[[1]](#footnote-1)

Smart Cities: Performance analysis of small towns to make them smarter.

Mateus da Rosa Zanatta., *Student, UFSM*, Michael Cassens, *Adviser, UM*, and Bertolini Cristiano, Co-Adviser, UFSM

*Abstract*—TODO

*Index Terms*—TODO

# INTRODUCTION

Nowadays many governmental institutions in Brazil are opening their non-sensitive databases to research, willing citizenries might analyze the released data. Thus, citizens would be able to acknowledge the country situation in different government department. However many times these data are not well organized, having tables without relationships and/or with fragmented/non-concise data. Hence making difficult to interpret what is inside of the set of data.

In this way the present article will use databases related to Frederico Westphalen micro region. Which nowadays, according to last demographic census performed in 2010, has 30.558 inhabitants [7]. Being the biggest city among the 27 cities in a region with 174.605 inhabitants [8]. The region is located in the interior of southern region.

The databases from cities included in this region were released by Fundação de Economia e Estatística (FEE), and the salary average where released by the Instituto Brasileiro de Geografia e Estatistica (IBGE). Allowing the creation of a ranking that will show how different these cities are. Also it will demonstrate to authorities what characteristics they should improve to have a better environment.

Also, it was created a Facebook App to allow people to sign up in the website and provide their timeline posts to be analyzed by this application. When gathering the data it was defined several hashtags to people use in their post, and make easier to the application to identify complaints about cities people are living in. In addition this application is also gathering together posts from Facebook pages of 13 cities, because not all cities have a page, allowing to in the future perform data mining operations to find useful information

To develop the logic of the application it will be utilized Laravel Framework, a PHP framework. Aiming a quicker server-side development and integration with database. As the Database Management System (DBM) it will be worked with MongoDB. Being chosen because it has a document oriented storage, allowing to work with data in a more generic way and faster querying routines than relational databases.

# Objective

This project wanted to gather open databases from the cities belonging to the Frederico Westphalen micro region and implement methods to rank and measure their performance. Also gathering data from Facebook to identify people’s complaints.

# Goals

- Identified open databases which contain relevant data;

- Imported all relevant open database to MongoDB;

- Studied and applied technique to develop ranking and indices;

- Created a ranking to compare the performance of each city;

- Enabled the visualization of strengths and weaknesses of each city;

- Utilized a non-relational database;

- Created dashboard to facilitate the visualization of the generated information;

- Utilized HTML 5 and Javascript to develop the dashboard;

- Utilized the framework PHP Laravel to process the obtained information;

- Utilized Facebook to gather people thoughts about their city;

# Related work

This project was based on the European smart cities project created by Vienna University of Technolohy targeting medium-sized European cities with the objective to provide a mean of learning specific fields of urban development [9]. Where were selected 77 cities with an urban population between 100.000 and 500.000, at least 1 university and catchment are less than 1.500.000 inhabitants [9].

In contrast, this project selected 27 cities with an urban population between 1.500 and 29.000 inhabitants. Aiming that common people and those belonging to local administration can visualize how their city is performing among 6 key fields, allowing people to think over about its situation. In addition to results acquired from the public databases it was also created a section to gather information from Facebook, where registered users would allow this application to retrieve their posts on Facebook and store specific hashtags.

# Background

Traditionally it is utilized relational databases, but in 1998 Carlo Strozzi introduced the term NoSQL, what originally stated to a relational database that dispensed SQL interface, using UNIX commands to read and write data [1]. Nowadays NoSQL (Not only SQL) can be defined as a database class which does not make use of relation concepts applied in relational databases. Furthermore it allows faster reading and writing operation, yet it facilitate scalability, clustering and failure tolerance [2].

Document oriented databases store document collections, where each document means objects that have an identifier (ID) and group of fields as strings, lists or nested documents [3]. Also, documented oriented databases are free of schemas and allow the storage of complex data structures, for instance trees, dictionaries and collections [4].

As oriented document database examples it is possible to quote CouchDB and MongoDB. Where CouchDB uses JSON format, is JAVA implemented and allows replication and consistency. Otherwise MongoDB is C++ implemented and permit concurrency and replication [3].

When developing software it is necessary to choose some tools to start coding, usually it is chosen tools with good documentation, permit a quick development and easy to work with. So, choosing a framework will allow to abstract some coding and save time. To this work it was chosen the Laravel framework, an open-source PHP framework which follows model-view-controller (MVC) pattern and provide several ways to access databases.

To visualize the generated information there are several tools as well as JQuery Visualize, jqPlot, D3.js which is a JavaScript library that uses HTML, SVG and CSS to render diagrams and charts, jpGraph is a PHP library to create plots on server-side and Google Charts which is a tool highly flexible and a complete documentation [5].

Once it is needed to collect data from Facebook it was necessary to use its Application-Programming Interface (API), which is “a set of programming instructions and standards for accessing a Web-Based software application or Web Tool” [10]. Being the Facebook Graph API “the primary way to get data in and out of Facebook’s platform” [11]. This API allows the application to make HTTP requests to retrieve data from Facebook to those who registered an APP. It will generate a couple of secure codes, such as app secret number, APP ID, and an access token which should be sent with the data request.

# Methodology

The project is based on the idea of Smart Cities, where open databases related to health care, education and government are gathered together. To perform this project were selected cities which are included in the Frederico Westphalen micro region. And imported their open databases into a non-relational database, MongoDB. Thus, loading the data into MongoDB allowed us to create queries that returned information. Also, a framework PHP called Laravel, HTML 5 and Javascript were utilized to do the information processing and show the information got from the MongoDB queries. Then, it was possible to create a ranking, which compares the cities performance, and visualize the cities strengths and weaknesses.

Based on the European smart cities project [9], after gathering together the data it was identified 29 fields to be used on this project. These fields, here called domains, where aggregated into 6 key fields: environment, health, economy, government expenditures, employment, and education.

TABLE I

KEY FIELDS AND THEIR DOMAINS

|  |  |  |
| --- | --- | --- |
| 1. Meio ambiente (Environment) | Impact | |
| * 1. Destino do lixo (Garbage disposal) |  |
| * + 1. Outro (Other) | -1 | |
| * + 1. Jogado (Dumped) | -1 | |
| * + 1. Enterrado (Buried) | -1 | |
| * + 1. Queimado (Incinerated) | -1 | |
| * + 1. Coletado (Collected) | 0 | |
| * 1. Esgotamento sanitário (Sewage) |  | |
| * + 1. Fossa rudimentar (Cesspit) | -1 | |
| * + 1. Fossa séptica (Septic tank) | 0 | |
| * + 1. Outro (Other) | -1 | |
| * + 1. Rede pluvial (Sewerage system) | 0 | |
| * + 1. Rio, lago ou mar (River, lake or sea) | -1 | |
| * + 1. Vala (Drainage ditch) | -1 | |
| 1. Saúde (Health) |  | |
| * 1. Esgotamento sanitário (Sewage) |  | |
| * + 1. Vala (Drainage ditch) | -1 | |
| * + 1. Rio, lago ou mar (River, lake or sea) | -1 | |
| * + 1. Fossa séptica (Septic tank) | 0 | |
| * 1. Abastecimento de água (Water supply) |  | |
| * + 1. Outra (Other) | -1 | |
| * + 1. Poço ou Nascente (Well or spring) | -1 | |
| * + 1. Rede Geral (Public water supply) | 0 | |
| 1. Economia (Economy) |  | |
| * 1. Empresas (Companies) |  | |
| * + 1. Número de estabelecimentos (Number of companies) | 0 | |
| * 1. Contabilidade Social |  |
| * + 1. PIB (Citie's GDP) | 0 |
| * + 1. Valor adicionado bruto (GVA) | 0 |
| 1. Finanças públicas (Government expenditures) |  |
| * + 1. Tributos municipais (Municipal taxes) | 0 |
| * + 1. Tributos federais (Federal taxes) | 0 |
| * + 1. Tributos estaduais (State taxes) | 0 |
| * + 1. Despesas realizadas (Government expending) | 0 |
| 1. Emprego (Employment) |  |
| * + 1. Número vinculos empregatícios (Employment) | 0 |
| * + 1. Remuneração média (Average salary) | 0 |
| 1. Educação (Education) |  |
| * 1. Educação infantil (Preschool) |  |
| * + 1. Matrículas (Enrollments) | 0 |
| * 1. Ensino Fundamental (Elementary school and Middle school) |  |
| * + 1. Taxa de aprovação (Pass rate) | 0 |
| * + 1. Taxa de reprovação (Failure rate) | -1 |
| * + 1. Taxa de distorção idade série (Distortion rate age-class) | -1 |
| * + 1. Taxa de abandono (Dropout rate) | -1 |
| * + 1. Número de concluintes (Number of graduates) | 0 |
| * 1. Ensino Médio (High school) |  |
| * + 1. Taxa de aprovação (Pass rate) | 0 |
| * + 1. Taxa de reprovação (Failure rate) | -1 |
| * + 1. Taxa de distorção idade série (Distortion rate age-class) | -1 |
| * + 1. Taxa de abandono (Dropout rate) | -1 |
| * + 1. Número de concluintes (Number of graduates) | 0 |

Because the selected cities have different population size it was necessary to find a method to standardize the data. Then, it was decided to apply the z-transform (standard scores) method, which “transform all indicator values into standardized values with an average equals to 0 and a standard deviation equals to 1. This method has the advantages to consider the heterogeneity within groups and maintain its metric information.” [6]. Allowing the visualization of the average movement, showing when the cities are below or above the average. Thus when the z-score acquired is a negative value it means the city is below the average otherwise it is above the average. In case of a z-score equals to 0 (zero) it means the city is in the average.

As an example of the formula it is possible to consider a set of numbers with a standard deviation and an average. Therefore resulting in the following for the first:

Once it was obtained the standardized values it was possible to create a ranking with the selected cities. Being decided to use the Dense ranking method. In this type of ranking when items are compared as equals they will receive the same ranking position, otherwise it will be assigned the next position to the lower value.

Because some domains will make a negative impact their values where multiplied by -1, as it is possible to see on Table 1. Therefore the z-score will be inverted making a negative impact in the ranking. For instance, when the domain cesspit has a positive value, indicating that there are many occurrences of cesspit on determined city, thus it means there is a negative impact in the environment. So, it is needed to invert this value to decrease the city points.

In addition it was created an app on Facebook to allow the application to gather data from people signed in. Thus, the application allow people to use hashtags in their posts which permit it to identify people thoughts about their city. The hashtags were defined as a compound, where the first hashtag is the city’s name and the second one is an issue that can be found in the city.

As it is possible to see on Table II some cities have more than one hashtag. It happens because either it is possible to find a couple of different spellings or the city has an abbreviation.

TABLE II

CITIES’ HASHTAGS

|  |  |
| --- | --- |
| City | Allowed Hashtags |
| Alpestre | #Alpestre |
| Ametista do Sul | #AmetistadoSul, #AmetistaSul |
| Caiçara | #Caiçara, #Caicara |
| Constantina | #Constantina |
| Cristal do Sul | #CristaldoSul, #CristalSul |
| Dois Irmãos das Missões | #DoisIrmãosdasMissões, #DoisIrmaosdasMissoes, #DoisIrmaosMissoes |
| Engenho Velho | #EngenhoVelho |
| Erval Seco | #ErvalSeco |
| Frederico Westphalen | #FW, #FredWest, #FredericoWestphalen |
| Gramado dos Loureiros | #GramadodosLoureiros, #GramadoLoureiros |
| Iraí | #Iraí, #Irai |
| Liberato Salzano | #LiberatoSalzano |
| Nonoai | #Nonoai |
| Novo Tiradentes | #NovoTiradentes |
| Novo Xingu | #NovoXingu |
| Palmitinho | #Palmitinho |
| Pinheirinho do Vale | #PinheirinhodoVale, #PinheirinhoVale |
| Planalto | #Planalto |
| Rio dos Índios | #RiodosÍndios, #RiodosIndios, #RioÍndios, #RioIndios |
| Rodeio Bonito | #RodeioBonito |
| Rondinha | #Rondinha |
| Seberi | #Seberi |
| Taquaruçu do Sul | #TaquaruçudoSul, #TaquarucudoSul, #TaquaruçuSul, #TaquarucuSul" |
| Três Palmeiras | #TrêsPalmeiras, #TresPalmeiras |
| Trindade do Sul | #TrindadedoSul, #TrindadeSul |
| Vicente Dutra | #VicenteDutra, #ViDu |
| Vista Alegre | #Vista Alegre |

Using the hashtags on Table II is possible to identify which city the user is talking about. Additionally they should use any other hashtag listed on Table III. As well as it happens with cities’ hashtags there are some spelling variations inasmuch as synonyms.

TABLE III

COMPLAINT’S HASHTAGS

|  |
| --- |
| Allowed Hashtags |
| #pavimento |
| #transito, #trânsito |
| #poluicao, #poluição, #poluiçao |
| #estacionamento |
| #calcada, #calçada, #calçadas, #calcadas, #passeio, #passeios |
| #saude, #saúde |
| #atendimento, #funcionários, #funcionarios, #funcionarios |
| #emprego, #trabalho |
| #aluguel |
| #imoveis, #imóveis |
| #diversão, #diversao |
| #infraestrutura |
| #violencia, #violência |
| #drogas |
| #educacao, #educação |
| #impostos |
| #SaneamentoBásico, #SaneamentoBasico |
| #buracos, #buraco |
| #lixo |

# Solution

As already mentioned it was used the z-transform method to calculate standard scores for the key fields of each city. Being aware t would be accessed on several classes it was created one class to determine the standard scores. This class separated the z-transform method in three different function, one to calculate the standard deviation, other to determine the average and the function which would obtain the z-scores. Permitting the application to have access to its functions from anywhere.

This way it was possible to create classes for each one of the key fields specified in the Table 1. Which are requesting data from in the database and performing the z-transform calculation for each one of the domains described in the Table 1. The result of the calculations obtained from these classes are being used in a separated class to create a ranking based on the key fields’ z-scores.

The ranking class get the z-scores of each key field, multiply by -1 those domains which must make a negative impact on the raking, sum all of the scores, multiply them by 1000 and truncate them, allowing the application to compare the z-scores with 3 numbers of precision, and finally sorting them from the high value to low. Then the application apply the Dense ranking method and assign a position for each city.

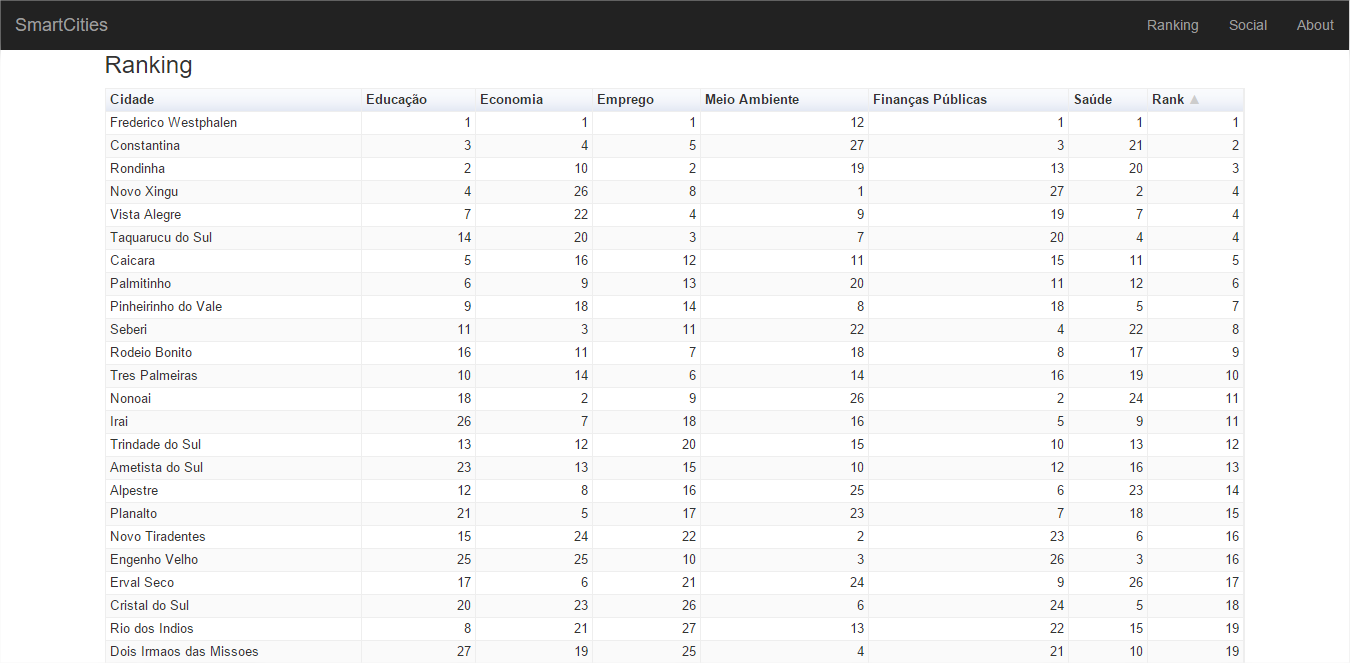


Fig. 1. The ranking generated after processing the gathered data

In addition, with the z-scores generated, it was possible to create several charts to show the cities’ performance in the region. Showing how far one city is from the region average. Once the region average is equal 0 (zero) it is possible to see the city’s accomplishments when compared with the region average. In other words, if the given city has a positive z-score it means it is making a positive impact in the region otherwise the impact is negative. Also, if the city’s z-score is equals 0 (zero) it means that the city is on the average. Being made 7 charts, one chart with the key field’s z-scores and 6 other charts for the domains of each key field.

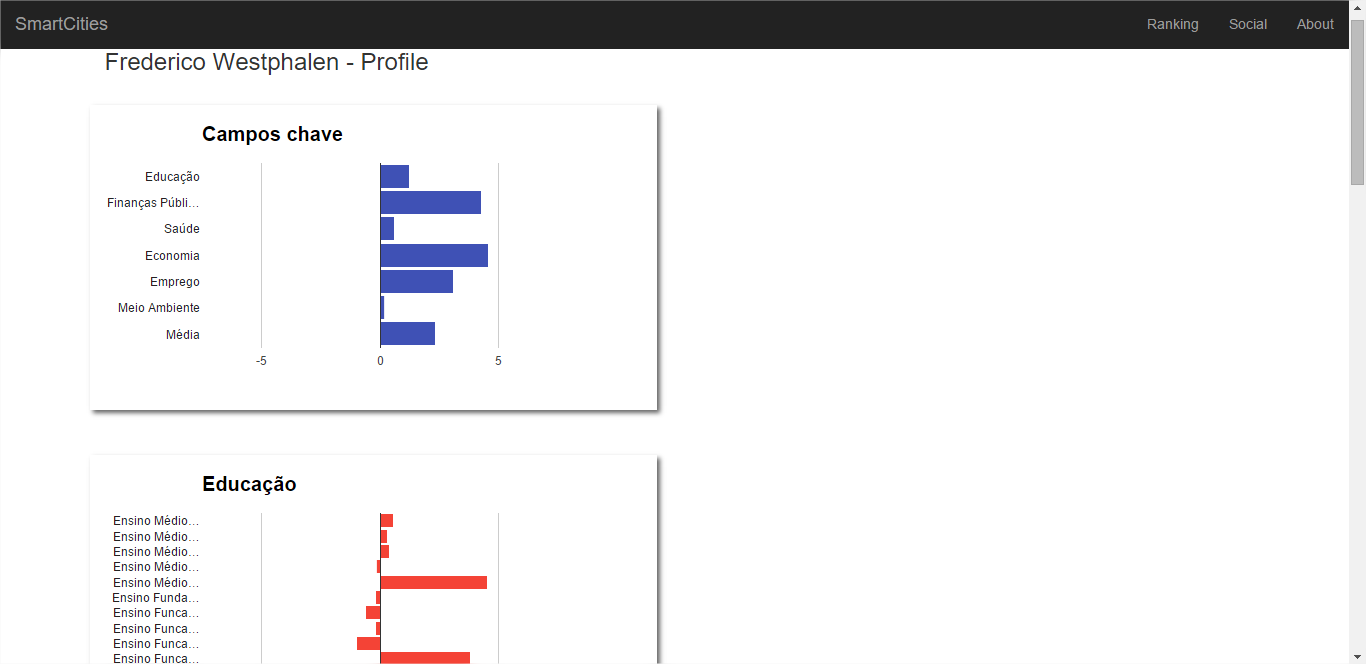


Fig. 2. Profile page

Furthermore it was created a Facebook App for collecting user and page posts, targeting hashtags which could be used to identify people complaints. This application will provide a button to allow the user to login on Facebook and provide their information, such as name, birthday, email, and an access token to allow the application to read their posts. This access token will allow the application to gather users post for approximately 60 days. After having the access token expired, the user should login on Facebook again, if they want to keep having their posts tracked. While the access token is valid, the application will, every night at midnight, perform a requisition to Facebook API requiring the user’s posts, reading them looking for a compound of hashtags. This compound of hashtags has to have at least one of cities’ hashtag and one of the previously defined complaint’s hashtag.

After collecting the data from Facebook it was possible to create a ranking with the top ten hashtags used in the region, the percentage of user per city, and the age range.



Fig. 3. Overview

This overview make possible show what people think about the region. Also it was created a page displaying a top ten ranking for each city as well as a chart with the percentage of each hashtag.



Fig. 4. City’s page

# Conclusion

TODO

References

1. LITT, Steve. NoSQL: The Unix Database (With awk). Available: <http://www.troubleshooters.com/lpm/200704/200704.htm>. Access in 08 Abr. 2014.
2. NASCIMENTO, Jean. NoSQL – você realmente sabe do que estamos falando?. Available:
3. <http://imasters.com.br/artigo/17043/bancodedados/nosql\_voce\_realment/>. Access in 22 Mar. 2014.
4. LÓSCIO, Bernadette Farias; OLIVEIRA, Hélio Rodrigues de; PONTES, Jonas César de Sousa. NoSQL no desenvolvimento de aplicações Web colaborativas. Available: < http://www.addlabs.uff.br/sbsc\_site/SBSC2011\_NoSQL.pdf >. Access in 24 Mar. 2014.
5. RAHIEN, Ayende. That NoSQL Thing – Document Databases. Disponível em: <http://ayende.com/blog/4459/that-no-sql-thing-document-databases>. Access in 22 Mar. 2014.
6. SUDA, Brian; HAMPTON-SMITH, Sam. The 36 best tools for data visualization. Available: <http://www.creativebloq.com/design-tools/data-visualization-712402>. Access in 30 Mar. 2014.
7. SMART CITIES. The Smart City Model. Available: <http://www.smart-cities.eu/index.php?cid=2&ver=3>. Access in 30 Mar. 2014.
8. IBGE. Rio Grande do Sul, Frederico Westphalen. Available: <http://cod.ibge.gov.br/234ZE >. Access in 02 Dec. 2015.
9. IBGE. Tabela 608 – População residente, por stuação do domicílio e sexo. Available: <http://www.sidra.ibge.gov.br/bda/tabela/listabl.asp?z=cd&o=5&i=P&c=608>. Access in 02 Dec. 2015.
10. TUWIEN. Europeansmarcities. Available: <http://www.smart-cities.eu/?cid=-1&ver=4>. Access in 20 Jun. 2015.
11. HOW STUFF WORKS. What is an API?. Available: <http://money.howstuffworks.com/business-communications/how-to-leverage-an-api-for-conferencing1.htm>. Access in 03 Dec. 2015.
12. FACEBOOK. Graph API Overview. Available: <https://developers.facebook.com/docs/graph-api/overview/>. Access in 03 Dec. 2015.

1. [↑](#footnote-ref-1)