INSTITUTO POLITÉCNICO NACIONAL

ESCUELA SUPERIOR DE CÓMPUTO

Data Mining

Practice no. 3

ID3 Algorithm Implementation

Student: López Ayala Eric Alejandro

Professor: Hernández Contreras Euler

June 18th, 2018

Content

ABSTRACT....3
INTRODUCTION....3
THEORETICAL FRAMEWORK....4
    Data mining....4
    Decision Trees....5
    ID3 Algorithm....5
        Algorithm Pseudocode....6
    Modified ID3 Algorithm....7
        Algorithm Pseudocode....7
DATABASE ENTITY RELATIONAL MODEL....8
DATABASE RELATIONAL MODEL....9
USER MANUAL....10
REFERENCES....14

ABSTRACT

In this document, ID3 algorithm of decision tree is implemented in order to help us to make decisions over a real database implementation. In this work we will make use of a music application database to extract and process the given information.

Through a web application the user will decide which attributes and class wants to analyze and generate its respective ID3 decision tree.

A theoretical content will be review, to help us to understand how this algorithm works and that it is possible to apply it not only for specific purposes but for a much wider real-life application that require business intelligence and decision making.

INTRODUCTION

In today's world, the organizations strive for neck to neck competition. To exist in the market, every organization must take correct and efficient decisions. So, decision making activity is the most important activity for the businessmen.

They must analyze all the existing data and conditions for decision making. They must extract new information. Only because of this new information the decision makers can take competitive decisions.

The technique to extract new knowledge from the existing information is known as data mining, there are different techniques to mine the data from databases. One important technique is classification and segmentation, under which decision trees are created to predict the data from the existing one. Decision trees are created to predict the data from the existing one. Decision trees are created with the help of different algorithms. One such algorithm, namely, ID3 is used here.

Spotify is music streaming service which give us access to hundreds of songs and vary content of music artist from all over the world.

In addition to basic functions such as listening to music, Either way, you can:

- Chooser what you want to Search or Explore new music.
- Receive recommendations in customizable functions.
- Create your own music collections.
- See what are listening your friends or look for releases.
- Add your own radio list, in which there will be music streaming 24/7

Spotify is available in its website, which is compatible with the latest web browsers such as Mozilla Firefox, Google Chrome, and Safari. Soon Spotify will be available for your tablet and mobile devices.

THEORETICAL FRAMEWORK

Data mining

Data mining can be as the extraction of hidden predictive information from large databases, is a powerful new technology with great potential to help companies focus on the most important information in their data warehouses.

Decision Trees

A decision tree is a powerful and popular tool for classification an d prediction. Decision trees represent rules, a decision tree is a predictive model that, as its name implies, can be viewed as a tree. Specifically, each branch of the tree is a classification question and the leaves of the tree are partitions of the dataset with their classification.

Decision tree is a classifier in the form of a tree structure, where each node is either:

- A leaf node indicates the value of the target: attribute(class) of examples.
- A decision node, specifies some test to be carried out on a single attribute-value, with one branch and sub-tree for each possible outcome to test.

Decision making tree algorithms transform raw information into ruled based decision trees. The ID3 algorithm is one of the most common algorithms to make decision trees.

ID3 Algorithm

Most algorithms that have been developed for learning decision trees are variations on a core algorithm that employs a top-down, greedy search through the space of possible decision trees. Decision tree programs construct a decision tree from a set of training cases.

ID3 decision tree algorithm creates a hierarchical structure of classification rules “If-Then” looking for a tree. Much work has been done in the field of decision tree algorithm.

J. Ross Quinlan originally developed ID3 algorithm at the University of Sydney. He first presented ID3 in 1975 in a book “Machine Learning”, vol 1, no. 1. ID3 is based on the concept Learning System algorithm. ID3 is an acronym of Interactive Dichotomizer and it is a precursor of C45 algorithm which have applications for both Machine Learning as well as Processing of the Natural Language.

To understand how this algorithm works, first of all, it is necessary to understand the following concepts:

- **Node**: Name or identifier of an attribute.
- **Branch**: Possible values of an attribute associated to a node.
- **Leaf**: Set of named and classified examples of a class.
- **Attribute**: It is a factor that influence the classification or decision.
- **Class**: Possible values of a solution.
- **Example**: It is a combination of the given attributes.
- **Entropy**: It is the measure of randomness that a system has. That is, in a certain situation, the probability of each of the possible outcomes occurring.

  It is defined by the following formula:

  $$Entropy(S) = - p_p log_2 p_p - p_n log_2 p_n$$

  Where p_p is the proportion of possible examples according to their target classification.

- **Information Gain:** Measures how well a given attribute separates the training examples according to their target classification.

  It is defined by the following formula:

  $$Gain(S,A) = Entropy(S) - \sum_{v \in Value(A)} |Sv| / |S| Entropy(Sv)$$

Algorithm Pseudocode

```
-Create a root node for the tree.

-If all examples are positive, return the single-node tree root, with the label = +.

-If all examples are negative, return the single-node tree root, with the label = -.

-If the number of predicting attributes is empty, then return the single node tree Root,
with label = most common value of the target attribute.

-Otherwise Begin:

        - A = The attribute that best classifies examples.

        - Decision tree attribute for root = A.
        - For each possible value, $V_p$ of A.

                - Add a new Tree branch below root, corresponding to the test A = $v_i$

                - Let examples($V_i$), be the subset of examples that have value $V_i$ for A.

                - If examples ($V_i$) is empty, then below this new branch add a leaf
                node with label = most common target value in the examples.

        - Else below this branch add the subtree ID3 (Examples($V_i$)), Target Attribute,
        Attributes-{A})
- End

- Return root
```

Modified ID3 Algorithm

In ID3 Algorithm, every attribute has the binary valued domain (i.e positive and negative). But it is also possible that we have some specific attributes that have multiple valued domain (i.e high, medium, low). For such attributes the algorithm can be modified as bellow.

Algorithm Pseudocode

```
-Create a root node for the tree.

-If all examples are of the same value say high, medium, low. Return the single-node tree
root, with the label = most common value of the target attribute.

-If the number of predicting attributes is empty, then return the single node tree Root,
with label = most common value of the target attribute.

-Otherwise Begin:

        - A = The attribute that best classifies examples.

        - Decision tree attribute for root = A.

        - For each possible value, $V_p$ of A.

                - Add a new Tree branch below root, corresponding to the test A = $a_i$

                - Let examples($a_i$), be the subset of examples that have value $a_i$ for A.

                - If examples ($a_i$) is empty, then below this new branch add a leaf
                node with label = most common target value in the examples.
```

        - Else below this branch add the subtree ID3 (Examples(a_i)), Target Attribute, Attributes-{A})
    - End

    - Return root

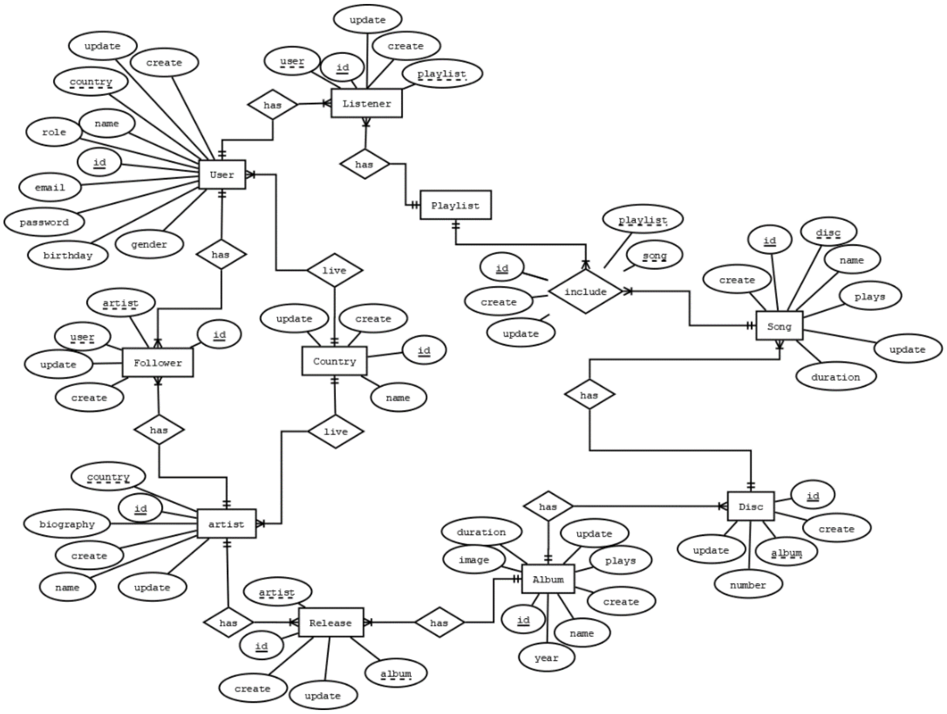
For implementing the modified ID3 algorithm, the information Gain and Entropy is calculated as follows:

$$\text{Gain}(S, \text{Attribute}) = \mathit{Entropy}(S) - \sum_{v \in \{v1, v2, \dots\}} \left(\frac{|Sv|}{|S|}\right) * \text{Entropy}(Sv)$$

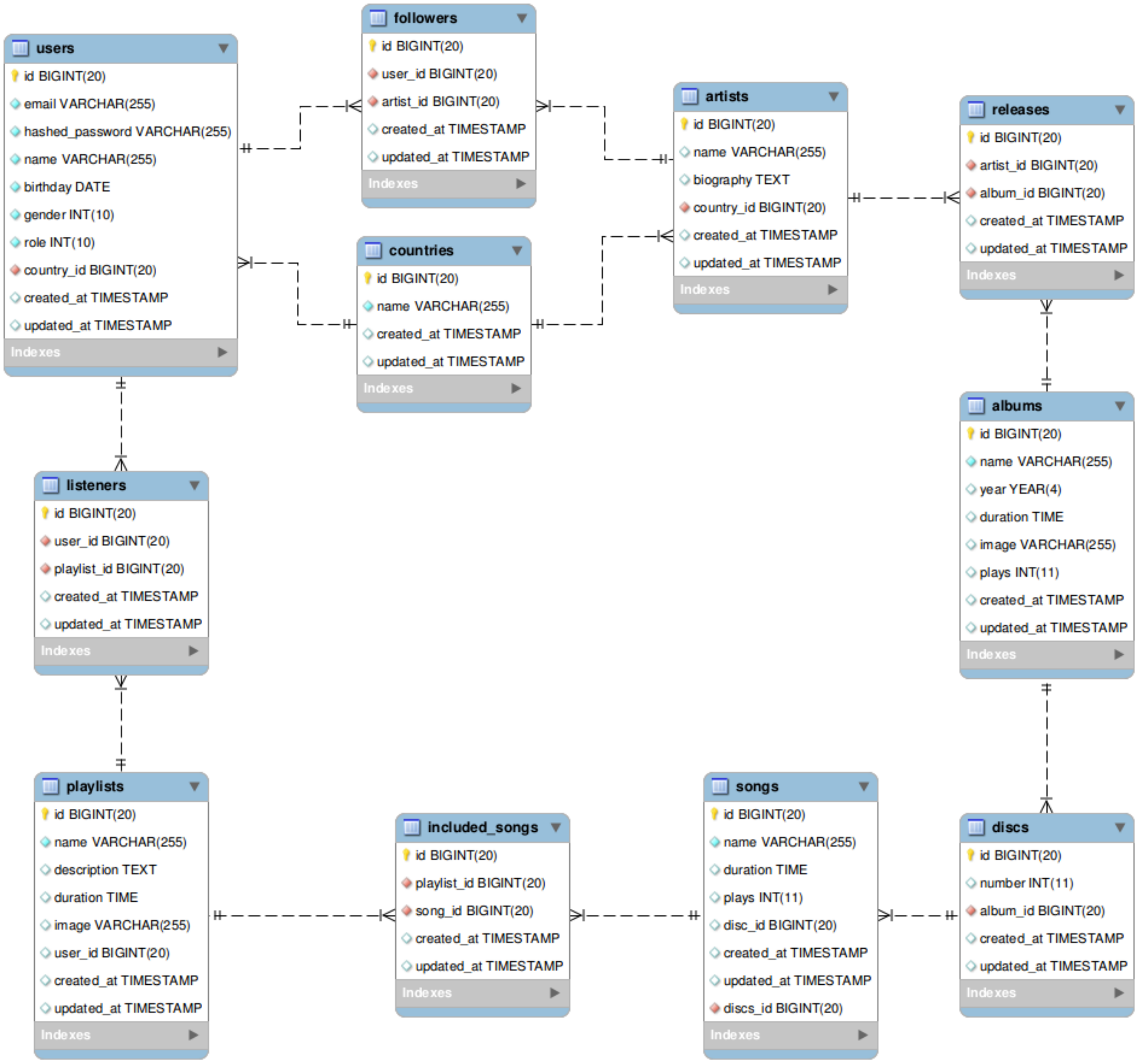
$$\text{Entropy}(S) = \sum_{i=1}^{c} -\text{pilog2pi}$$

$$\text{Gain}(S, \text{Attribute}) = \mathit{Entropy}(S) - \sum_{v \in \{v1, v2, \dots\}} \left(\frac{|Sv|}{|S|}\right) * \text{Entropy}(Sv)$$

DATABASE ENTITY RELATIONAL MODEL



DATABASE RELATIONAL MODEL

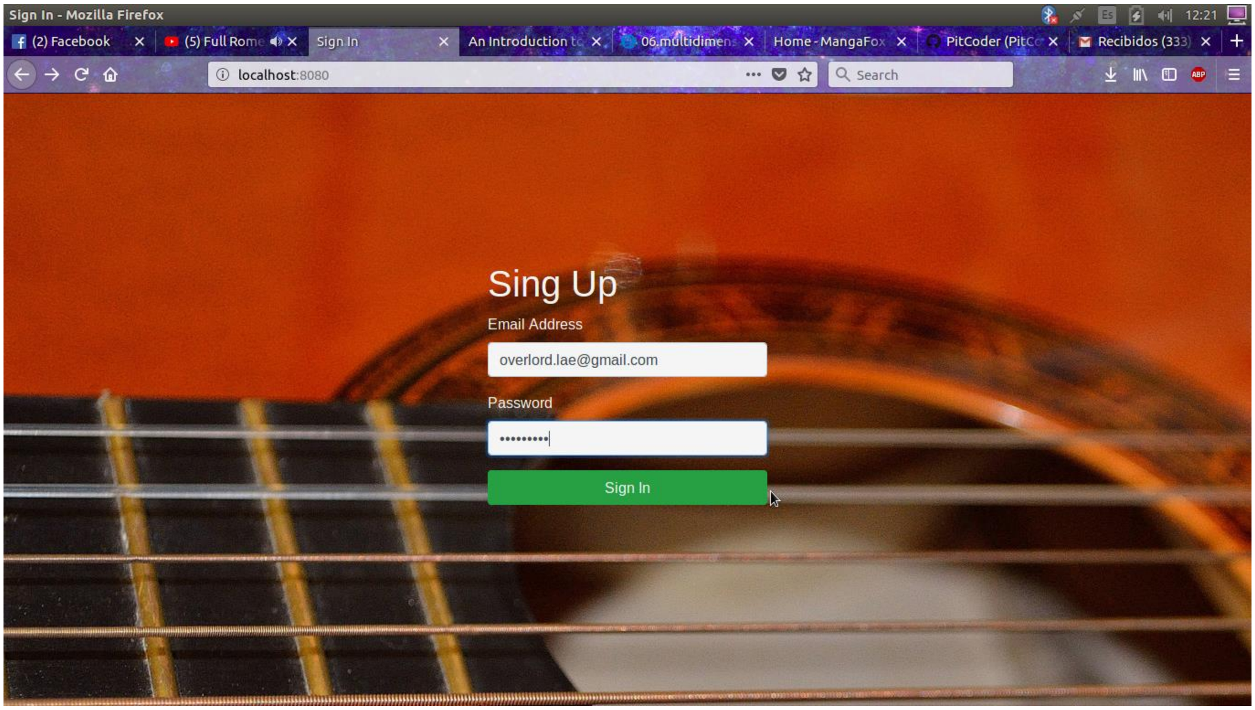


The following image shows our database relational model:

The model includes the entities, the attributes of each one and finally their relations between each other.

USER MANUAL

At first our user has to login into the system with Admin permissions, for this he must input an email and password matching to an admin role registered in the database.



localhost:8080

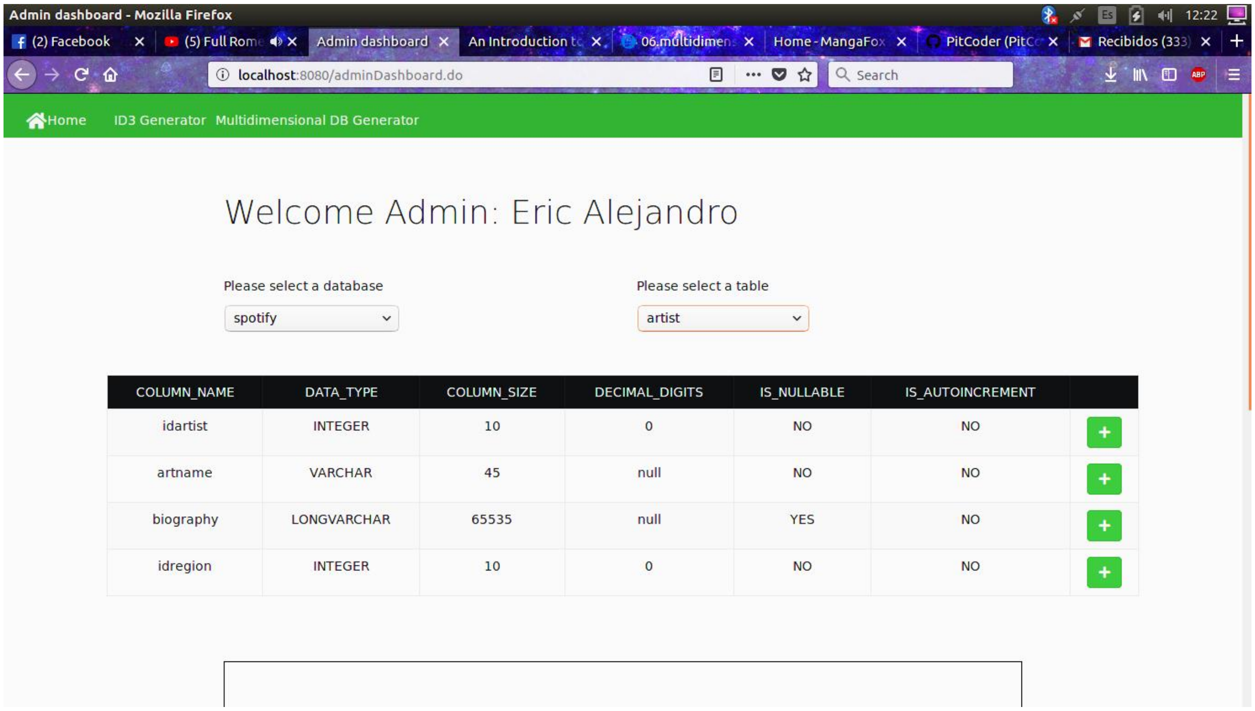
Sing Up

Email Address: overlord.lae@gmail.com

Password: •••••••••

Once the user has successfully logged in as an Admin, the following view will be display, then the user must do click in the “ID3 Generator” tab.

Then it must select a database and then one of its tables, for the ID3 generator “spotify” database must be selected, once the database and table had been selected the table metadata will be display as follows:


Admin dashboard - Mozilla Firefox
localhost:8080/adminDashboard.do
Home
ID3 Generator
Multidimensional DB Generator

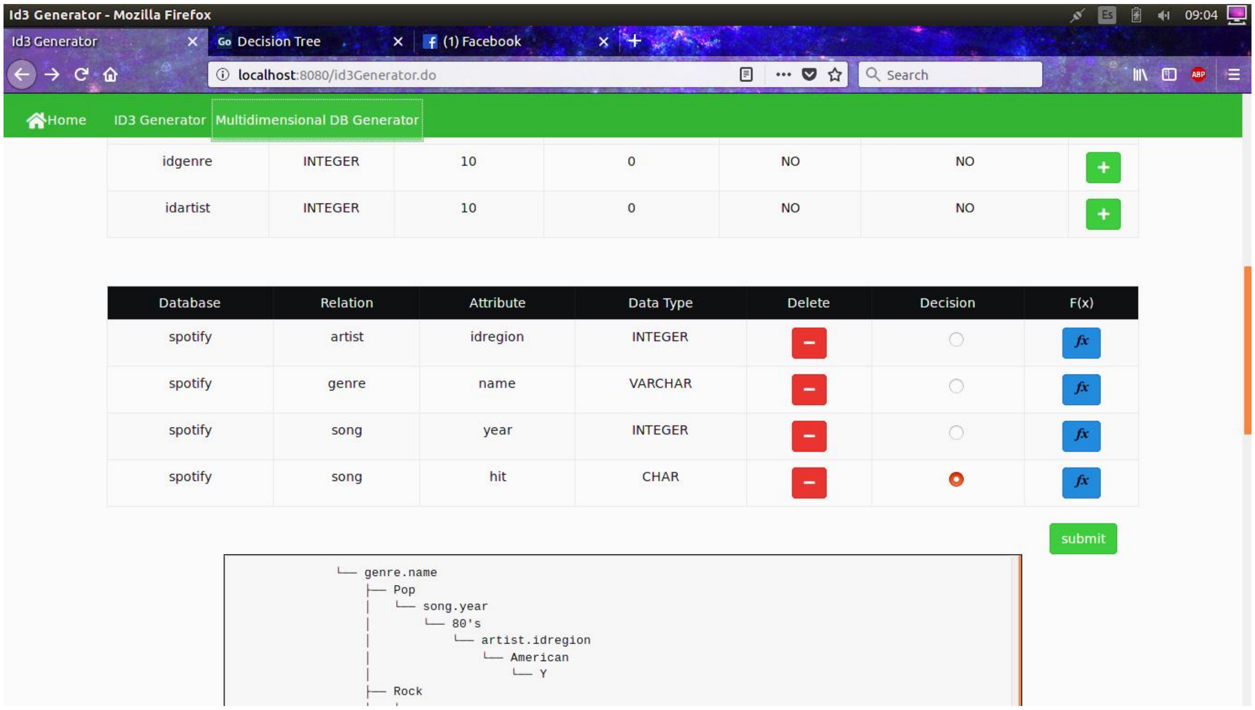
Welcome Admin: Eric Alejandro

Please select a database: spotify

Please select a table: artist

COLUMN_NAME	DATA_TYPE	COLUMN_SIZE	DECIMAL_DIGITS	IS_NULLABLE	IS_AUTOINCREMENT	
idartist	INTEGER	10	0	NO	NO	+
artname	VARCHAR	45	null	NO	NO	+
biography	LONGVARCHAR	65535	null	YES	NO	+
idregion	INTEGER	10	0	NO	NO	+

Once the table has appeared the user has the option to add the attribute for the system to analyze it, then that attribute will be added to the options selected and it will appear as follows:


Id3 Generator - Mozilla Firefox
localhost:8080/id3Generator.do
Home
ID3 Generator
Multidimensional DB Generator

idgenre	INTEGER	10	0	NO	NO	+
idartist	INTEGER	10	0	NO	NO	+

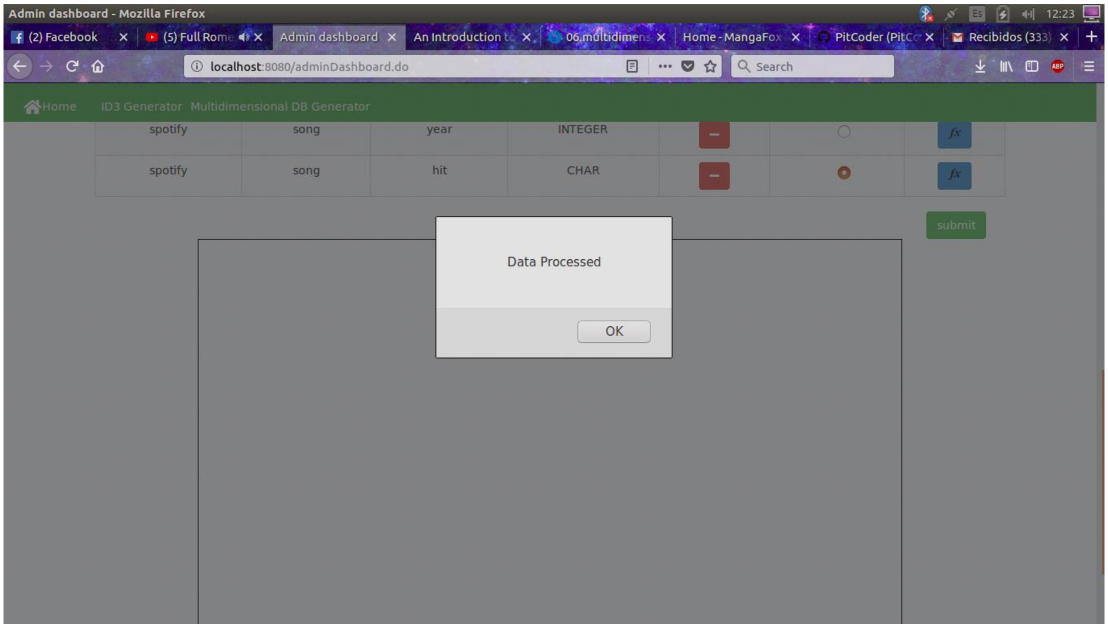
Database	Relation	Attribute	Data Type	Delete	Decision	F(x)
spotify	artist	idregion	INTEGER	−	○	fx
spotify	genre	name	VARCHAR	−	○	fx
spotify	song	year	INTEGER	−	○	fx
spotify	song	hit	CHAR	−	●	fx

submit

```
└── genre.name
    ├── Pop
    │   └── song.year
    │       └── 80's
    │           └── artist.idregion
    │               └── American
    │                   └── Y
    │
    ├── Rock
```

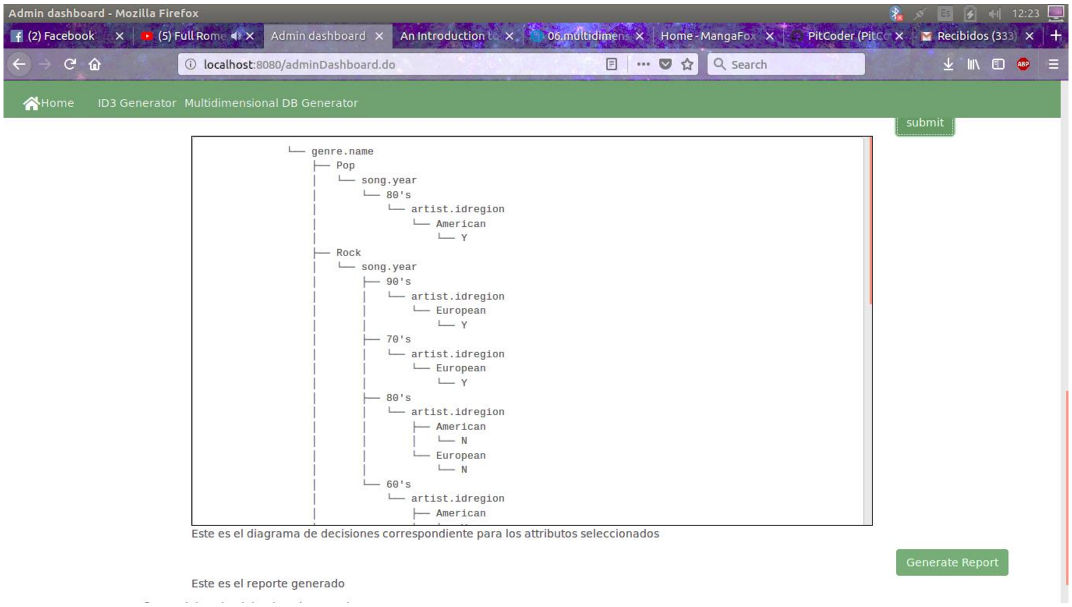
Then the user must choose which of the attributes added will be our class (decision). Once the user has clicked one of the radio buttons, summit button will be available to the user, so now we can generate the ID3 tree decision.

The user must press the summit button in order to send the information to the server and generate the tree. The once the server has finished processing the information, the following image will be displayed, along with the graphical representation of the tree.



spotify	song	year	INTEGER	−	○	fx
spotify	song	hit	CHAR	−	●	fx

Data Processed

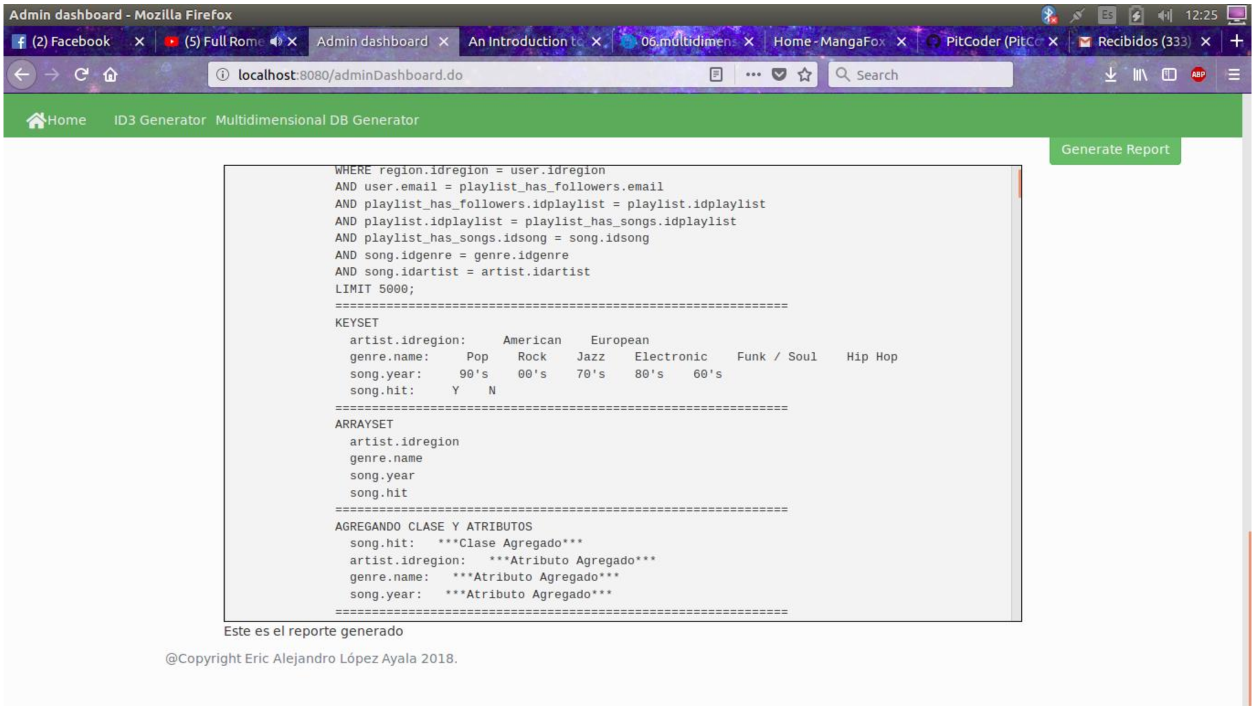


```
          └── genre.name
              ├── Pop
              │   └── song.year
              │       └── 80's
              │           └── artist.idregion
              │               └── American
              │                   └── Y
              │
              ├── Rock
              │   └── song.year
              │       ├── 90's
              │       │   └── artist.idregion
              │       │       └── European
              │       │           └── Y
              │       │
              │       ├── 70's
              │       │   └── artist.idregion
              │       │       └── European
              │       │           └── Y
              │       │
              │       ├── 80's
              │       │   └── artist.idregion
              │       │       ├── American
              │       │       │   └── N
              │       │       └── European
              │       │           └── N
              │       │
              │       └── 60's
              │           └── artist.idregion
              │               ├── American
```

Este es el diagrama de decisiones correspondiente para los attributos seleccionados

Este es el reporte generado

Finally, the user has the option to generate a more detailed report of the ID3 generation process, for this it has to press the “Generate Report” button, all generated data during ID3 creation will be send from the server and it will be displayed on a log.



```
               WHERE region.idregion = user.idregion
               AND user.email = playlist_has_followers.email
               AND playlist_has_followers.idplaylist = playlist.idplaylist
               AND playlist.idplaylist = playlist_has_songs.idplaylist
               AND playlist_has_songs.idsong = song.idsong
               AND song.idgenre = genre.idgenre
               AND song.idartist = artist.idartist
               LIMIT 5000;
               ==============================================================
               KEYSET
                 artist.idregion:     American    European
                 genre.name:     Pop    Rock    Jazz    Electronic    Funk / Soul    Hip Hop
                 song.year:     90's    00's    70's    80's    60's
                 song.hit:     Y    N
               ==============================================================
               ARRAYSET
                 artist.idregion
                 genre.name
                 song.year
                 song.hit
               ==============================================================
               AGREGANDO CLASE Y ATRIBUTOS
                 song.hit:   ***Clase Agregado***
                 artist.idregion:   ***Atributo Agregado***
                 genre.name:   ***Atributo Agregado***
                 song.year:   ***Atributo Agregado***
               ==============================================================
```

Este es el reporte generado

@Copyright Eric Alejandro López Ayala 2018.

REFERENCES

Kurt Thearling, “An Introduction to Data Mining”, a paper published in “Data Mining-Vol-1”, ICFAI [2002].

[2] http://www.bandmservices.com/DecisionTrees/Decision _trees.htm

[3] Quinlan, J.R. [1986], Introduction of decision trees, “Machine Learning”.

[4] H.Hamilton. E. Gurak, L. Findlater W. Olive, “Overview of Decision Trees” as published on the website http://www.cs.uregina.cd/~dbd/cs831/notes/ml/dtrees/4_ dtrees1.html

[5] http://en.wikipedia.org/wiki.ID3_algorithm#algorithm

[6] Nagabhushana, S., [2006] “Data Warehousing-OLAP and Data Mining”, New Age International Publishers.

[7] Chaudhari, S., and Dayal, U. [1997] “An Overview of Data Warehousing and OLAP Technology”,SIGMOD Record, Vol.26, No. 1, March 1997. [8] Breiman,L., Friedman,J.H., Olsen,R.A., and Stone,C.J.(1984) “Classification and Regression Trees”,Belmont,C.A. : Wadsworth Statistical Press.