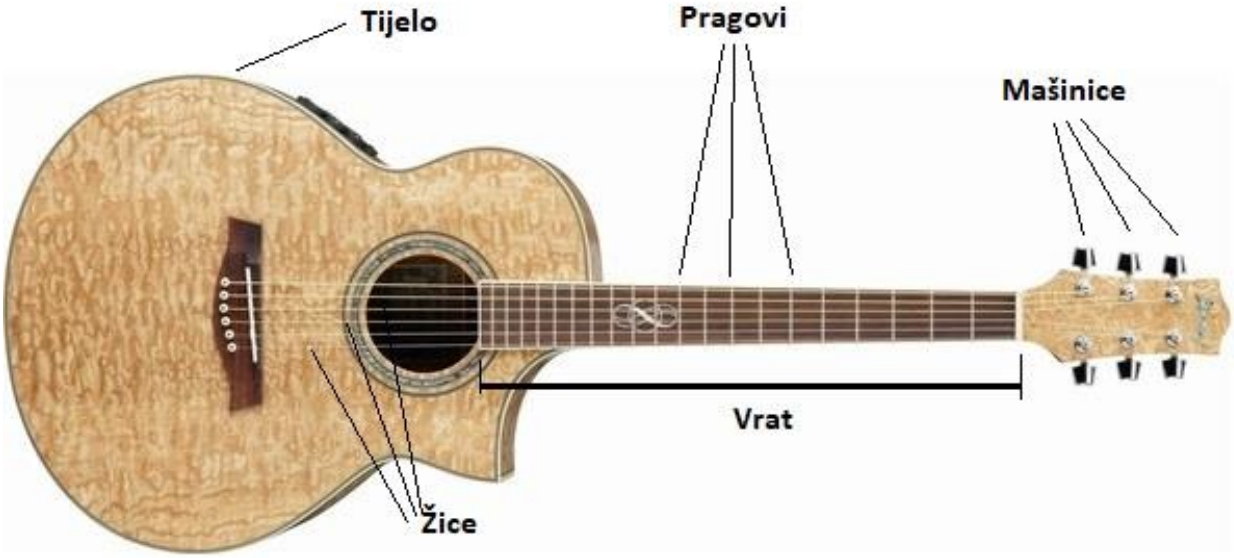
Prema [4] tonom nazivamo složeni zvuk nastao periodičnim i pravilnim titranjem izvora zvuka. Za razliku od ostalih neodređenih zvukova (šumova) ton karakteriziraju četiri određene stavke:

- visina tona - određuje je frekvencija titranja izvora zvuka
- jačina tona - određuje je amplituda izvora
- trajanje tona - razlika između početka i kraja titranja izvora
- boja tona – kod glazbala boju uvjetuju različiti materijali, način izrade, veličina i oblik glazbala

Visine tonova dogovorene su na svjetskoj razini, a nazivi tonova su u većini svijeta slični. Materijali koji se koriste na glazbenim instrumentima podložni promjenama u vremenu čiji uzrok mogu biti promjena temperature, promjena vlažnosti zraka, nakupljanje nečistoća i slično. Tako se žica glazbala koja titranjem proizvodi ton s vremenom može rastegnuti što uzrokuje odstupanje od dogovorene visine tona. Iz toga razloga potrebno je ugoditi visinu tona prije korištenja glazbenih instrumenata.

2.3. Akustična gitara

Akustična gitara, slika 2.2., je glazbeni instrument koji je u većini slučajeva izrađen od drveta, moguće su izvedbe od metala, plastike, ugljičnih vlakana i slično. Najbitniji dijelovi su tijelo, vrat i žice. Najčešće ima šest metalnih žica različitih debljina. Žice gitare su na jednome kraju učvršćene na početku vrata i pružaju njegovom cijelom dužinom do drugog kraja gdje su pričvršćene za tijelo gitare. Zvuk nastaje okidanjem jedne ili više žica, bilo prstima ili trzalicom. Na taj način žica učvršćena na dva kraja periodički titra i prenosi vibracije na tijelo gitare koje se ponaša kao svojevrsno pojačalo. Na vratu gitare nalaze se metalni pragovi. Pritiskom žica od pragove prstima druge ruke skraćuje se duljina na kojoj se odvija periodičko titranje i time mijenja frekvencija kojom žica titra. Tako je gitaristu omogućeno sviranje tonova različitih visina. Visinu tona ugađa se pomoću mašinica koje napinju žice, a nalaze se na vrhu vrata gitare. Tako se otpuštanjem žice visina tona snižava, a zatezanjem povećava.



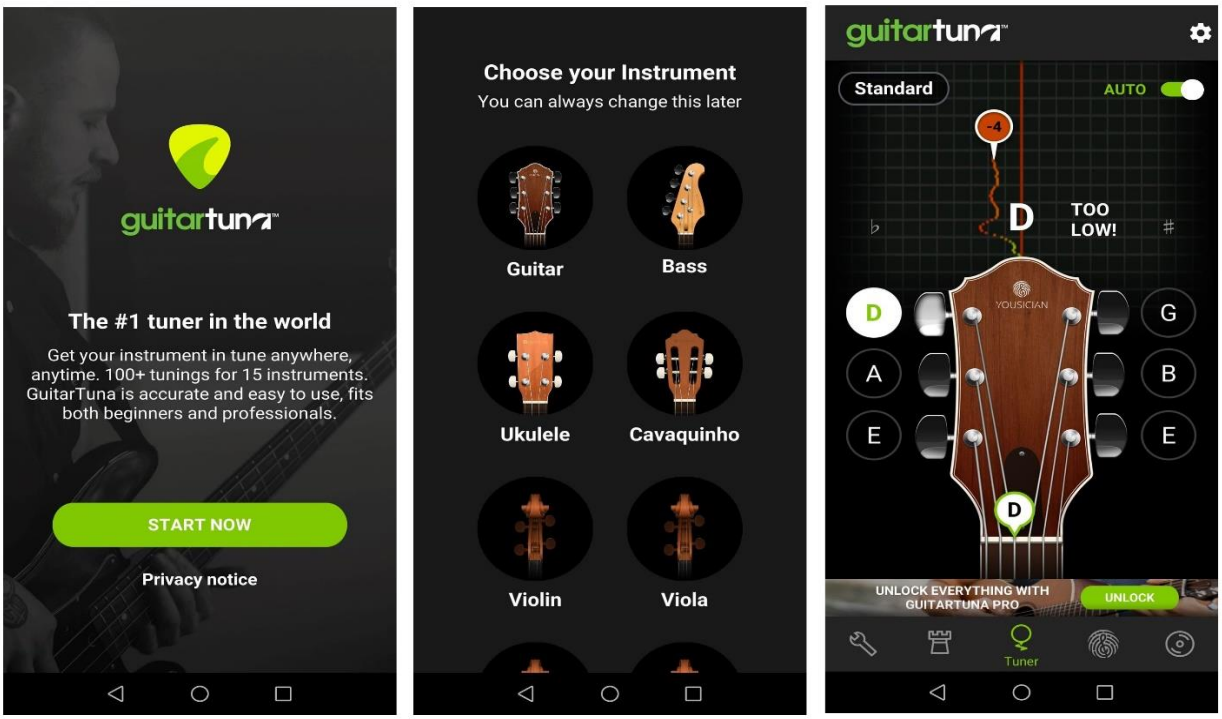
Slika 2.2. Dijelovi akustične gitare [5]

3. POSTOJEĆA RJEŠENJA

Unutar ovoga poglavlja predstavljena su neka od postojećih rješenja koja su trenutno dostupna na *Trgovini Google Play*. Ova rješenja korisnici su ocijenili najvišom prosječnom ocjenom te se stoga nalaze u nastavku ovoga rada. Objašnjeno je kako se koristi kojim rješenjem te je slikama prikazan njihov dizajn i konačni izgled.

3.1. GuitarTuna

Ova aplikacija nalazi se na prvome mjestu jer je trenutno najpopularnija na svjetskoj razini s ocjenom 4.8 od maksimalnih 5. Prema [6], ova aplikacija instalirana je na 50 i više milijuna Android mobilnih uređaja diljem svijeta. *GuitarTuna* aplikacija dostupna je i na iOS uređajima te je tako ukupno sa Android uređajima preuzeta na preko 100 miliona uređaja.

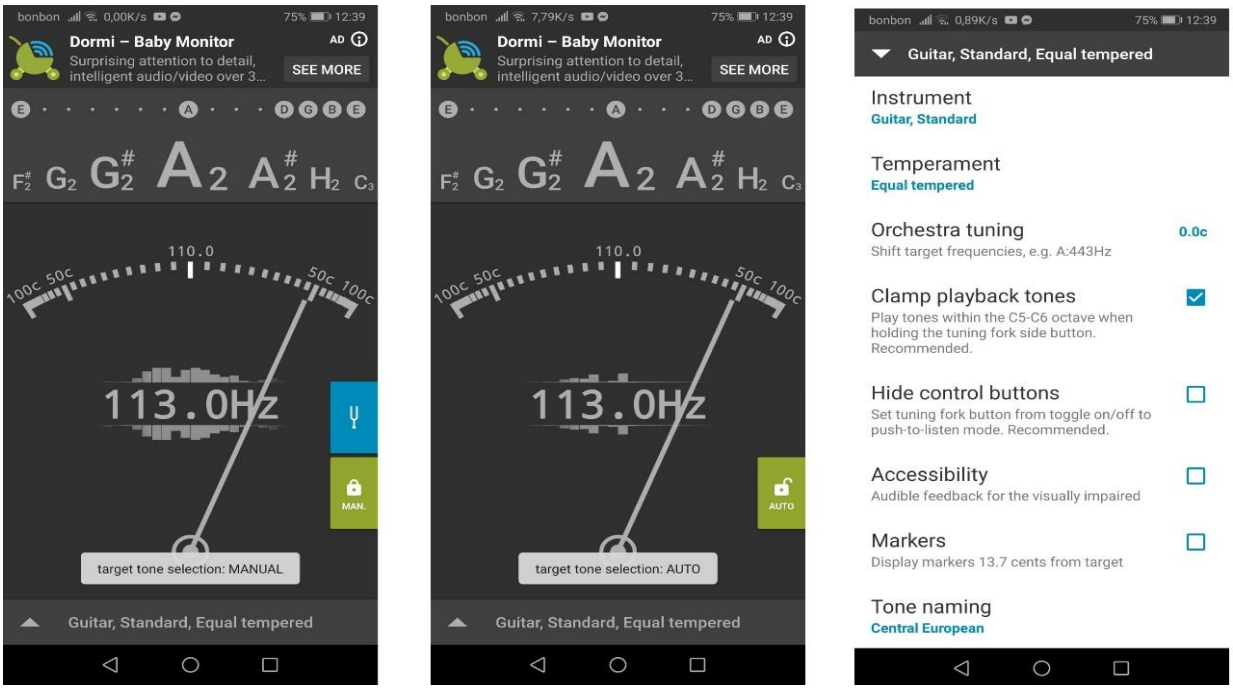


Slika 3.1. GuitarTuna - Izgled korisničkog sučelja

Vidljivo na slici 3.1., pri prvom pokretanju aplikacije korisnik može, osim gitare, odabrati još 9 drugih instrumenata . Ako odabere gitaru automatski je zadan standardni raspored tonova. Sve što treba je odsvirati žicu i iznad gitare se jasno vidi odstupa li ton.

3.2. Tuner – gStrings Free

GStrings mobilna aplikacija za ugađanje tona gitare trenutno ima ocjenu 4.6 od 5. Prema [7] preuzeta je na više od 220 tisuća mobilnih uređaja. Karakterizira je jednostavno korištenje i mnoštvo mogućnosti. Nakon instalacije sve što korisnik treba je potvrditi dozvolu mikrofona. Kromatični raspored tonova (definirani su svi mogući tonovi u rasponu od 27.5-3322kHz) omogućava ugađanje tona za bilo koji instrument. Mogućnost izabira automatskoga ili ručnoga odabira određenoga tona mijenja se pritiskom na ikonu lokota kao na slici 3.2. Osim toga ima par korisnih postavki koje služe za prilagodbu prema potrebama korisnika. Jedina kritika korisnika su oglasi koje je moguće ugasiti jedino uz plaćanje aplikacije.

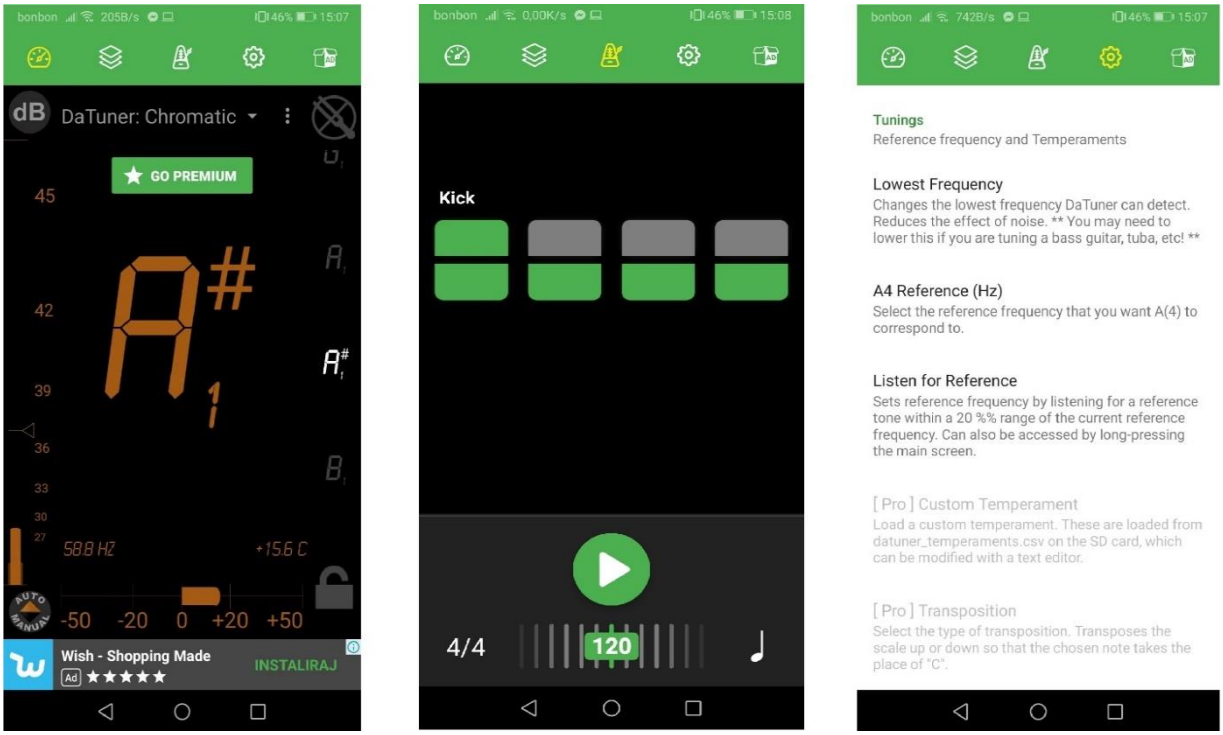


Slika 3.2. gStrings Free Tuner - Izgled korisničkog sučelja

3.3. DaTuner

Prema [8] ova aplikacija nosi ocjenu 4.4 od mogućih 5 na *Google Play-u*, uz to instalirana je na više od 10 milijuna mobilnih uređaja. Kao i kod gStrings-a radi se o kromatičnome rasporedu tonova te izbor instrumenta nije nužan prije uporabe. Za inačicu bez reklama sa više promjenjivih postavki nužna je godišnja nadoplata. Na slici 3.3. vidljivo je korisničko sučelje koje, osim ugađanja tona i raznih prilagodbi korisniku, nudi i jednostavan metronom. Na glavnome zaslonu, slika 3.3., pored trenutno prepoznatoga tona, sa lijeve strane nalazi se mjerač razine jakosti zvuka

u decibelima. Kao i kod gStrings-a pritiskom na ikonu lokota odabire se ručni odabir ugađanoga tona ili automatsko prepoznavanje tona.

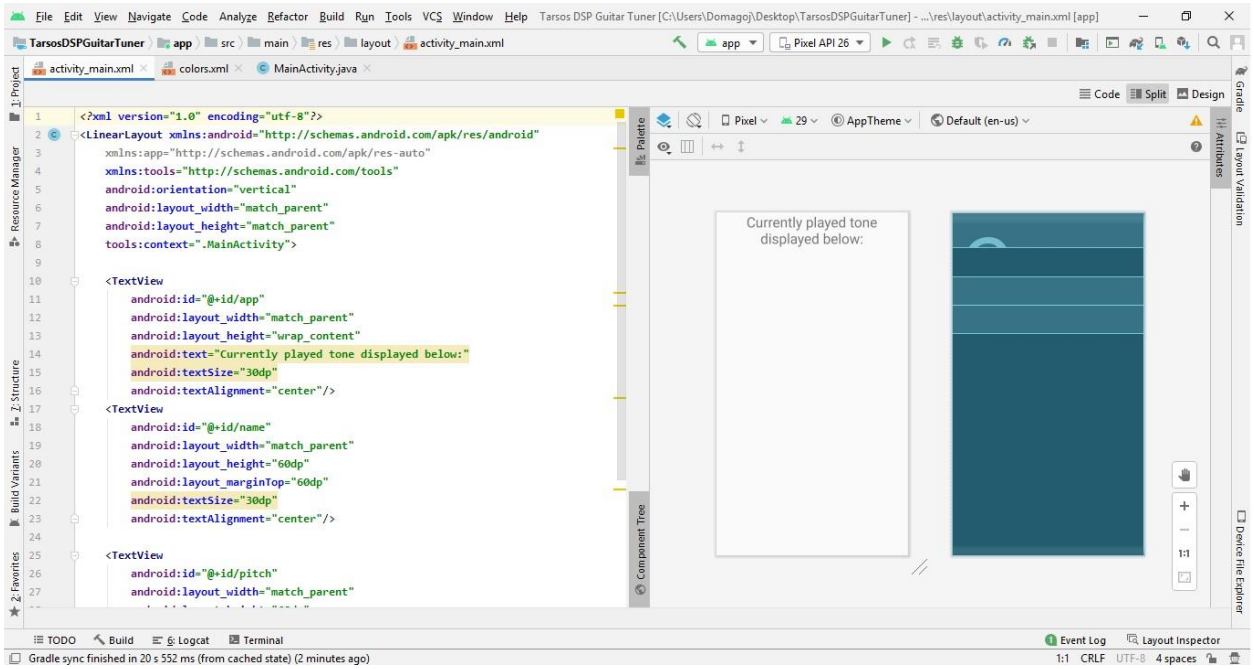


Slika 3.3. DaTuner - Izgled korisničkog sučelja

4. OPIS PRIMJENJENIH TEHNOLOGIJA

4.1. Android studio

Prema [9], Android studio, temeljen na InelliJ IDEA integriranom razvojnom okruženju, je integrirano razvojno okruženje dostupno za Linux, macOS i Windows čija je glavna svrha razvoj Android aplikacija. Izgled sučelja razvojnoga okruženja vidljiv je na slici 4.1.. Na Google I/O konferenciji u svibnju 2013. godine predstavljen je javnosti. Tek u prosincu 2014. godine objavljena je prva stabilna inačica 1.0. Java i C++ programski jezici i dalje su podržani iako je od svibnja 2014. službenim jezikom Android Studija proglašen Kotlin. U srpnju 2020. godine objavljena je najnovija inačica 4.0.1..



Slika 4.1. Android studio - Izgled korisničkog sučelja

4.2. XML opisni jezik

Eng. EXtensible Markup Lagnuage, skraćeno XML, naziv je proširivoga jezika koji služi za označavanje dokumenata i datoteka. Dalje, prema [10], svrstava se među jezike za označavanje kao što je za primjer HTML. Format u kojemu se jezik zapisuje vrlo je jednostavan i razumljiv. Najčešće se sprema u obliku tekstualne datoteke, a sastoji se od oznaka i sadržaja. XML ne smatramo programskim jezikom jer se ne „izvodi“. Na slici 4.2. prikazan je izgled XML koda unutar Android studio okruženja.

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:tools="http://schemas.android.com/tools"
android:orientation="vertical"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".QuestionFragment">

<TextView
    android:id="@+id/text_q"
    android:layout_width="wrap_content"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:textSize="30dp"
    android:layout_gravity="center"/>


</LinearLayout>
```

Slika 4.2. XML kôd unutar Android Studija

4.3. Java programski jezik

Java je programski jezik i računalna platforma koji je 1955. godine objavila tvrtka Sun Microsystems. Prije prve objavljene inačice razvijala se četiri godine. Dalje, prema [11] većina jezika u vremenu kada je java bila objavljena nije imala mogućnost izvoditi se na svim operacijskim sustavima bez prepravljanja kôda. Upravo je Java tu prepreku savladala uz postojanje *Java Virtual Machine-a* te tako omogućila izvođenje na svim operacijskim sustavima. Mobilne aplikacije, web aplikacije te serverske aplikacije banki samo dio stavki u kojima se primjenjuje Java. U ožujku 2020. objavljena je najnovija inačica Java 14.

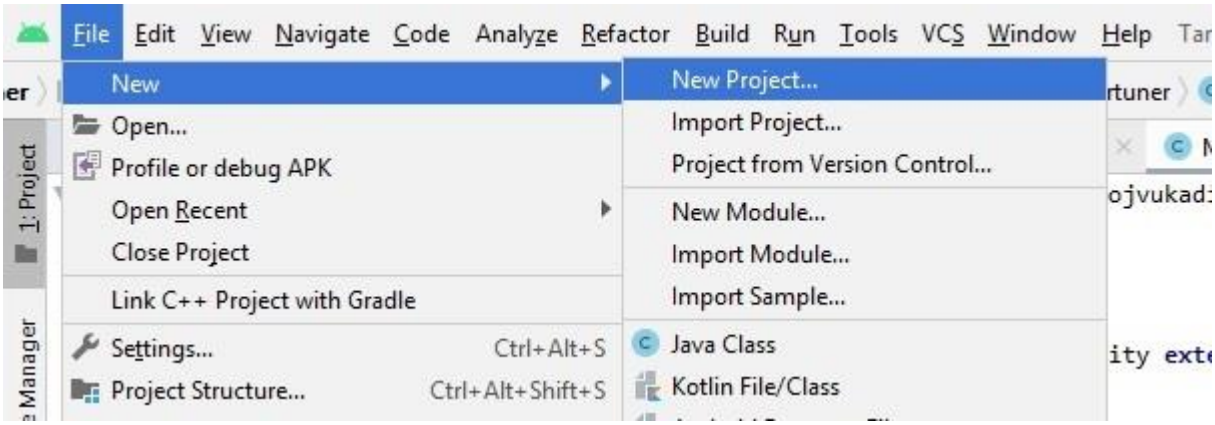
4.4. TarsosDSP biblioteka

Ova biblioteka napisana u Java programskome jeziku svoju svrhu pronalazi kod obrade zvuka u stvarnome vremenu. Znatno olakšava obradu zvukova jer je vrlo jednostavna za korištenje i ne sadrži nikakve vanjske zavisnosti. Prema [12], ova biblioteka osim algoritama za detekciju osnovne frekvencije sadržava i algoritam za skraćivanje odnosno produživanje zvučnih zapisa, ima par implementiranih filtera zvuka, zvučne efekte među kojima je najpopularniji algoritam za mijenjanje visine tona u stvarnome vremenu.

Službeno je predstavljena 2014. godine na 53. *AES(Audio Engineering Society)* konferenciji, a za njen nastanak zaslužni su Joren Six, Marc Leman i Olmo Cornesli, znanstvenici sveučilišta grada Ghent koji se nalazi u Belgiji.[13]

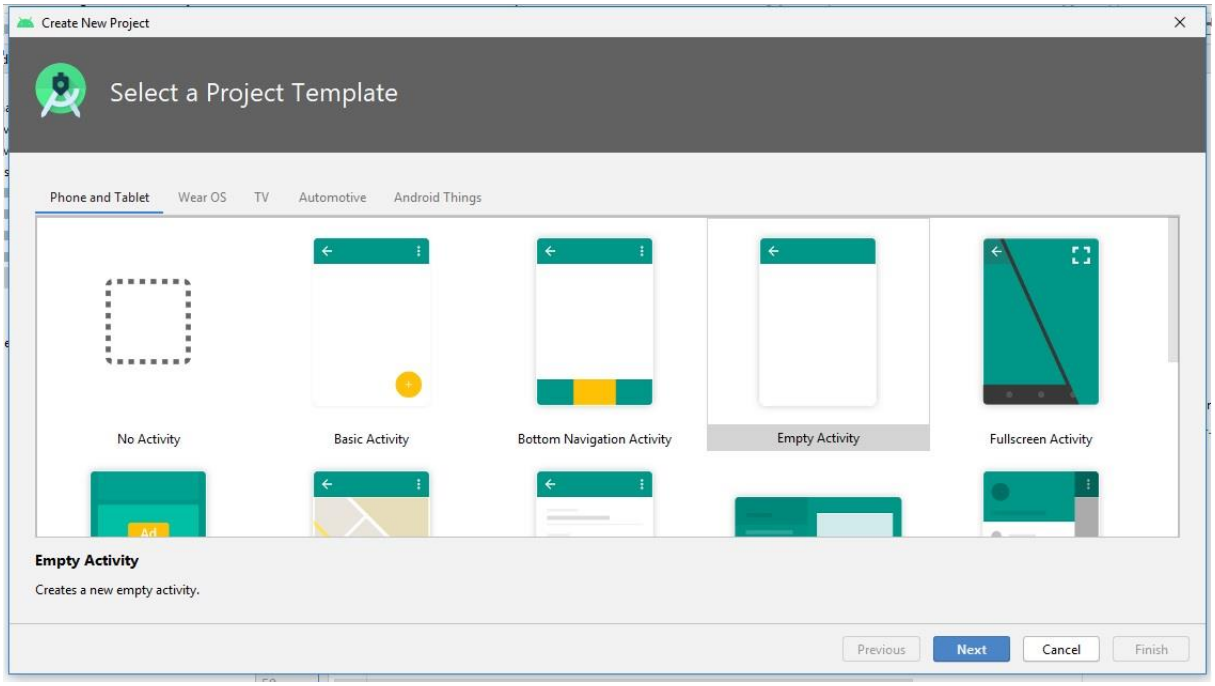
5. REALIZACIJA MOBILNE APLIKACIJE
5.1. Kreiranje projekta u Android Studiju

Slika 5.1. prikazuje redoslijed naredbi koje je potrebno klikom miša stisnuti unutar korisničkog sučelja Android Studija kada se nalazimo u otvorenome projektu. *File*, zatim *New* i konačno odabrati *New Project*.



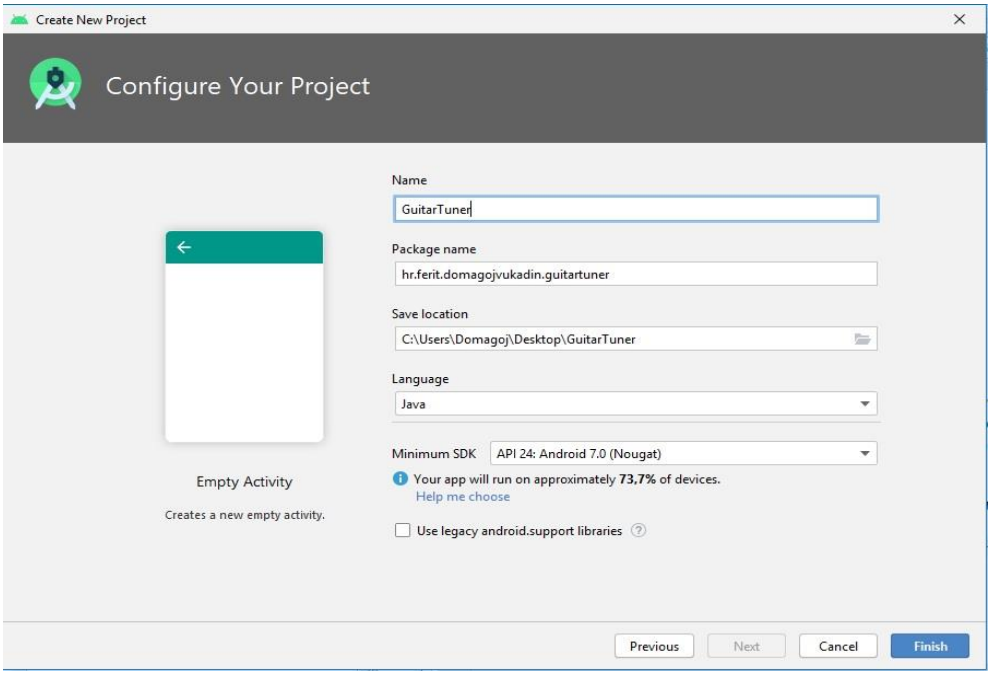
Slika 5.1. Android Studio korisničko sučelje

Potom se otvara zasebni prozor sa slike 5.2. unutar kojega treba odabrati *Empty Acitvity* i kliknuti na *Next*. Time je određeno kakav će biti izgled početnoga zaslona u budućoj aplikaciji. Osim toga, ovdje se vrši odabir, radi li se aplikacija za mobilni uređaj, tablet ili nešto drugo.



Slika 5.2. Odabir vrste zaslona i uređaja

Na slici 5.3. prikazan je idući korak u kojemu odabiremo ime aplikacije, naziv paketa, odabiremo lokaciju na lokalnom uređaju gdje će projekt biti pohranjen, programski jezik i najnižu inačicu verzije u kojoj će aplikacija biti podržana.



Slika 5.3. Konfiguracija novoga projekta

5.2. Dodavanje dozvola

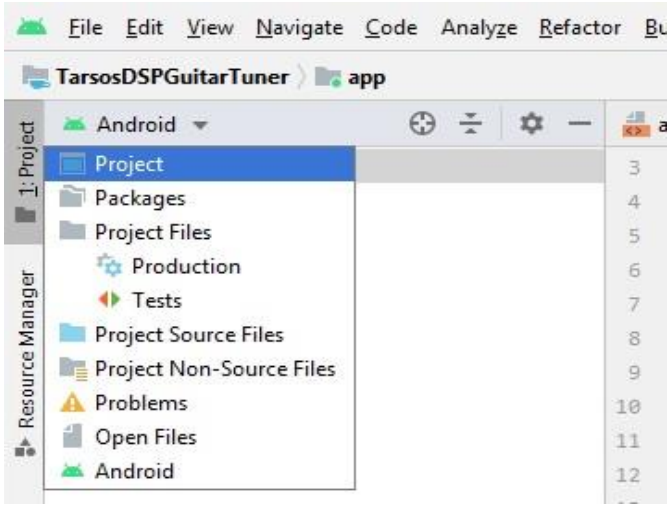
Unutar *Android Manifest* datoteke koja je napisana u XML opisnome jeziku potrebno je dodati dozvolu sustava, na slici 5.4. istaknute žutom bojom, za korištenje mikrofona. Ukoliko ovo ne napravimo aplikacija neće moći primati ulazni signal sa mikrofona uređaja.

Slika 5.4. Dozvola sustava u Android Manifest datoteci

5.3. Dodavanje biblioteke (Tarsos DSP)

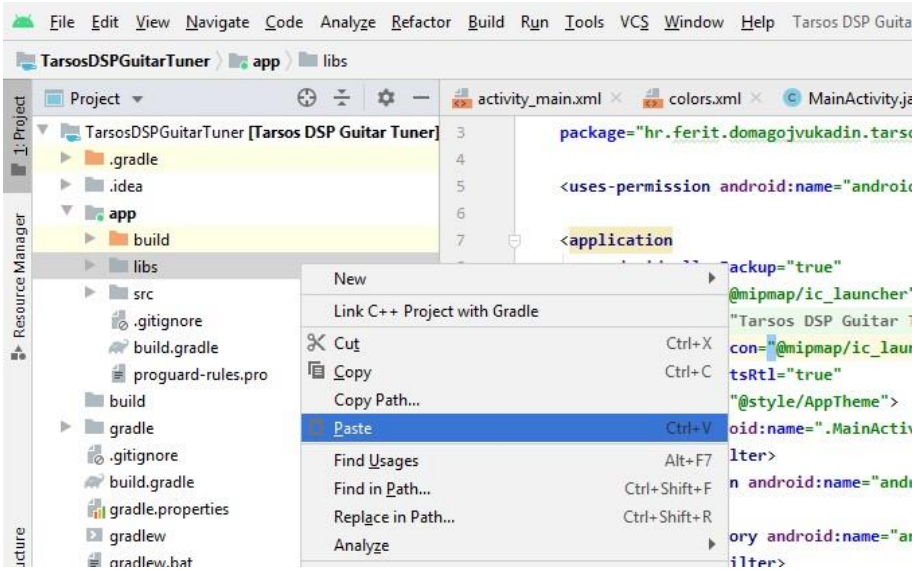
Idući korak prikazuje jedan od načina za pravilno dodavanje biblioteke koju ćemo kasnije moći koristiti pri pisanju koda u Java programskome jeziku. Na slici 5.5. je prikazan gornji lijevi ugao

korisničkoga sučelja u Android Studiju kada je otvoren projekt. Klikom na strelicu desno od natpisa *Android* otvara se izbornik u kojemu je potrebno odabrati *Project.*



Slika 5.5. Odabir pogleda na projekt

Nakon toga treba stisnuti desni klik na mapu *libs*, kao na slici 5.6., koja se nalazi unutar mape *app* te unutra zalijepiti zadnju inačicu Tarsos DSP biblioteke koju je moguće preuzeti na službenim stranicama pod [12].



Slika 5.6. Prikaz mape biblioteka

Nakon što je datoteka zalijepljena zadnji korak je stisnuti desni klik na datoteku i odabrati *Add As Library…* kako bi dodali biblioteku . Nakon toga bi se unutar zavisnosti, prikazano u kodu 5.7., trebala nalaziti linija koda istaknuta žutom bojom koja je svojevrsna potvrda da je biblioteka uspješno dodana.

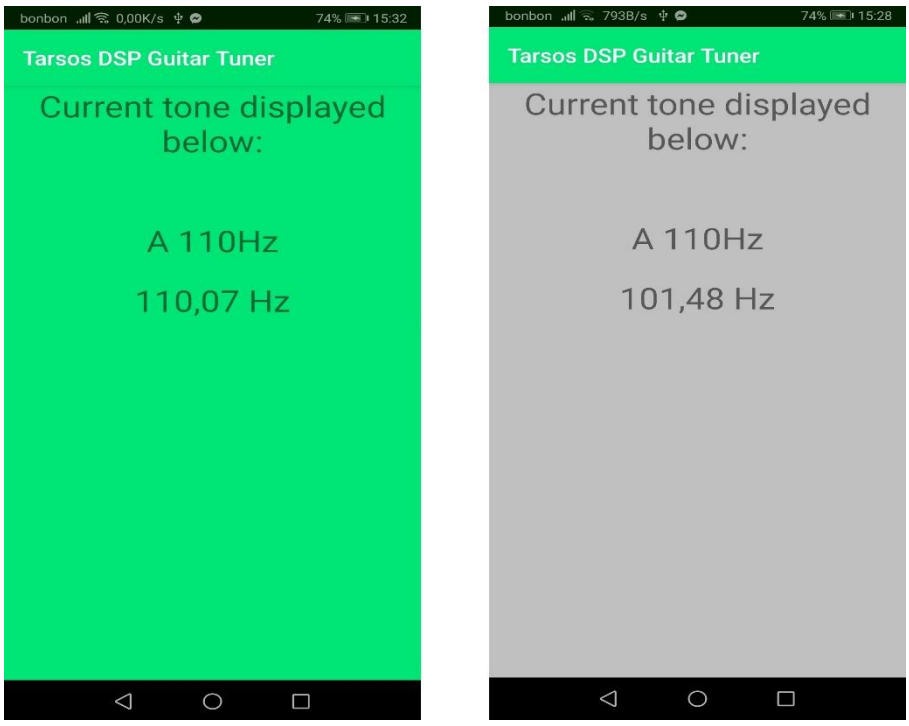
```
dependencies {
    implementation fileTree(dir: "libs", include: ["*.jar"])
    implementation 'androidx.appcompat:appcompat:1.1.0'
    implementation 'androidx.constraintlayout:constraintlayout:1.1.3'
    implementation files('libs\\TarsosDSP-latest.jar')
    testImplementation 'junit:junit:4.12'
    androidTestImplementation 'androidx.test.ext:junit:1.1.1'
    androidTestImplementation 'androidx.test.espresso:espresso-core:3.2.0'

}
```

Kôd 5.7. Prikaz zavisnosti

5.4. Izgled (XML)

Kako bi korištenje aplikacije bilo što jednostavnije, na glavnom zaslonu nalazi se prikaz osnovne frekvencije koju očitava mikrofon uređaja i najbliži ton sa frekvencijom u hercima. Takvim dizajnom, slika 5.8., pojednostavljeno je testiranje preciznosti ugađanja tona. Unutar *activity_main.xml* datoteke, u kodu 5.9., prikazana je realizacija ovih dvaju elemenata pomoću kojih se prikazuju frekvencije. Na slici 5.8., osim prikaza frekvencija, prikazana je promjena boje zaslona aplikacije. Zelena boja zaslona signalizira korisniku da je ton koji proizvodi žica gitare točno ugođen.



Slika 5.8. Prikaz izgleda glavnoga zaslona

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/app"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Currently played tone displayed below:"
        android:textSize="30dp"
        android:textAlignment="center"/>

    <TextView
        android:id="@+id/name"
        android:layout_width="match_parent"
        android:layout_height="60dp"
        android:layout_marginTop="60dp"
        android:textSize="30dp"
        android:textAlignment="center"/>

    <TextView
        android:id="@+id/pitch"
        android:layout_width="match_parent"
        android:layout_height="60dp"
        android:textSize="30dp"
        android:textAlignment="center"/>

</LinearLayout>
```

Kôd 5.9. Prikaz XML koda

5.5. Funkcionalnost (Java)

Java kôd ove aplikacije sastoji se od par temeljnih dijelova. Prvi dio nalazi se unutar *OnCreate()* metode koja se poziva kod otvaranja početnoga, ujedno i glavnoga, zaslona. Unutar metode, kôd 5.10., nalazi se kôd koji je zadužen za postavljanje mikrofona uređaja kao ulaza, pomoću metoda iz biblioteke koju smo dodali omogućen je prikaz signala sa mikrofona kao osnove frekvencije u hercima. U pozadini koda putem biblioteke odvija se pretvorba signala iz analognoga u digitalni i pronalazak osnovne frekvencije putem algoritma kojega odaberemo.

```
AudioDispatcher dispatcher = AudioDispatcherFactory.fromDefaultMicrophone( i: 22050, i1: 1024, i2: 0);

PitchDetectionHandler phd = (pitchDetectionResult, audioEvent) → {
        final float freqHz = pitchDetectionResult.getPitch();
        runOnUiThread(() → {
                frequencyList.add(freqHz);
                checkAndCalculate(freqText, noteText);

        });
};

AudioProcessor pitchProcessor = new PitchProcessor(PitchProcessor.PitchEstimationAlgorithm.FFT_YIN, v: 22050, i: 1024, phd);
dispatcher.addAudioProcessor(pitchProcessor);

Thread audioThread = new Thread(dispatcher, name: "Audio thread");
audioThread.start();
```

Kôd 5.10. Korištenje mogućnosti Tarsos DSP biblioteke

Svaki puta kada *PitchDetectionHandle*r dohvati frekvenciju ona se sprema u listu. Nakon toga poziva se metoda *checkAndCalculate()*, kôd 5.11., koja provjerava je li lista popunjena. Nakon što je lista ispunjena metoda računa prosječnu vrijednost frekvencije i poziva metode *freqDisplay()*, *toneDisplay()* i *changeColor()*. Na taj način je ostvaren uglađeniji prikaz frekvencija i povećana preciznost.

```
public void checkAndCalculate(TextView freqText, TextView noteText) {
    if (frequencyList.size() == 10) {
        float sum = 0f;
        for (float frequency : frequencyList) {
            sum += frequency;
        }
        float frequency = sum / 10;
        frequencyList.clear();

        freqDisplay(freqText, frequency);
        toneDisplay(noteText,frequency);
        changeColor(view, frequency);
    }
}
```

Kôd 5.11. Metoda *checkAndCalculate()*

Metoda *freqDisplay()*, kôd 5.12., prima frekvenciju i zaokružuje ju na dvije decimale. Ukoliko je frekvencija veća od sedamdeset herca prikazuje se na zaslonu.

```
public void freqDisplay(TextView freqText, float freqInHz) {

    String round = String.format("%.2f", freqInHz);
    round = round + " Hz";
    if (freqInHz >= 70) {
        freqText.setText(round);
    }
    else freqText.setText("");
}
```

Kôd 5.12. Metoda *freqDisplay()*

Metoda pod imenom *toneDisplay()* sadrži niz *if else* uvjeta kojima se pomoću frekvencije određuje koji će se ton prikazati na zaslonu aplikacije, kôd 5.13.. Na taj način ostvaren je odabir bilo kojega raspona frekvencija za koji se želi prikazati točan ton sa odgovarajućom frekvencijom.

```
public void toneDisplay(TextView noteText, float freqInHz){

    if(freqInHz < 70){
        noteText.setText("Play a string...");
    }
    else if(freqInHz >= 70 && freqInHz < 100) {
        noteText.setText("E 82.41Hz");
    }
    else if(freqInHz >= 100 && freqInHz < 130) {
        noteText.setText("A 110Hz");
```

Kôd 5.13. Logika unutar *toneDisplay()* metode

Na sličan način definirana je i metoda *changeColor()*, kod 5.14., koja preko if else uvjeta poziva metode *goGreen()* ili *goGrey()*, kod 5.15., s obzirom na to koliko trenutna frekvencija odstupa od tona na koji bi žica gitare trebala biti ugođena.

```
public void changeColor(View v, float freqInHz){

        if(freqInHz >= 82.21 && freqInHz < 82.61){
            goGreen(v);
        }
        else if(freqInHz >= 109.8 && freqInHz < 110.2){
            goGreen(v);
        }
        else if(freqInHz >= 146.6 && freqInHz < 147){
            goGreen(v);
```

Kôd 5.14. Logika unutar *changeColor()* metode

```
public void goGreen(View v) { view.setBackgroundResource(R.color.colorGreen); }

public void goGrey(View v) { view.setBackgroundResource(R.color.colorGrey); }
```

Kôd 5.15. Metode *goGreen()* i *goGrey()*

6. TESTIRANJE APLIKACIJE I ANALIZIRANJE REZULTATA

6.1. Testiranje preciznosti aplikacije pri ugađanju tona

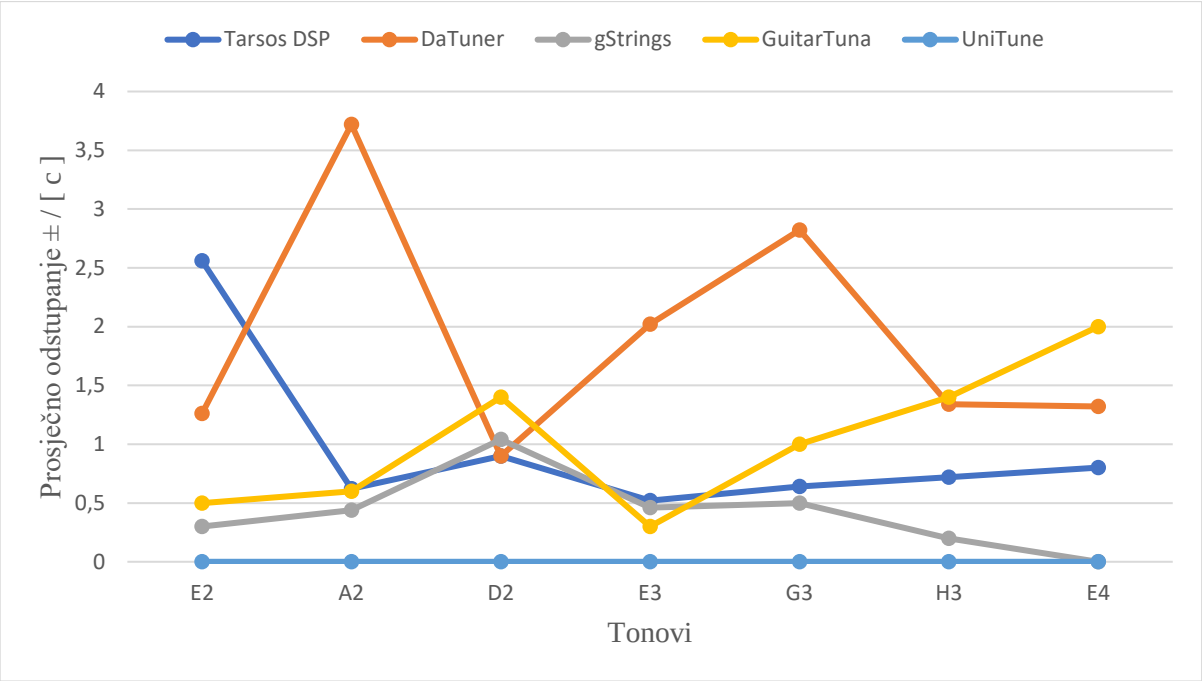
Cilj ovoga testiranja je utvrditi postoje li odstupanja između frekvencije koju proizvodi glazbalo i one koja se prikazuje na zaslonu uređaja. Za primjer uzmimo da žica gitare proizvodi zvuk od 110.55 Hz. Ako se na uređaju prikazuje 110Hz znači da je odstupanje 0.55 Hz. U tom slučaju aplikacija nije pouzdana za profesionalno korištenje. Ako bi se gitara svirala uz digitalni glasovir koji uvijek ima točno ugođene tonove nastala bi osjetna razlika.

Raspon od 110Hz do 220Hz, odnosno bila koja frekvencija pomnožena sa dva, naziva se oktava. Internacionalni dogovor jest da se svaki idući poluton računa kao umnožak frekvencije prethodnoga polutona i dvanaestoga korijena broja dva. Na taj način oktava je podijeljena na 12 polutonova. Tako je na primjer 116.5406Hz poluton koji dolazi nakon 110Hz. Kada se razlika između ta dva polutona podijeli sa sto dobije se jedan „cent“ (oznaka c), odnosno jedinica kojom mjerimo preciznost pri ugađanju tona.

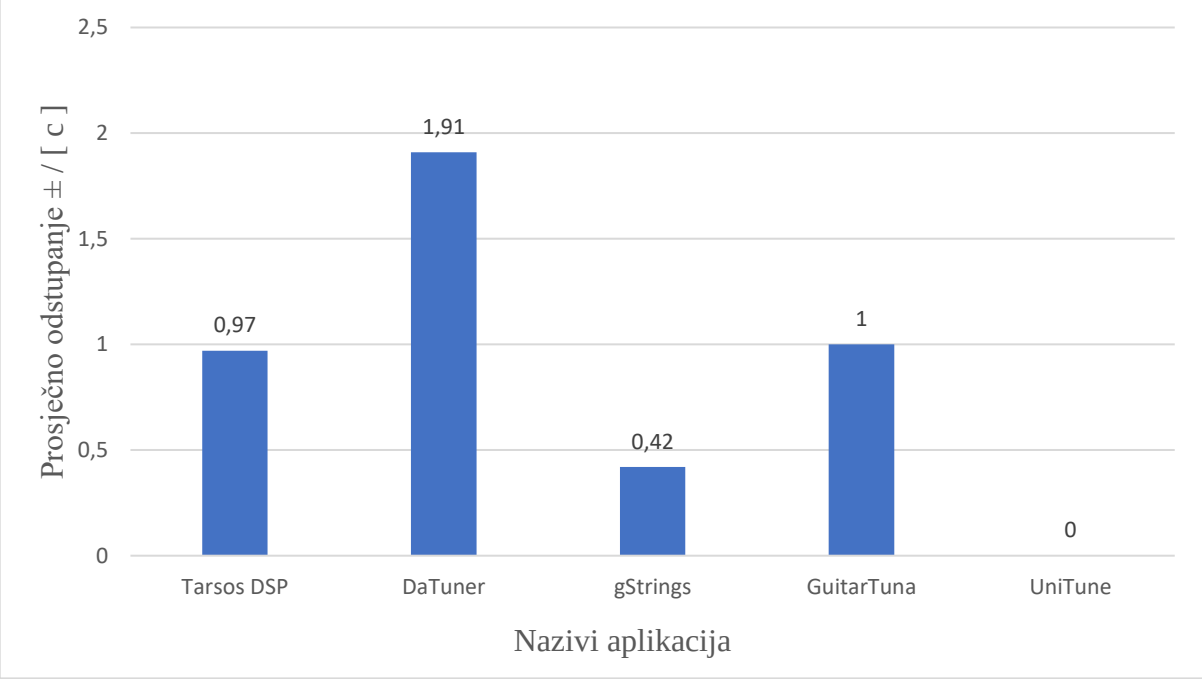
Kada su u pitanju akustična glazbala odstupanja u visini tona su uvijek prisutna zbog raznih čimbenika kao što su trenje, vlaga, temperaturne razlike i sl., stoga se teži ugađanju tona koje će biti što bliže točnoj frekvenciji. Prema [14], odstupanje koje iznosi ± 3c smatra se dovoljno dobrim za amaterske glazbenike, a odstupanja koja iznose ± 0.01c i manje smatraju se normom za profesionalne glazbenike. U obzir se uzima pretpostavka da će spektar ljudi sa savršenim sluhom vjerojatnije doći na nastup profesionalaca nego amatera te će osobi sa takvim sluhom odstupanje od ± 3c biti osjetno, čak i iritantno.

6.2. Testiranje pomoću digitalnoga klavira

Mobilni uređaj postavljen je desetak centimetara ispred zvučnika digitalnoga klavira. Na digitalnom klaviru naređen je zvuk akustične gitare. Testirane su četiri aplikacije i TC Electronic-ov UniTune. Osim standardnih šest tonova gitare još je dodana i oktava najdubljega tona s obzirom na to da mikrofon mobilnoga uređaja slabije reagira na niske frekvencije. Izvršeno je po deset mjerenja svake žice čija su prosječna odstupanja prikazana na grafu 6.1.



Graf 6.1. Prosječno odstupanje aplikacija naspram UniTune-a po tonovima

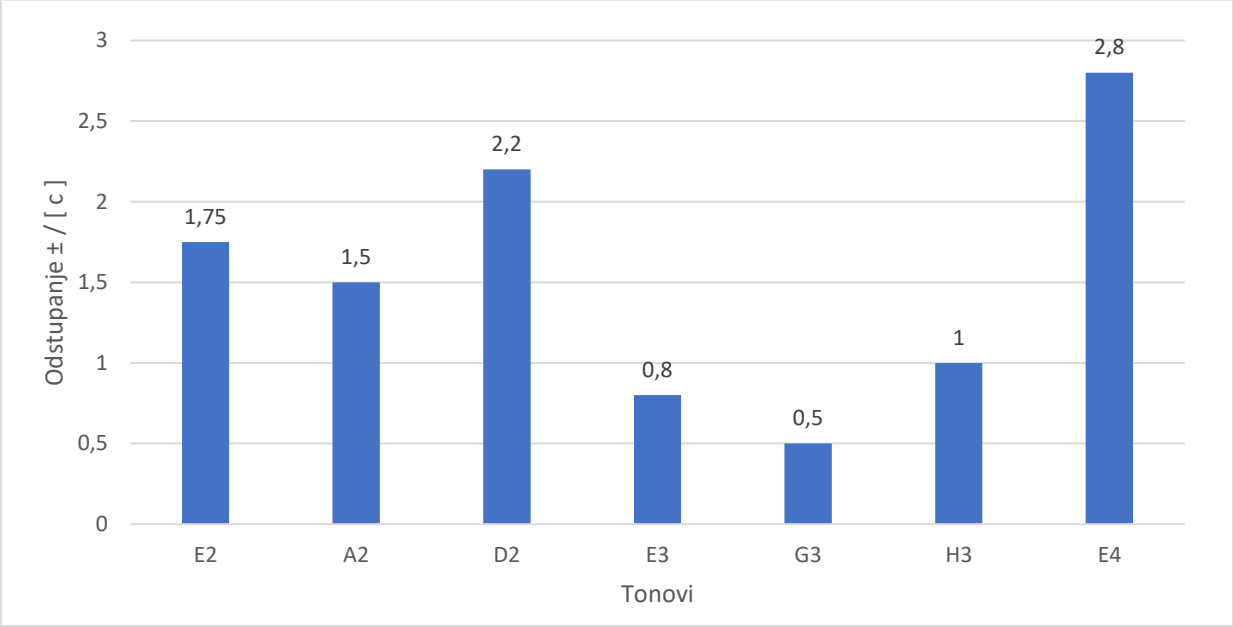


Graf 6.2. Prosječno odstupanje aplikacija naspram UniTune-a

Sa grafova 6.1. i 6.2. jasno je vidljivo da je aplikacija realizirana pomoću Tarsos DSP biblioteke dala izvrsne rezultate odnosno da je po preciznosti u rangu sa aplikacijom GuitarTuna koja je trenutno najpopularnija na svijetu.

6.3. Usporedba preciznosti aplikacije sa TC Electronic UniTune-om

Budući da UniTune nije imao odstupanja pri testiranju sa digitalnim klavirom pomoću njega je ugođen ton gitare kako bi se izmjerilo odstupanje mobilne aplikacije. Pametni telefon je postavljen desetak centimetara od gitare što odgovara stvarnoj situaciji kada gitaristi mobilni uređaj stave na koljeno prilikom sviranja gitare u sjedećem položaju. Na svakoj žici obavljeno je višestruko mjerenje koje je dalo je konstantne rezultate. Na grafu 6.3. prikazana su odstupanja dobivena mjerenjem. Uz ± 0.02 centa odstupanja koja propisuje proizvođač UniTune-a ukupna prosječna vrijednost odstupanja iznosi ± 1.77 centi.



Grad 6.3. Odstupanje aplikacije naspram UniTune-a

Razlog većega odstupanja pri mjerenju sa akustičnom gitarom jest taj da digitalni klavir imitira zvuk najskupljih gitara te stoga daje puno jasniji zvuk gitare.

7. ZAKLJUČAK

Cilj ovoga rada bio je realizacija Android mobilne aplikacije pomoću koje se ugađa ton akustične gitare. Pri razvoju je jasno prikazano korištenje Android Studio razvojnoga okruženja i uvod u korištenje Tarsos DSP biblioteke za obradu zvučnih signala. Nakon što je aplikacija uspješno realizirana obavljena su testiranja i usporedba rezultata mjerenja sa sličnim aplikacijama koje su trenutno dostupne za android uređaje. Činjenica da se aplikacija mjerenjem pokazala preciznijom od dvije druge aplikacije koje su trenutno vodeće dovoljno govori o uspješnosti realizacije. Prosječno odstupanje koje iznosi ± 0.97 centi (približno odstupanju od 1 posto) zadovoljiti će potrebe svakog amaterskog glazbenika. Korištenjem ove aplikacije amaterski glazbenici nemaju potrebu kupovati skupocjene uređaje za ugađanje tona već je dovoljno da samo preuzmu i instaliraju aplikaciju na svoj mobilni uređaj. Neka od daljnjih proširenja aplikacija mogu biti unapređenje dizajna, upotreba različitih algoritama za pronalazak osnovne frekvencije, povećanje broja uzoraka na temelju kojih se računa prosječna frekvencija te razna proširenja koja bi korisniku omogućila prilagođavanje prema njegovim potrebama.

LITERATURA

[1] Skupina autora, Akustika, Tehnička enciklopedija, dostupno na:
https://tehnika.lzmk.hr/tehnickaenciklopedija/akustika.pdf
(9.7.2020.)

[2] Skupina autora, Akustika, Wikipedija, dostupno na:
https://hr.wikipedia.org/wiki/Akustika#cite_note-1
(9.7.2020.)

[3] Skupina autora, Applying boundary conditions to standing waves, Briliant, dostupno na:
https://ds055uzetaobb.cloudfront.net/brioche/uploads/eLxQFDJ4ez-normalmodes.png?width=1500
(28.7.2020.)

[4] Skupina autora, Ton, Wikipedija, dostupno na:
https://hr.wikipedia.org/wiki/Ton
(9.7.2020.)

[5] Akustična gitara, dostupno na:
https://images.equipboard.com/uploads/item/image/5089/ibanez-ew20ase-nt-exotic-wood-acoustic-guitar-xl.jpg
(22.8.2020.)

[6] GuitarTuna, Google Play Trgovina, dostupno na:
https://play.google.com/store/apps/details?id=com.ovelin.guitartuna&hl=hr
(10.7.2020.)

[7] gStrings, Google Play Trgovina, dostupno na:
https://play.google.com/store/apps/details?id=org.cohortor.gstrings&hl=hr
(10.7.2020.)

[8] DaTuner, Google Play Trgovina, dostupno na:
https://play.google.com/store/apps/details?id=com.bork.dsp.datuna&hl=hr
(10.7.2020.)

[9] Skupina autora, Android studio, Developers, dostupno na:
https://developer.android.com/studio/intro
(20.8.2020.)

[10] Damir Kirasić, XML tehnologija i primjena u sustavima procesne informatike, IEEE, dostupno na:
https://www.ieee.hr/_download/repository/mipro_xml_tekst.pdf
(20.8.2020.)

[11] Skupina autora, Java, Wikipedija, dostupno na:
https://en.wikipedia.org/wiki/Java_(programming_language)
(20.8.2020.)

[12] Tarsos DSP, GitHub, dostupno na:
https://github.com/JorenSix/TarsosDSP
(10.7.2020.)

[13] Joren Six, Olmo Cornelis, Marc Leman, TarsosDSP, a Real-Time Audio ProcessingFramework in Java, AES 53RD INTERNATIONAL CONFERENCE, London, UK, 2014. dostupno na:
https://0110.be/files/attachments/411/aes53_tarsos_dsp.pdf
(11.7.2020.)

[14] Guitar tuner accuracy – How accurate is enough?, Carvin audio, dostupno na:
https://carvinaudio.com/blogs/guitar-bass-education/guitar-tuner-accuracy-how-accurate-is-enough#comments
(25.8.2020.)

SAŽETAK

Ovaj završni rad prikazuje izradu i testiranje mobilne aplikacije koja služi za ugađanje tona akustične gitare. Korisničko sučelje je vrlo jednostavno kako bi amaterski glazbenici mogli koristiti aplikaciju bez ikakvih nedoumica. Mobilna aplikacija razvijena je u Android Studio integriranom razvojnom okruženju. Funkcionalnost aplikacije realizirana je pomoću programskoga jezika Java uz primjenu biblioteke pod imenom TarsosDSP. Izgled korisničkog sučelja je definiran koristeći XML opisni jezik. Najbitniji koraci razvoja aplikacije i različita testiranja prikazani i jasno objašnjeni.

Ključne riječi: Android, gitara, glazba, Java

ABSTRACT

Development of an Android mobile application for tuning the tone of acoustic guitar

This final thesis shows the development and testing of the mobile aplication that serves as acoustic guitar tuner. The user interface is fairly simple so that amateur musicians can use the application without any hesitations.The mobile application was developed in Android Studio IDE. The functionality of the application has been evolved using the Java programing language with the use of a library called Tarsos DSP. User interface layout is defined using the XML descriptive language. The most important steps of application development and various tests are presented and clearly explained.

Keywords: Android, guitar, music, Java

ŽIVOTOPIS

Domagoj Vukadin rođen je 22.3.1997. u gradu Đakovu, Hrvatska. U rodnome gradu pohađao je i završio Osnovnu školu Vladimir Nazor, Đakovo. Nakon što je svaki razred završio odličnim uspjehom i osvojio prvo mjesto na županijskom natjecanju iz predmeta tehnička kultura, upisuje matematičko-prirodoslovnu gimnaziju Antun Gustav Matoš, Đakovu. Po završetku srednjoškolskoga obrazovanja 2016. godine upisuje preddiplomski sveučilišni studij na Fakultetu elektrotehnike, računarstva i informacijskih tehnologija u Osijeku, smjer računarstvo. Uz hrvatski jezik govori i dobro poznaje engleski. Tijekom fakultetskoga školovanja proširio je znanja iz programskih jezika C-a i HTML-a te stilskoga jezika CSS-a, a stekao je osnovna znanja iz C++ i Java programskoga jezika.