



Vrije
Universiteit
Brussel

Bachelorproef AMuRate
3e Bachelor Computerwetenschappen

David Sverdlov
dsverdlo@vub.ac.be

January 5, 2013



Contents

1	Inleiding	3
1.1	Doel	3
1.2	Referenties	3
1.2.1	AMuRate	3
1.2.2	Android	3
1.2.3	Ice Cream Sandwich	3
1.2.4	Last.fm	3
1.3	Afkortingen en definities	3
1.3.1	API	3
1.3.2	XML/JSON	3
2	Achtergrond	4
2.1	Last.fm/api: REST	4
2.2	REST architectuur	4
3	Voorgond	4
3.1	Website	4
3.2	Naam en logo	4
4	Vereisten	5
5	Implementatie	5
5.1	Database	5
6	Problemen	5
6.1	Synchronous tasks	5
6.2	Tracks verschillende JSON versies	5
6.3	Database layout	5
7	Voortgang	6

1 Inleiding

1.1 Doel

Het doel van dit project is om een applicatie te ontwikkelen voor het mobiele platform Android. Deze bachelorproef werd gepromote door prof. De applicatie moet gebruikers in staat stellen om muzieknnummers op te zoeken en een beoordeling te kunnen geven. Deze scores worden dan opgeslagen in een database. Er wordt gewerkt met een lokale en een externe database. De applicatie zal ontwikkeld worden voor Android versie 4.0 (Ice Cream Sandwich) om het grootste doelpubliek aan te spreken.

1.2 Referenties

1.2.1 AMuRate

<http://dsverdlo.github.com/AMuRate>

1.2.2 Android

<http://www.android.com/>

1.2.3 Ice Cream Sandwich

<http://www.android.com/about/ice-cream-sandwich/>

1.2.4 Last.fm

<http://www.last.fm/home>

1.3 Afkortingen en definities

1.3.1 API

API staat voor Application Programming Interface en zorgt vaak voor de communicatie tussen programmas, door de scheiding te vormen tussen verschillende lagen van abstracties.

1.3.2 XML/JSON

XML staat voor Extensible Markup Language en is een van de meest gebruikte opmaaktalen, die gestructureerde gegevens kunnen omzetten in platte tekst. (Om het zo makkelijk(er) door te kunnen sturen.)

JSON is een afkorting van JavaScript Object Notation en is een alternatieve simpele manier om objecten voor te stellen als platte tekst.

2 Achtergrond

2.1 Last.fm/api: REST

Om (informatie over) muziek te kunnen opzoeken moet er gebruikt worden van een grote online database met een werkbare API. Last.fm is een muziek recommendation service met een enorme online muziek database. Last.fm biedt ook een gratis (lees: voor geregistreerde gebruikers) API aan, die iedereen toelaat om mobiele/desktop programmas of web services te bouwen met hun data.

Om methoden (zoekopdrachten) van de API aan te vragen, moet er een call (oproep) gestuurd worden. Deze gebeurt via HTTP naar de Last.fm server die antwoordt via REST style XML (of JSON op aanvraag).

2.2 REST architectuur

NYI

3 Voorgond

3.1 Website

In dit project wordt er gebruik gemaakt van Git als versiebeheersysteem, omdat het een gratis, eenvoudige en betrouwbare manier is om de broncode te beheren. Nog een voordeel is, dat Github de mogelijkheid biedt om snel en simpele websites te maken voor projecten. De website voor dit project is dus terug te vinden op: <http://dsverdlo.github.com/AMuRate/>. Hier kan men de open-source broncode bekijken, het logboek of dit document raadplegen, en de applicatie (.apk) downloaden voor Android 4.0 toestellen.

3.2 Naam en logo

De naam van het project moet een kleine beschrijving geven van wat de applicatie doet/ waar hij voor dient. AMuRate komt van "Android Music Rating". Omdat het project vooral geïmplementeerd zou worden via een Android emulator, en simpelweg "AMR" te kort en onduidelijk zou zijn, werd er gekozen voor AMuRate. Het logo (zoals te zien op de voorpagina) is een gecomponeerde ster in een bol die iets weg heeft van een "dragon ball" (uit de Japanse animatieserie Dragonball Z). In de serie bestaan er maar 7 bollen op de hele wereld en wanneer die allemaal verzameld zijn, mag er een wens gedaan worden. Gelijkaardig aan hoe elke artiest hoopt 5 sterren (/ 7 dragon balls) te verzamelen. In de ster zelf zijn drie letters gekleurd ("AMR"), naar de naam van de applicatie.

4 Vereisten

ID	Beschrijving	Status
1	Titel (en artiest) opzoeken	done
2	Track volledig weergeven	done
3	Enkel op artiest opzoeken	in progress
4	Album tonen	done
5	Score opslagen in database	done
6	Score ophalen in lokale database	done
7	Score opsturen naar remote db	NYI
8	Zoek geschiedenis opslagen	in progress
9	Stream/preview 30sec songs	in progress
10	Onderzoek uitbreidbaarheid (ios?)	NYI
11	Informatie ophalen van andere media	NYI

5 Implementatie

5.1 Database

Insert UML database layout here

6 Problemen

6.1 Synchronous tasks

6.2 Tracks verschillende JSON versies

6.3 Database layout

Een record per track, met een counter, maar kan geen goed gemiddelde maken dan. Ook id

7 Voortgang

Datum	Informatie
24-10	Informatie ontvangen i.v.m. opdracht
07-11	Afspraak met begeleiders en promoter, acceptatie bachelorproef
21-11	Android project aangemaakt met simpele GUI. Ook Last.fm account aangemaakt en unieke API sleutel aangevraagd.
05-12	Gebruikers kunnen echte calls maken en muziek opzoeken. Zitten nog bugs in en er kan nog niet gezocht worden naar artiesten
19-12	Er kan ook gezocht worden op artiesten, GUI werd herzien, abstractie in code toegevoegd.
xx-xx	BLOK: Todo: interne database aanmaken en mee communiceren. Ook alle bugs eruit halen.