MINISTÉRIO DA CIÊNCIA, TECNOLOGIA, INOVAÇÕES E COMUNICAÇÕES
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

sid.inpe.br/mtc-m21c/2019/02.27.11.58-TDI

VGI PROTOCOL AND WEB SERVICE FOR HISTORICAL DATA MANAGEMENT

Rodrigo Monteiro Mariano

Master's Dissertation of the Graduate Course in Applied Computing, guided by Drs. Karine Reis Ferreira, and Luis Antonio Coelho Ferla, approved in March 11, 2019.

URL of the original document:
<http://urlib.net/8JMKD3MGP3W34R/3SR2Q3H>

INPE
São José dos Campos
2019

PUBLISHED BY:

Instituto Nacional de Pesquisas Espaciais - INPE
Gabinete do Diretor (GBDIR)
Serviço de Informação e Documentação (SESID)
CEP 12.227-010
São José dos Campos - SP - Brasil
Tel.:(012) 3208-6923/7348
E-mail: pubtc@inpe.br

MINISTÉRIO DA CIÊNCIA, TECNOLOGIA, INOVAÇÕES E COMUNICAÇÕES
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

sid.inpe.br/mtc-m21c/2019/02.27.11.58-TDI

VGI PROTOCOL AND WEB SERVICE FOR HISTORICAL DATA MANAGEMENT

Rodrigo Monteiro Mariano

Master's Dissertation of the Graduate Course in Applied Computing, guided by Drs. Karine Reis Ferreira, and Luis Antonio Coelho Ferla, approved in March 11, 2019.

URL of the original document:
<http://urlib.net/8JMKD3MGP3W34R/3SR2Q3H>

INPE
São José dos Campos
2019

Cataloging in Publication Data

Mariano, Rodrigo Monteiro.

M337v VGI protocol and web service for historical data management / Rodrigo Monteiro Mariano. – São José dos Campos : INPE, 2019.

xxii + 82 p. ; (sid.inpe.br/mtc-m21c/2019/02.27.11.58-TDI)

Dissertation (Master in Applied Computing) – Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2019.

Guiding : Drs. Karine Reis Ferreira, and Luis Antonio Coelho Ferla.

1. VGI 2. Web service 3. VGIMWS 4. Historical data 5. Pauliceia 2.0. I.Title.

CDU 681.324

Aluno (a): ***Rodrigo Monteiro Mariano***

Título: "VGI PROTOCOL AND WEB SERVICE FOR HISTORICAL DATA MANAGEMENT"

Aprovado (a) pela Banca Examinadora em cumprimento ao requisito exigido para obtenção do Título de *Mestre* em *Computação Aplicada*

Dr. Nandamudi Lankalapalli Vijaykumar

N. L. Vijaykumar

Presidente / INPE / SJCampos - SP

( ) Participação por Video - Conferência

(X) Aprovado ( ) Reprovado

Dra. Karine Reis Ferreira

Karine R Ferreira

Orientador(a) / INPE / São José dos Campos - SP

( ) Participação por Video - Conferência

(X) Aprovado ( ) Reprovado

Dr. Luis Antonio Coelho Ferla

Orientador(a) / UNIFESP / Guarulhos - SP

( ) Participação por Video - Conferência

( ) Aprovado ( ) Reprovado

Dr. Antônio Miguel Vieira Monteiro

Membro da Banca / INPE / SJCampos - SP

( ) Participação por Video - Conferência

(✓) Aprovado ( ) Reprovado

Dra. Silvana Philippi Camboim

N. L. Vijaykumar

Convidado(a) / UFPR / Curitiba - PR

(X) Participação por Video - Conferência

( ) Aprovado ( ) Reprovado

Este trabalho foi aprovado por:

( ) maioria simples

(X) unanimidade

São José dos Campos, 11 de março de 2019

“Tuus totus ego sum, et omnia mea tua sunt”.

Saint Louis-Marie Grignion de Montfort
in *“A Treatise on the True Devotion to the Blessed Virgin”*, 1712

*For **Jesus Christ**, Incarnate Wisdom, by the hands of the **Blessed Virgin**.*

ACKNOWLEDGEMENTS

First and foremost, I would like to declare my very large gratitude to God for the opportunity of receiving the grace of completing this Master's degree, and for His provision and mercy in my life. I would like to communicate my deep appreciation to the Blessed Virgin who never ceases to intercede for us, next to Her Divine Son. A special thanks to my supervisors, Karine Ferreira and Luis Ferla, for the opportunity of participating of the Pauliceia 2.0 project. I would also like to thank Yasmin Wassef for her help in developing of the VGI Protocol for the Pauliceia 2.0 project. I thank Vijay for his help in correcting the text. I would like to thank Carlos Noronha for his amazing help related to the Pauliceia 2.0 server. I thank the numbering team of the Pauliceia 2.0 project, who helps me a lot to write the Chapter 5. I would like to thank Miguel Monteiro for his assistance, useful critiques and advices. I thank CAPES for the two first months of student scholarship. I thank FUNCATE for the assistance for the GEOINFO congress in December 2017. My grateful thanks to FAPESP for granting student scholarship, regarding the grant #2017/03852-9. At last, I would like to thank my family and friends for their support and encouragement during my life.

ABSTRACT

Volunteered Geographic Information (VGI) is a version of crowdsourcing where volunteers produce, assemble and disseminate geographic information into websites. The citizens are encouraged to produce geographic information in web sites, as OpenStreetMap, using their own knowledge. VGI provides some advantages (e.g. generate detailed geographic information with low cost), nevertheless VGI data does not guarantee the quality. For that reason, there is a need to improve their quality. This is done through quality measures, indicators, approaches and the definition of a VGI protocol. The creation of a VGI protocol is important, because it drives the data collection of geographic data yielded by users. VGI protocol establishes a standardization for collaborative projects, improving the quality of citizen-derived geographical data sets and helping in the reuse of the protocol for other applications. It provides a definition of the processes that the citizen can do, right from the initialization in VGI platform, the description of the data model (as type of collected data and type of users), the methods of data collection, quality control, until the feedback from/to the users. Pauliceia 2.0 is a project whose aim is to develop a computational platform for manipulation of historical data collaboratively. Researchers contribute with these data and assist in quality control. In the Pauliceia 2.0 project, VGI is used to gather and share historical data resulted from researches provided by historians; and to produce historical geographical data sets of São Paulo city from 1870 to 1940. It is also used to collect other historical data, as the manual vectorization of ancient maps, collection of old addresses and the acquisition of historical photos associated to places. The main objective of this work is to define a VGI protocol for historical data and build a VGI Management Web Service (VGIMWS) based on the defined protocol, in the context of Pauliceia 2.0 project. This document presents a literature review, a stable version of the VGI protocol and of the web service for historical data. The protocol and VGIMWS were designed and built in the context of Pauliceia 2.0 project, however they are generic for historical data. So, they can be applied to other collaborative historical projects.

Keywords: VGI. Web Service. VGIMWS. Historical Data. Pauliceia 2.0.

PROTOCOLO E SERVIÇO WEB VGI PARA O GERENCIAMENTO DE DADOS HISTÓRICOS

RESUMO

Informação Geográfica Voluntária (VGI) é uma versão de crowdsourcing onde os voluntários produzem, reúnem e disseminam informações geográficas em sites na web. Os cidadãos são encorajados a produzir informações geográficas em sites, como o OpenStreetMap, usando seus próprios conhecimentos. O VGI provê algumas vantagens (e.g. gerar informações geográficas detalhadas com baixo custo), todavia os dados do VGI não garantem a qualidade. Por esse motivo, há a necessidade de melhorar sua qualidade. Isso é feito através de métricas de qualidade, indicadores, abordagens e a definição de um protocolo VGI. A criação de um protocolo VGI é importante, porque dirige a coleta de dados geográficos fornecidos por usuários. O protocolo VGI estabelece uma padronização para projetos colaborativos, melhorando a qualidade dos conjuntos de dados geográficos derivados por cidadãos e auxiliando na reutilização do protocolo para outras aplicações. Ele fornece uma definição dos processos que o cidadão pode fazer, desde a inicialização na plataforma VGI, a descrição do modelo de dados (como tipo de dados coletados e tipo de usuários), os métodos de coleta de dados, controle de qualidade, até o feedback de/para os usuários. Pauliceia 2.0 é um projeto cujo objetivo é desenvolver uma plataforma computacional para manipulação de dados históricos de forma colaborativa. Os pesquisadores contribuem com esses dados e ajudarão no controle de qualidade. No projeto Pauliceia 2.0, o VGI é usado para coletar e compartilhar dados históricos resultantes de trabalhos de pesquisa fornecidos por historiadores; e para produzir conjuntos de dados geográficos históricos da cidade de São Paulo de 1870 à 1940. É utilizado também para coletar outros dados históricos, como a vetorização manual de mapas antigos, coleta de endereços antigos e a aquisição de fotos históricas associadas a lugares. O objetivo principal deste trabalho é definir um protocolo VGI para dados históricos e construir um Serviço Web de Gerenciamento de VGI (VGIMWS) baseado no protocolo definido, no contexto do projeto Pauliceia 2.0. Este documento apresenta uma revisão da literatura, uma versão estável do protocolo VGI e do serviço web para dados históricos. O protocolo e o VGIMWS foram projetados e construídos no contexto do projeto Pauliceia 2.0, mas eles são genéricos para dados históricos. Então, eles podem ser aplicados a outros projetos históricos colaborativos.

Palavras-chave: VGI. Serviço Web. VGIMWS. Dados Históricos. Pauliceia 2.0.

LIST OF FIGURES

LIST OF TABLES

Page

LIST OF ABBREVIATIONS

API	– Application Programming Interface
CRS	– Coordinate Reference System
CSCs	– Commercial Surveying Companies
EPSG	– European Petroleum Survey Group
GIS	– Geographic Information System
HistMapathon	– Historic Mapping Marathon
HTTP	– Hypertext Transfer Protocol
ISO	– International Organization for Standardization
JSON	– JavaScript Object Notation
NMAs	– National Mapping Agencies
NSDI	– National Spatial Data Infrastructure
OGC	– Open Geospatial Consortium
OSM	– OpenStreetMap
REST	– Representational State Transfer
SDI	– Spatial Data Infrastructure
SOAP	– Simple Object Access Protocol
UFPR	– Federal University of Paraná
URI	– Uniform Resource Identifier
VGI	– Volunteered Geographic Information
VGIMWS	– Volunteered Geographic Information Management Web Service
WGS	– World Geodetic System
WSDL	– Web Service Definition Language
XML	– eXtensible Markup Language

CONTENTS

Page

1 INTRODUCTION

Volunteered Geographic Information (VGI), citizen-contributed geographic information, collaboratively contributed geospatial information, collaborative mapping and science 2.0 are terms used to express the general subject of collaborative work and citizen-derived geographical information. The work of See et al. (2016) describes a revision of these terms, giving explanations and showing key issues in the current state of this subject. The terms are categorized regarding the following main points: (1) information or process that can be used to generate it; (2) if the contributions are actives or passives; and (3) if the user-generated information are spatial or non-spatial.

VGI provides an alternative mechanism for acquisition and compilation of geographic information from volunteers, through of using of web tools to collect and process these data. (GOODCHILD, 2007). It was first defined by Goodchild (2007) as *"the harnessing of tools to create, assemble, and disseminate geographic data provided voluntarily by individuals"*. Goodchild and Li (2012) describe VGI as a version of crowdsourcing, with the target of handling geographic information. Estellés-Arolas and Guevara studied and produced a single definition of crowdsourcing, that is: *"a type of participative online activity in which an individual, an institution, a non-profit organization, or company proposes to a group of individuals of varying knowledge, heterogeneity, and number, via a flexible open call, the voluntary undertaking of a task. The undertaking of the task, of variable complexity and modularity, and in which the crowd should participate bringing their work, money, knowledge and/or experience, always entails mutual benefit. The user will receive the satisfaction of a given type of need, be it economic, social recognition, self-esteem, or the development of individual skills, while the crowdsourcer will obtain and utilize to their advantage what the user has brought to the venture, whose form will depend on the type of activity undertaken"* (ESTELLÉS-AROLAS; GONZÁLEZ-LADRÓN-DE-GUEVARA, 2012). While the data collection on citizen science projects can be conducted, for example, with printed forms, crowdsourcing, by nature, is online. This makes the crowdsourcing more restricted. This category of participation does not need to be open for all, it may be restricted only to certain groups (SEE et al., 2016).

There are several projects that work with VGI, for example: OpenStreetMap (OSM), Google Map Maker, Flickr and Wikimapia. OSM is the best-known VGI system (GOODCHILD; LI, 2012). OSM is an editable map of the entire world, created by untrained volunteers. It allows to work with free geographic data, having an open content license (OPENSTREETMAP, 2017b). OSM uses the local knowledge of the volunteers to build updated maps. It is open data, anyone can use the data of the OSM for any objective, since crediting the OSM and its users (OPENSTREETMAP, 2017a). Google Map Maker granted users to update maps data around the world, adding and managing most features on maps, as points of interest, parks, roads, lakes and so on. The volunteers could build maps in several countries, using their own expertise and local knowledge (KATRAGADDA; JAIN, 2008). Google Map Maker was closed in 2017 and its resources are being placed into Google Maps (GOOGLE, 2018). Flickr is a photo service that allow people to share their photos with other users (e.g. friends and family). People can annotate their photos using keywords, called tags, which describe or provide information about the photo (SIGURBJÖRNSSON; ZWOL, 2008). VGI is produced implicitly in Flickr, when the users annotate their photos with a geospatial reference, as giving the geographic coordinates or describing the geospatial attribute in textual form (SENARATNE et al., 2017). Wikimapia is an open collaborative map, where the users can insert place tags and share their local expertise. Its objective is to describe the entire world, compiling, organizing and providing free access to its geographical objects, using the user's experience (WIKIMAPIA, 2018). Wikimapia has a recognition system, that provide awards and a hierarchy of roles, regarding to the contributions made by volunteers (e.g. the role of Advanced User is given for a user who has good acknowledgment) (GOODCHILD; LI, 2012).

Mapping agencies use strict protocols to drive the geographic data collection, while the VGI projects normally have lack of specifications or only provide fuzzy instructions (MOONEY et al., 2016). For that reason, Mooney et al. (2016) propose a generic protocol that guides the vector data collection. This protocol is important, because it creates a standardization for the manipulation of vector data in VGI projects, improving the quality of citizen-derived geographical data sets amassed by a project and to assist the reuse of these data for other applications. The main stages of the VGI Protocol, proposed by Mooney et al., are: initialisation, data collection, self-assessment and quality control, data submission and feedback to the community.

1.1 Motivation

Pauliceia 2.0 project has as objective to design and build a computational platform to manage historical data collaboratively. It allows to select, collect, digitize and share historical data of the São Paulo city from 1870 to 1940. VGI is used by historians in the contribution of their historical data, provided by their own researches, and through the manual vectorization of the streets and buildings of the historical maps. The contributions are done through the concept of layers. The researchers are able to create thematic layers to save their research data. This data may be addresses, buildings, streets or other information related with their research. So, it is possible for the researchers to generate maps and views of their own research, contributing with the data inside the system.

There are two main user groups of the platform: historians, as explained before, and the citizens. While the historians contribute with their historical data provided by their researches, the citizens can contribute with their local knowledge. They can insert, for example, historical documents that they have (e.g. a 1930 photo), vectorize old maps or notify observations in the available layers.

The Pauliceia 2.0 team is composed by the following institutions: National Institute for Space Research (INPE), UNIFESP Guarulhos, Arquivo Público do Estado de São Paulo and Emory University, and it is funded by FAPESP. INPE did the research and development work of the Pauliceia 2.0 platform, where two scientific initiation (SI) students made the front-end, a SI student did the geocoding web service and the author of this work did the research, design and development of the VGI protocol for historical data and the VGI management web service. The UNIFESP team surveyed the first historical data of the platform, such as the vectorization of the central area of São Paulo and the collection of the historical addresses of this same region. In addition to that, they also publicized the platform together with the Arquivo Público team, while the Emory University team helped us in certain concepts of the Pauliceia 2.0 platform.

There are some historical projects with similar characteristics to Pauliceia 2.0. Regarding the OpenStreetMap, there is the OpenHistoricalMap, that uses the OSM infrastructure to create a map of the history of the world collaboratively (OPENSTREETMAP, 2018d). Other OSM project is the HistOSM[1], that is a system that uses the historical objects of the OSM for exploration. OSM provides a RESTful API that manages the raw geodata from/to OSM database using the XML format to exchange (OPENSTREETMAP, 2018a)(OPENSTREETMAP, 2018b). With this API, it is possible to link with the OpenHistoricalMap and to handle its historical data.

Building Inspector[2] is a project that manages historical map data. It uses the volun-

[1] http://histosm.org/

[2] http://buildinginspector.nypl.org/

teers to solve quality control tasks on web site. The project has an API[3] to exchange data, using GeoJSON. In ATLMaps[4] portal it is possible to overlay maps, to perform the visualization of geospatial data and points location; including annotations, images, audios and etc. (WHITE; GILBERT, 2016). At this moment just private people can contribute to the ATLMaps, as researchers or students of them.

1.2 Objective

The main objective of this work is to specify a VGI protocol for historical vector data collection in collaborative way and to develop a VGI Web Management Service (VGIMWS) based on this protocol, in the context of Pauliceia 2.0 project. These data contain geometries and attributes that describe historical places; for example, a historical building is represented by a polygon and its properties can be "number" and "demolished date". Pauliceia 2.0 VGI protocol defines a reasoning for the historians contribute into the project platform. VGIMWS was designed and built in order to manage the historical data, based on the Pauliceia 2.0 VGI protocol. The protocol and web service can be used by other historical data projects similar to Pauliceia 2.0.

The specific objectives are the following:

- to review VGI measures and strategies to evaluate the quality of collaborative data. So, analyzing what metrics would be feasible for the historical data of the Pauliceia 2.0 project;
- to plan HistMapathons. During a HistMapathon, users were able to contribute to the system by inserting historical addresses to improve the accuracy of the geocoding web service;
- to organize the manual testing phase of the platform. In this period, the concepts of VGI and other of the Pauliceia 2.0 platform were analyzed and tested.

Both during HistMapathons and the manual testing phase, problems and improvements were reported by users through the Pauliceia 2.0 platform e-mail list. After that, this feedback was evaluated and described in detail in a project wiki, so the developers could read and implement the appropriate solutions.

[3] http://buildinginspector.nypl.org/data

[4] https://atlmaps.org/

1.3 Contribution

The main contributions of this study are:

- VGI protocol for historical data, that are used by Pauliceia 2.0 final users;
- VGIMWS, that is a web service to handle spatiotemporal data, based on the above VGI protocol. Pauliceia 2.0 portal connects to this service, managing the historical data;
- "VGI Protocol and Web Service for Historical Data Management". An article, that is derivative from this work. This article was published on Proceedings XIX GEOINFO, December 05-07, 2018, Campina Grande, PB, Brazil, p 103-115. Available from http://mtc-m16c.sid.inpe.br/col/sid.inpe.br/mtc-m16c/2018/12.27.18.29/doc/p10.pdf;

2 LITERATURE REVIEW

2.1 Collaborative work and citizen-derived geographical information

Besides VGI, there are other terms that refer to collaborative work and citizen-derived geographical information, as crowdsourcing, Ambient Geographic Information, Citizen-contributed Geographic Information, Collaborative mapping and Collaboratively Contributed Geospatial Information (SEE et al., 2016).

VGI is the use of tools to produce, assemble and disseminate geographic information provided by volunteers into websites, providing an alternative mechanism for acquisition and compilation of geographic information. According to Goodchild, VGI is *"the widespread engagement of large numbers of private citizens, often with little in the way of formal qualifications, in the creation of geographic information. They are largely untrained and their actions are almost always voluntary, and the results may or may not be accurate. But collectively, they represent a dramatic innovation that will have profound impacts on geographic information systems (GIS) and on the discipline of geography and its relationship to the general public. VGI is a special case of the more general Web phenomenon of user generated content"* (GOODCHILD, 2007). In another publication, the authors describe VGI as being *"a version of crowdsourcing in which members of the general public create and contribute georeferenced facts about the Earth's surface and near-surface to websites where the facts are synthesized into databases"* (GOODCHILD; LI, 2012).

The contributions of VGI contain a geographical location and a description, with several attributes, recurring from that location (GOODCHILD; LI, 2012). The users are often untrained and in spite of their knowledge and background, they create geographic information on web platforms (e.g. OpenStreetMap, Flickr or Wikimapia) (SENARATNE et al., 2017). The representation of VGI data may be done by means of a point, line or polygon. Even with high potential, by acquiring geographic information quickly, detailed and with low cost, VGI by default does not provide quality assurance in its data (GOODCHILD; LI, 2012).

VGI can be described by the following types: (1) map, (2) image and (3) text. Map-based VGI is when the VGI sources consist of basic geometries (i.e. points, lines and polygons) to build a map. Some examples are: OpenStreetMap, Google Map Maker, Map Insight and Wikimapia. Image-based VGI is normally generated implicitly, when the users take photos of objects and they attach a geospatial reference to it. Some examples are: Instagram, Flickr and Panoramio. Text-based VGI, as

Image-based VGI, is commonly created implicitly. It is done when the volunteers give geographic information in texts, using handheld devices (e.g. smartphones or laptops). They are regularly microblogs. Some examples are: Reddit, Twitter or others blogs (SENARATNE et al., 2017).

The imprecision of VGI data is explained by the fact that humans express the geographical regions and their relations imprecisely, through vague concepts. This imprecision of location is not only due to the fact that geographical entities are of a continuous nature, but also because of the quality and limitations of spatial knowledge (HOLLENSTEIN; PURVES, 2010). To provide reliable services or extracting useful information from such data, requires contributions from at least a quality standard. Inaccurate information, whether malicious or not, can be minimized by using appropriate quality indicators and measures for these various VGI contributions (SENARATNE et al., 2017). In this case, the work of Senaratne et al. (2017) provide a detailed review of methods for evaluating the quality of VGI data.

National Mapping Agencies (NMAs) and Commercial Surveying Companies (CSCs) use sturdy protocols that guide the collection of geographic data. VGI projects often have lack of standards or only provide loose guidelines, rather than rigorous specifications. Despite of the VGI can theoretically achieve high quality standards without rigid protocols, its absence is often an important source of errors in the data, representing a barrier to its further diffusion and reuse (MOONEY et al., 2016).

The need of establishing standards and protocols for VGI projects is not new. Girres and Touya (2010) cite that the lack of specification is one of the key points that may lead into poor data quality. Moreover, heterogeneity of the contributions may be caused by the lack of specification in the data collection. According to Mooney et al. (2016) some researchers have warned about community and societal threats, posed by the lack of protocols and mentioned their relevance to VGI projects, suggesting the definition of protocols to ensure the high data quality. The protocols are important to facilitate and extend the reuse of VGI data, for purposes and applications different from what they were collected originally.

The main stages of the VGI Protocol proposed by Mooney et al. (2016) are: initialisation, data collection, self-assessment and quality control, data submission and feedback to the community. In Initialisation the user must understand the project specifications. In the Data Collection happens the planning of the data collection process. In Self-Assessment and Quality Control describes how the user will do the review of the data collected before be sent to the server. In the Data Submission, after the necessary revisions, the user will send the data collected, then it will make a final verification of the data. In Feedback to the Community describes the channels of discussion (mailing lists, social networks and so on) that the users can express their comments and observations.

2.2 Open Science

The main literature of this section is Schiermeier (2018), because it is a great and helpful article about Open Science.

With the advancement of science and the production of volumes of research data, it began a concern in conserving these data for future use. For this reason, funders of research projects are now asking for applicants to create a data-management plan to avoid data loss and offer help to other scientists on how to use the data in the future.

A data-management plan describes how researchers will administer their data over and after a project, and include building, distributing and conserving research data, such as texts, algorithms, images or software. A lot of funders are asking for grant candidates to supply data plans. The researchers will explain what data they will produce; how these data will be recorded, defined and protected; and who will be able to access these data after the project is finished. Data management is one method that research sponsors and research institutions are applying "open science", a effort to create scientific research and data accessible for anyone. Research communities have unlike habits and routines, because of that, data-management requirements can vary.

Stored research data needs to have appropriate metadata that define their origin and aim, for other people can reuse them. The data creator needs to inform who will take care of the information after the project finishes, that it should be to an office. When to compose a data-management plan it is needed to inform restrictions about data privacy and ethical aspects.

It is a good idea to determine early about the types and quantity of data the researcher will assemble and how to organize them. Thus, it will help researchers to prevent problems with data loss and diffusion.

At work of Wilkinson et al. (2016), they describe how it is important to enhance the reuse of scholarly data, proposing a set of principles called FAIR Data Principles. FAIR Principles was created based on a effort of a group of partners, such as representing academia, funding agencies, scholarly publishers and industry, to establish a guideline to improve the reusability of their data. Theses principles have as one objective to improve the intelligence of machines to automatically discover and consume the data, helping in reproducibility, reusability and transparency of that data.

The four FAIR Principles are Findability, Accessibility, Interoperability and Reusability. Findability is when the data and metadata are appointed like a unique identifier; when they are defined using abundant metadata and they must have an identifier; and when data and metadata are indexed in a searchable system. Accessibility means that the data and metadata are gettable by their identifier through an open free standardized communication protocol. Interoperability is when the data and metadata use a standard language for representation. Reusability means that the data and metadata are amply described by suitable properties; that they are published using an approachable data usage license and they are avaiable using domain-relevant community standards. A lot of repositories have already developed several characteristics of FAIR Principles through a various technology options. Examples of these applications are: Dataverse[1], FAIRDOM[2], ISA[3], Open PHACTS[4], wwPDB[5] and UniProt[6] (WILKINSON et al., 2016).

Lima et al. (2018) describe the creation propose of a Spatial Data Infrastructure (SDI) to be implemented at Federal University of Viçosa (UFV). A SDI is a set of technologies and deals among institutions in order to make available the spatial data access easily, helping geographic information sharing and interoperability. This SDI is composed by data built at academic through the Brazilian National Spatial Data Infrastructure (NSDI) standards. NSDI is the 6.666/08 decree instituted in Brazil in 2008. It aims to handle the creation, storage and share of spatial data in Brazil using standards and deals.

2.3 Pauliceia 2.0 project

Pauliceia 2.0 project aims to develop an online computational platform for collaborative management of historical data, which contains a geographic location (i.e. spatiotemporal data). The historians are able to analyze, organize and publish ur-

[1] https://dataverse.org/

[2] http://fair-dom.org/

[3] https://isa-tools.org/

[4] https://www.openphacts.org

[5] http://www.wwpdb.org/

[6] http://www.uniprot.org/

ban historical data sets. VGI is used by historians to produce and share historical geographic data (e.g. maps) and views of their own research. This is done through the manual vectorization of the streets and buildings from the historical maps on the Pauliceia 2.0 portal or uploading a well-defined geographic file format, contributing to the data within the system. Historical data sets consist essentially of points, lines and polygons, which respectively represent historical addresses, streets and buildings.

On Pauliceia 2.0 platform there are two main user groups that can contribute with historical data: historians and citizens. Historians will enter data from their researches, making them more widespread among researchers, the history community and citizens who would like to know the history of regions registered in the system. Citizens will be able to contribute historical data that they contain, such as historical documents (e.g. photos or videos) that they need to be stored in other platforms such as YouTube or Dropbox. In addition to that, they will be able to vectorize historical maps, such as streets, buildings or adding historical addresses based on the old books.

Pauliceia 2.0 portal supplies tools that allow users to create and modify spatial locations and boundaries of features using vector data types. All volunteers can also create and update attribute values associated to resources using basic data types, as textual or numerical, and apprise links to documents. At the moment, the Pauliceia 2.0 platform does not provide tools to edit and create raster data types.

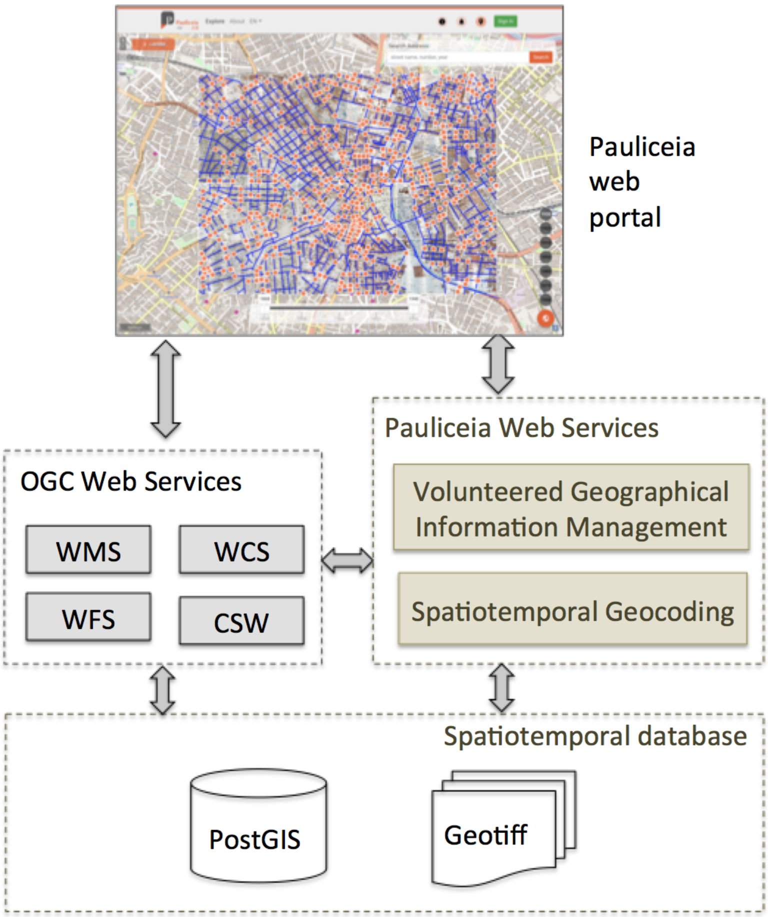
The project enriches the comprehension of the history of São Paulo (SP) city during the period from 1870 to 1940, that is the temporal scope of the Pauliceia 2.0. In addition to offer an innovative research model for the Digital Humanities, which promotes collaborative work and the free flow of knowledge. For that reason, the Pauliceia 2.0 project has a specific domain with a structured community, composed by historians and their history students.

The historical cut from 1870 to 1940 was chosen, because it was a time when the city of SP grew significantly, having a dramatic process of urbanization, leaving from approximately 30,000 inhabitants to 1,300,000 in about 70 years (SÃO PAULO. SECRETARIA MUNICIPAL DE URBANISMO E LICENCIAMENTO, 2017). The main reasons for this event are: the increased production of coffee, coming of immigrants, creation of railroads and industry development (MOTA et al., 2007) (CARVALHO, 2007).

2.3.1 Pauliceia 2.0 platform architecture

Pauliceia 2.0 system is open source, that is, the code is available online in the Github repository[7]. The platform is online and service oriented. The architecture that was developed is shown in Figure 2.1. Service-oriented architectures are well suited to provide a better interoperability among the systems (FERREIRA et al., 2017).

Figure 2.1 - Architecture proposed for the Pauliceia 2.0 project.



SOURCE: Ferreira et al. (2018).

The Pauliceia 2.0 architecture contains two groups of web services. The first one consists of geographical web services defined by the Open Geospatial Consortium (OGC), such as: Web Map Service (WMS), Web Feature Service (WFS), Web Coverage Service (WCS) and Catalogue Service Web (CSW) (OPEN GEOSPATIAL

[7] https://github.com/pauliceia

CONSORTIUM, 2019). The OGC services are provided by a GeoServer. OGC has performed a essencial role in geospatial data interoperability by introducing web services standards for showing, sharing and handling geospatial data. Use the web services proposed by OGC to distribute the Pauliceia 2.0 database is important for integration of the platform with other systems, interoperability and data dissemination (FERREIRA et al., 2017). Example of system integration could be the use of NSDI standards and interoperability with SDIs (LIMA et al., 2018).

The second group has two web services that were developed to serve to particular and essential demands of the Pauliceia 2.0 project, extending the functionalities of the OGC standard services. The first one is called "Volunteered Geographical Information Management Web Service", that provides all functionalities for handling citizen-derived historical information, such as: user control; management of spatiotemporal data sets, notifications and denunciations. The second one supplies a spatiotemporal geocoding method for historical data (FERREIRA et al., 2017). As work aim, it has been developed the first web service, that will be described afterwards.

The vector spatiotemporal data of the Pauliceia 2.0 project are stored in a PostgreSQL database with spatial extension PostGIS, while the raster data are saved in GeoTIFF format files (FERREIRA et al., 2017).

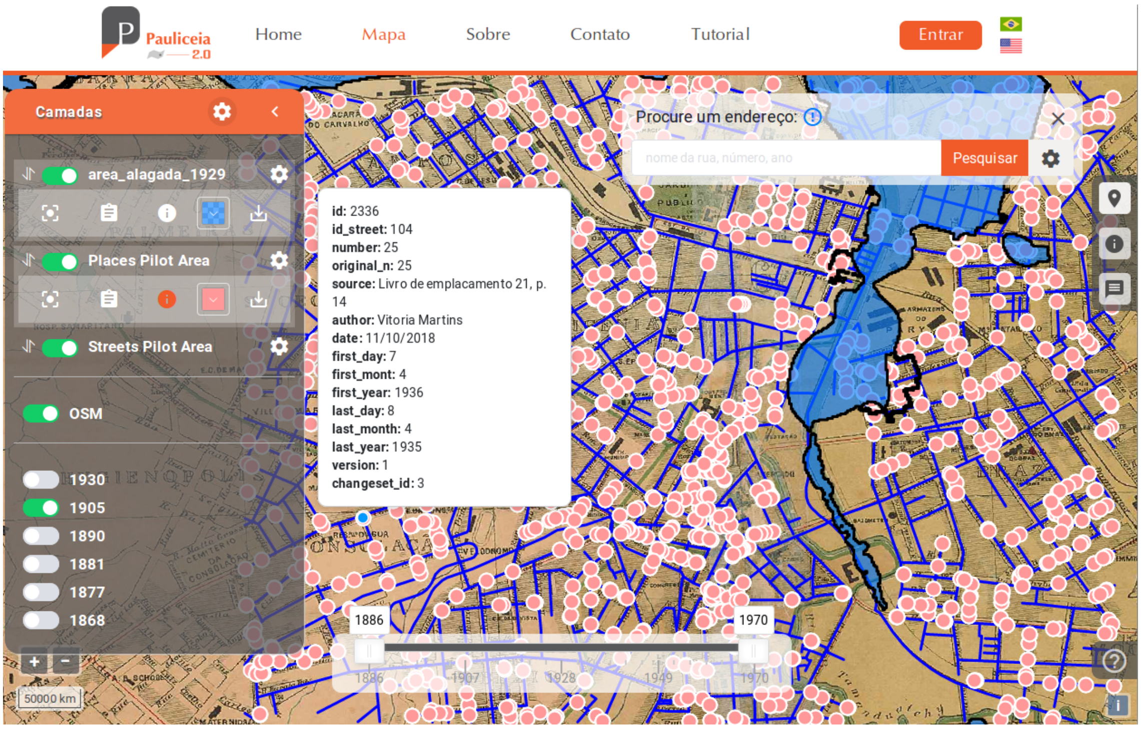
The entire Pauliceia 2.0 platform is divided into Docker containers, where each container performs a specific service. All project files, such as web portal, web services and spatiotemporal data, are saved to a server hosted by INPE. The focus of this work is in the VGI web service for historical data, in the context of Pauliceia 2.0 project.

Pauliceia 2.0 platform supplies a web portal, where the citizens can filter, visualize and download vector historical data sets; select historical maps; insert, update and delete historical data provide by their research; type notifications related to these data sets and write denunciations about bad data that the volunteers find on the platform.

2.3.2 Pauliceia 2.0 platform

Figure 2.2 shows the map page of the Pauliceia 2.0 platform and it is available on the following URL: `http://www.pauliceia.dpi.inpe.br`. Through this portal, project researchers or any other citizens can insert in the historical database spatiotemporal, data provided by their historical research about São Paulo city in the period from 1870 to 1940, such as houses, hotels, churches , factories, squares and so on.

Figure 2.2 - Map page of the Pauliceia 2.0 project.



id: 2336
id_street: 104
number: 25
original_n: 25
source: Livro de emplacamento 21, p. 14
author: Vitoria Martins
date: 11/10/2018
first_day: 7
first_mont: 4
first_year: 1936
last_day: 8
last_month: 4
last_year: 1935
version: 1
changeset_id: 3

SOURCE: Author's production.

The platform has a friendly and intuitive interface. Its maintenance and improvement are done according to the needs of the researchers involved with the Pauliceia 2.0 project. For better organization and control of the inserted data, the users need an authentication when starting their activities in the system. For this, three levels of users have been established: normal, curator and administrator. Normal ones can add or edit historical data, producing layers related to their historical research associated with keywords or references. This can be done manually or by bulk import. Curators are those who promote the collaboration of certain keywords and geographic space within the platform. They can update any user's layer information and organize them by creating new keywords for the layer. Administrators can insert, update and delete all entities of the Pauliceia 2.0 platform, also managing the denunciations made by users.

Figure 2.2 shows the map page of the Pauliceia 2.0 computing platform. In the background, it is shown the map of the central area of São Paulo of 1905 activated. When logged in, the user can insert layers of vector data, and in Figure shows three layers that were fed by some user, where: (1) 'area_alagada_1929' is a layer of polygons, which indicates the flooded area of São Paulo in 1929 (i.e. blue polygons); (2) 'Places Pilot Area' is a layer of points, illustrating the historical addresses of the central area of São Paulo in 1930 (i.e. pink points); and (3) 'Streets Pilot Area' is a layer of lines, which shows the streets of the central area of São Paulo in 1930 (i.e. blue lines).

When a layer is activated on the system, there is a gear next to the layer name where the user can: (1) zoom in on the entire layer by viewing it on the entire screen; (2) view the layer information; (3) get information from a specific feature; (4) edit the layer visualization (e.g. changing its colours); and (5) download vector data in the Shapefile format from that layer, where the user can open it again in some GIS (e.g. QGIS or TerraView).

In the right side menu, the user can search for some historical address, select information about a feature or a set of it, or the volunteer can view the notifications related to that layer. The user can search for some historical address using the standard "street, number, year" in the search bar, through the geocoding web service. The volunteer can also click on the gear next to the search bar and search for a list of historical addresses added within a CSV file.

At the bottom of the map is the Slider component, which is a spatiotemporal data viewer. It is used together to the map, having the function of filtering the time interval desired by the user, through an initial and final time defined in the portal. In this way, the users can choose the time interval that they want to work, not being limited to a single year. In addition, Slider lets the volunteer animates historical data over time.

2.4 Protocol for VGI projects

Mooney et al. (2016) propose a generic protocol, to be applied to several VGI projects. This protocol creates a standardization for projects that work with collaborative geographic data. This standardization can lead to an improvement in the quality of the contributions made by users. This is done because the quality of the VGI data is considered a barrier. This protocol intends to allow the inclusion of all volunteers of a VGI project, ranging from new users to experienced users. It is important to aid and broaden the reuse of VGI for other systems and goals than it was formerly created for. Because of that, ideas from this article were captured to be applied to the VGI protocol for historical data. Hence, their publication is the main literature for this section.

Mooney et al. (2016) introduce some topics of the protocol, developed by them, to achieve the objectives, standardization and finally the quality of projects based on VGI. These topics are: Data Model, Data Collection Methods and Vector Data Characteristics; thus, generating the protocol itself.

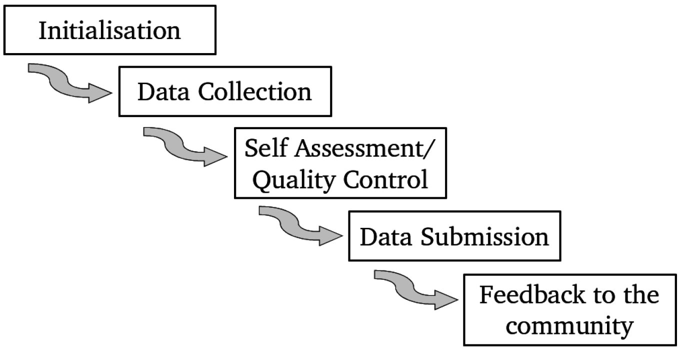
In Data Model, the VGI project should be presented in detail, explaining the motivation and objectives. This facilitates the contributor to understand why and how to collect the data. Regarding the contributed data, it is necessary to define which types of geometries will be used; its attributes and rules to ensure homogeneity (i.e. how one real world object should be represented). Examples of use cases should be included in the thematic layers. It is a good idea to encourage the contributors to become familiar with the service, before being enrolled in data collection. The contributors are encouraged to provide comments and observations.

Data Collection Methods describe how the collection will be done. One way is by manual vectorization, that is the acquisition of vector data from maps, aerial or satellite images. For example: a volunteer traces with a mouse on the resources shown on the computer screen, vectorizing them. This method is the most popular for this type of data acquisition. There is the field research, which is the vector data collection using equipment, such as Global Navigation Satellite System (GNSS), smartphones and so on. Bulk import is the integration of existing vector data inside the VGI project, through files.

Vector Data Characteristics detail the characteristics that are important and should be taken into account, such as Coordinate Reference Systems (CRS); topology and topological rules; level of detail or scale; metadata (e.g. source of data, resolution, date or time of scanning, comments about image quality, license and so on) and data quality.

A step-by-step, of how the data collection is done by the users is proposed by Mooney et al. (2016). These steps are the main stages of the VGI Protocol that is shown by the Figure 2.3. Its description is as follows: initialization, data collection, self-assessment/quality control, data submission and feedback to the community.

Figure 2.3 - Main steps of VGI Protocol



SOURCE: Mooney et al. (2016).

In Initialization the users must understand the project specifications, investing some time in a sample project to resolve questions before starting the real contribution. The contributors must verify if the collected data is appropriate for the project, in relation to the content and quality.

In Data Collection the volunteers must plan the data collection process, separating a part of their time for the data collection. They must have access to the project specifications to consult and make the data collection according to the instructions.

In Self-Assessment and Quality Control the contributors must review the collected data before being sent to the server. They should check if the data are adequate with respect to the geometric content, metadata and according to specifications. If they find errors, they should fix them.

In Data Submission the user will send the collected data, after the necessary revisions. Then, the volunteers are encouraged to do a final check of the data just submitted. The quality of the data is analyzed in relation to the coherence of the project. This final check is made in relation to the data that the users inserted on the platform or the data that were added by other contributors.

As a Feedback to the Community, the project should provide discussion channels, such as mailing lists, social networks and so on. Through these channels, the users can express their comments and observations.

2.5 Measures, indicators and approaches to the quality of the VGI

With VGI, it became possible for people to create their own digital geographic information through online maps, without any cost. This is done, because coordinates can be obtained with GPS or use images made available by third parties (e.g. Google Earth). Knowledge in cartography may be no longer necessary, because there are open-source software for building maps with high quality. A user can develop maps of their local area and may be more effective than a mapping specialist, due to their own local knowledge (GOODCHILD; LI, 2012). For that reason, it is important to study the quality and improvement of the data generated by VGI, which will be described later.

The quality of the VGI contributions can be described in essence by: quality measures and quality indicators. Quality measures use authoritative data (generated by reliable agencies, e.g. NMAs or CSCs) as a set of references, to evaluate the data generated by VGI, comparing them. This is done because it is believed that authoritative data always have high quality. Quality indicators are used when authoritative data are not available, which can be frequent, depending on the category of data (SENARATNE et al., 2017). In addition, Over et al. (2010) note that the quality of OSM data differs from authoritative data. The nature of volunteering may present trends in contributions. This is due to several factors, ranging from technical ability to cultural differences. As VGI data become more detailed over time and in certain areas, the use of authoritative data to evaluate the quality of VGI-generated data becomes less useful. This happens, because VGI data is often more complete and accurate than existing authoritative data sets (ANTONIOU; SKOPELITI, 2015).

In relation to quality measures for VGI, the International Organization for Standardization (ISO) has defined the quality of geographic information as: *"totality of characteristics of a product that bear on its ability to satisfy stated and implied needs"*[8]. There is a set of standards explained by ISO, which defines the quality measures of geographic information, such as (ISO, 2009) (JAKOBSSON; GIVERSEN, 2007):

- completeness: describes the presence and absence of data, its attributes and relationships between objects. To be evaluated, can use: comparison of number of characteristics, comparison of length or total area, or integrity index;

[8] https://www.iso.org/obp/ui/#iso:std:iso:19109:ed-1:v1:en

- consistency: level of coherence to logical rules of data structure (such as conceptual, logical or physical), assignment and relationships. In order to be evaluated, can use: semantic similarity between the tags, identification of entities with inadequate classification or system of recommendation of tags (algorithm that suggests relevant tags);

- positional accuracy: precision of the position of the resources. It is the closeness between a measure of a quantity and the accepted true value of it. To be evaluated can use: the euclidean distance of the point attributes or distances between centroid polygons;

- thematic accuracy: classification correction, correction of non-quantitative attributes and accuracy of quantitative attribute. To be evaluated can use: measurement of percentage (%) of the correct classification, confusion matrix or standard analysis of the kappa index;

- temporal accuracy: precision of a measure of time, temporal consistency and validity of data in relation to time. In order to be evaluated, can be studied the evolution of VGI data.

Regarding to quality indicators for VGI, they are qualitative (express data quality), such as purpose, usage and lineage. The purpose exposes the intention of the data set. Usage indicates what the functionality of the data. Lineage refers to the history of the data, since its collection, acquisition and derivation, until its final use. There are more abstract quality indicators to apply when the above are not applicable, as (SENARATNE et al., 2017):

- trustworthiness: a judgment based on subjective characteristics such as trust. Being acquired by good ratings of contributions or greater frequency of their use (FLANAGIN; METZGER, 2008);

- experience: user experience on the VGI platform. It can be captured when the volunteer registers in the portal, the amount of contributions made (both added or edited) or number of times that the user has used the online forums to discuss the data;

- credibility: defined as the credibility of a source or message, in which they have two dimensions: reliability and experience. To evaluate it, the source of the information is used as a basis, but it is not straightforward, because

VGI data are not authoritative, so the source may not be available. It is necessary to consider reliability and experience factors, in order to achieve this evaluation. The metadata about the origin of the VGI can provide a basis for this;

- text content quality: is the quality of text data based on the use of text characteristics, such as: length, structure, readability, revision history, use of specific terms and more. Usually applicable in text-based VGI;
- vagueness: ambiguity in data capture, such as: low resolution inaccuracy;
- local knowledge: users' knowledge of the geographic environment who they are mapping;
- recognition: provide rewards to the contributor for using the platform, such as virtual awards, known as Gamification, and opportunity to review their contributions by other users;
- reputation: is the ability to evaluate, mark, discuss and annotate the contributions, affecting the reputation of the user. Evaluation of the history of the interactions of the volunteer among the other collaborators.

According to Goodchild and Li (2012) there are still three approaches to ensure the quality of VGI: (1) crowdsourcing (generation of information by several people): it is the involvement of a group to validate and correct mistakes made by an individual collaborator; (2) social approaches: are reliable individuals who has a good reputation with their contributions to VGI, being able to act as *gatekeepers* to maintain and control the quality of other VGI contributions; and (3) geographic approaches: are the use of laws and knowledge of geography, as the first law of geography[9], to assess the quality.

Cechanowicz et al. (2016) and Hamari et al. (2014) recommend the use of Gamification techniques to encourage user participation, instigating them in data quality. Gamification is the use of game elements in contexts that are not games. It is a way to improve user engagement and motivation, increase the participation and provide a better experience. This can be used to get more than one contributor: more data, higher data quality, increased frequency and duration of participation (CECHANOWICZ et al., 2016). It is a process of enhancing services with stimulating features, using

[9] "Everything is related to everything else but closer things are more related than distant things" (TOBLER, 2004)

game experiences to get additional behavioral results. It is used to support the commitment of users and to improve the use of services. For example: increasing user activity, social interaction, quality and productivity of its actions (HAMARI et al., 2014).

2.6 Historical Data Projects

This section will describe the historical projects with similar characteristics to Pauliceia project.

ATLMaps[10] platform is a collaboration between Georgia State University and Emory University. Inside it, it is possible to combine maps, do the visualization of geospatial data and points location, contributed by volunteers to increase the knowledge about Atlanta. The contributors can create their own projects over the available layers, adding annotations, audios, images and so on. The users can superimpose maps from various times; for example, to compare the historical limits of the city in the time (WHITE; GILBERT, 2016). Currently the people who contributes to the content of ATLMaps are those involved in its development, researchers or students of them. It is not possible yet public or unknown contributors to insert data inside the platform. When is open submissions from a wider audience, the volume of material is expected to be low, for the administrators can manually review all new submissions, analyzing the quality and accuracy. It is not possible, at this time, to insert large quantities of materials.

Digital Harlem[11] is a research project about Harlem between 1915 and 1930, made by historians from the History Department of the University of Sydney, in Australia. It focuses on life of ordinary african new yorkers, capturing activities, places and relationships about the everyday life, through legal records and newspapers (DIGITAL HARLEM BLOG, 2017). The Digital Harlem was not developed with the intend of to be a public history project or a teaching resource (ROBERTSON, 2016). The site allows to search events and places, generate interactive maps, locate people in Harlem, to find places they often went and the results can be shown on a map with numerous layers.

British Library[12] contains a collection of maps available in a Online Gallery. The British Library allies with the Klokan Technologies to develop a Georeferencer sys-

[10] https://atlmaps.org/
[11] http://digitalharlem.org/
[12] https://www.bl.uk/maps

tem for crowdsourcing of gathering spatial metadata of historic mapping of Britain. The Georeferencer provides the metadata of the digitized collections, being its version 3.0 released in 2011. It represents advances in design, usability and crowdsourcing, as accessibility of the system, providing a simple process of contribution with quick results, having recognition and visible contribution (KOWAL; IDAL, 2012).

GeoHistoricalData[13] is a platform for collaborative digitization of various historical maps of Paris and France from the 18th century to the 20th century. This collaborative digitization is commonly done by the researchers for their immediate demands. The project has two main objetives: (1) investigate the territorial evolutions on different levels or scales; and (2) develop tools to answer with precision and flexibly these questions. Its datasets consist, in essence, of France from 1747 to 1950 and Paris from 1789 to 1950. The GeoHistoricalData is open source and its source code is available on GitHub[14].

GeoPeuple project has as goal to evaluate and vectorize old maps, and to develop a spatiotemporal database to analyze the densification of french territories. Its focus is on dissemination of the assembled data via an open web server. It grants an process of interactive spatiotemporal data analysis and a free access to the data. The GeoPeuple project has a RESTful Web API which allows the users to query the database and to give various management formats on the web. It uses PostgreSQL with PostGIS to store the historical data and the API is developed in Java (GROSSO et al., 2012).

2.6.1 Collaborative Historical Data Projects

This section presents historical projects that use VGI.

Building Inspector[15] is a project of the New York Public Library Laboratories in collaboration with Lionel Pincus and the Princess Firyal Map Division of the New York Public Library, made to manage historic map data. The computers are trained to do the heavy lifting and distribute the other quality control tasks to citizens, producing a detailed directory of the old New York. With this information it is possible to explore the city's past. The project allows to link historical documents (archives, old newspapers, photographs and so on) to places, giving the opportunity of new means to search, learn and discover the past. With respect to quality control,

[13] http://geohistoricaldata.org/
[14] https://github.com/GeoHistoricalData
[15] http://buildinginspector.nypl.org/

an algorithm proposed by Budig et al. (2016) is used for the extraction of the best polygonal representation, given several polygons, of constructions of the atlases of the nineteenth and early twentieth centuries. Given a set of polygons, which describe the same object (e.g. a building), the algorithm returns the best consensus. Internally the Building Inspector uses a API[16] to exchange data. The data is returned as a GeoJSON. This API returns the building footprints, colors present in a polygon, address numbers from a building, consolidated polygons, sheets, polygons from a sheet, place name, history, history for task and the place names with consensus.

OpenStreetMap provides a RESTful API to handle raw geodata from/to OSM database. The server works using HTTP request methods and the format of exchange is XML. For authentication OAuth[17] is used; however, read requests do not need authorization. The API is accessed from: `https://api.openstreetmap.org/`. When the developer wants to test his/her application, he/she must use the API (`https://master.apis.dev.openstreetmap.org/`) (OPENSTREETMAP, 2018a)(OPENSTREETMAP, 2018b) tester. This API can be used in programming languages using its OSM packages (e.g. Osmapi for Python). In Osmapi[18] it is possible to connect with OSM API passing the above URLs.

OpenHistoricalMap[19] (OHM) is a project that uses the OSM infrastructure to create a universal and detailed map of the world history, in a collaborative way. The user can enter data for margins, political boundaries, buildings and paths, similar to the OSM (OPENSTREETMAP, 2018d). Like the OSM, there is a RESTful API for the OHM live on `http://www.openhistoricalmap.org`. Using the same programming language package it is possible to connect with the OHM API and work with the historical geodata.

HistOSM[20] is an application for exploration of historical objects of OSM. Volunteers can trace the regional historical characteristics by enlarging and filtering the map, showing the objects of interest. If a click occurs on the objects individually, the user will see their detailed information, such as tags and links (e.g. associated images or websites), and it can also go to the OSM website to edit the objects and if necessary, add or update information.

OHM data and the HistOSM are based on the OSM, so the quality control is done

[16] http://buildinginspector.nypl.org/data
[17] https://wiki.openstreetmap.org/wiki/OAuth
[18] https://wiki.openstreetmap.org/wiki/Osmapi
[19] http://www.openhistoricalmap.org
[20] http://histosm.org/

by OSM tools. OSM uses tools to improve the quality of its data. These tools provide a list of errors in the data that volunteers can correct through editing tools. Some of these tools are divided into: bug reporting (e.g. Notes[21]); error detection (e.g. Keep Right[22]); monitoring (e.g. History Browser[23]); assistant (e.g. Traffic Sign Tool[24]) and tag statistics (e.g. Taginfo[25]) (OPENSTREETMAP, 2017d).

VGIMWS has similar characteristics to the works described above, being influenced by someones, like the OSM RESTful API. There is a project called Historic Event[26], whose purpose was to add historical events into OSM, but it was not continued. One of the main focuses of the Pauliceia 2.0 project is the crowdsourcing, because who feed the platform are the end users, who are people with historical data interested in collaborating. VGI is used to motivate citizens to participate in the collection of spatial information with quality, acting as sensors. The users will provide feedback (e.g. notifications and denunciations), in order to improve the data.

2.7 Differences between existing projects and the Pauliceia 2.0 project

OpenStreetMap has as objective to create a free, editable map of the entire world constructed by users without deep knowledge and it has an open-content license (OPENSTREETMAP, 2017a). OSM has a RESTful API to manage raw geodata from/to OSM database. This API uses HTTP request methods and the XML format to exchange its data (OPENSTREETMAP, 2018a)(OPENSTREETMAP, 2018b). Originally the Pauliceia 2.0 team wanted to use the OSM to store its spatiotemporal data provided by historians (more specifically the OpenHistoricalMap with the OSM API). After some analyzes, it was decided that is not a viable solution. Table 2.1 illustrates the why the Pauliceia 2.0 project will not use the OSM to save its spatiotemporal data.

On Pauliceia 2.0 the data provided by historical researchers are resulted from their researches, so they cannot be updated by anyone. While the data in OSM can be updated by anyone[27]. The data provided by researchers normally no longer exist (e.g. crimes in 1930 or a demolished hospital in 1900). On OSM does not make sense to insert these data. There is the tag “historic”, however it cannot be used for data

[21] http://wiki.openstreetmap.org/wiki/Notes

[22] http://wiki.openstreetmap.org/wiki/Keep_Right

[23] http://osm.virtuelle-loipe.de/history/

[24] http://osmtools.de/traffic_signs/

[25] http://wiki.openstreetmap.org/wiki/Taginfo

[26] http://wiki.openstreetmap.org/wiki/Proposed_features/historic_event

[27] http://wiki.openstreetmap.org/wiki/Pros_and_cons_of_contributing_data_public_domain

Table 2.1 - Differences between Pauliceia 2.0 project and OpenStreetMap.

Pauliceia 2.0 project	**OpenStreetMap**
Data sets provided by historians resulting from their researches can not be modified by anyone.	All data sets in OSM can be modified by anyone.
Normally historical researchers provide data that no longer exist.	In the OSM does not make sense to insert a object (e.g. building) that does not exist anymore. OSM tag “historic” is not for features that no longer exist.
Pauliceia 2.0 database has spatiotemporal features.	OSM does not have the concept of spatiotemporal features. OSM tag “end_-date” should not be used for roads.
Pauliceia 2.0 features are organized as layers (e.g. factories of Bom Retiro in 1930).	OSM does not have the concept of layers. Data on top of data is just a mess.
Pauliceia 2.0 project has a specific domain and community. The domain of Pauliceia 2.0 is related to a specific spatial and temporal scope, generating a structured community.	OSM has a general domain and community.

that no longer exist[28]. Pauliceia 2.0 data (e.g. addresses, streets and buildings) are spatiotemporal, nevertheless OSM does not have the concept of layers. Insert data on top of other data is just a mess in OSM[29]. There is the tag “end_date”, however it should not be used for streets[30].

Besides, the Pauliceia 2.0 project wants to handle private historical objects and events, that are the research of the historians. OpenHistoricalMap manages just historical objects and the Historic Event, that would work with historical events, was not continued. As the OpenHistoricalMap and the Historic Event are OpenStreetMap projects, the data should be public for anyone can modify them. There is the HistOSM, that is an online visualizer of historical objects of OSM, however it is the same case of OpenHistoricalMap, the HistOSM works with the historical objets of OpenStreetMap and it does not manage historical events.

Pauliceia 2.0 community is structured, with a very specific domain of geographical information, that are the historians and their students that produce historical data through their history researches. Besides, the focus of the Pauliceia 2.0 project is the

[28] https://wiki.openstreetmap.org/wiki/Historic

[29] http://wiki.openstreetmap.org/wiki/Import/Guidelines

[30] http://wiki.openstreetmap.org/wiki/Key:end_date

period of urbanization of São Paulo, that is from 1870 to 1940. For that reason, the Pauliceia 2.0 community is very different from the OSM community or other historical projects. Hence, the Pauliceia 2.0 team will not use the OSM-based projects to store its data, what justifies the proposed architecture in the section 2.3.1. In a nutshell, they are projects with different goals. OSM may be used as reference for the current date, for example, serving as a base map.

Building Inspector's API can be used to return the consolidated polygons, building footprints, address numbers from a building, sheets, polygons from a sheet, history and so on. While the VGIMWS is to focus in the collaborative data retrieved by researchers. These data are organized in spatiotemporal layers.

2.8 Web Service

Traditional applications have automated several tasks that people made manually, like complicated calculations. These kind of system is isolated, where each application have its own computer or home. Then, one application does not talk with other, because of that, they are not a great method to exchange data between them. This problem has started to be solved with distributed computing, where different systems can communicate with each other, even in different locations. CORBA, MTS and others technologies, provide applications that can find components of others applications that wants to interact. Then, a system can use a resource of other application in different computers, nevertheless there is a problem. The problem is that the client application needs to know the server application, using the same technology, being a closed system (CHASE, 2006).

RMI, CORBA and DCOM were used to devise client and server systems. Distributed systems with these technologies are highly coupled, because server and client depend/know each other. To create systems with loosely coupled are used the web services. Web services are independent platforms which the clients do not need to have any prior knowledge of them, before they actually use it (MUMBAIKAR et al., 2013). RESTful Services and SOAP Web Services are two architectural styles to develop web services (ADAMCZYK et al., 2011).

Simple Object Access Protocol (SOAP) is a lightweight communication protocol based on XML for exchange of information in distributed environment. SOAP protocol consists of three parts: (1) envelope: describe a framework to define what is inside one message and how to perform it; (2) encoding rules: define mechanisms to exchange instances of application-defined data types; and (3) remote procedure call (RPC) representation: describe a convention to represent remote procedure calls and responses (BOX et al., 2000). SOAP Web Services are based on XML and SOAP protocol (MUMBAIKAR et al., 2013). It is transmitted encoded messages in XML over HTTP. This XML is defined by the Web Service Definition Language (WSDL) (MULLIGAN; GRAČANIN, 2009). The WSDL is a specification that produces a model and an XML format to define web services. This specification sets up a isolated description of the abstract functionality from concrete structures of a service description (CHINNICI et al., 2007).

The Representational State Transfer (REST) architectural style was introduced by Roy Thomas Fielding in his doctoral thesis (FIELDING; TAYLOR, 2000). In this architecture, the client service sends a request to the server service, so the server handle the request, returning a response. This is done by transfer of representations of resources, which one resource is a thing, only determined by a URI. A resource is represented by a document that has the current or intended state. The REST is established on the adoption of nouns and verbs to identify the resources and use the HTTP methods (i.e. GET, PUT, POST and DELETE) to get, insert, update and remove the resources. SOAP messages require message format like envelope and header, what it is not necessary for REST (MUMBAIKAR et al., 2013).

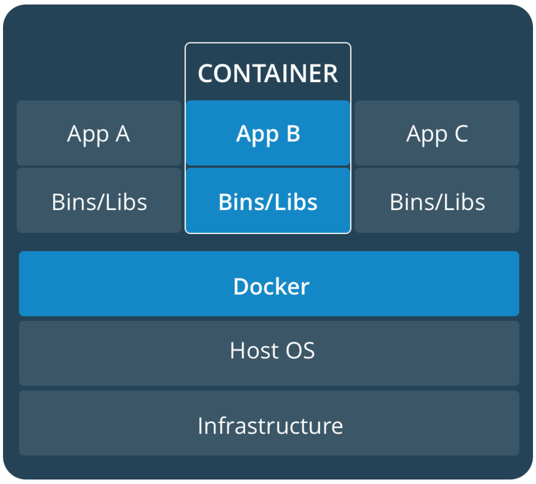
RESTful services have the following properties: (1) stateless, (2) uniform interface and (3) addressability (MUMBAIKAR et al., 2013). Stateless means that all request from client to server must have all crucial information to interpret the request. Then, the session state is kept absolutely on the client, the server cannot store any context (FIELDING; TAYLOR, 2000). Each transaction is unique and unconnected with past transaction, because all necessary data to execute the request is contained in the request. Not being necessary to maintain any session in the server (MUMBAIKAR et al., 2013). Uniform interface is when the development is decoupled from the services, encouraging the autonomous evolvability (FIELDING; TAYLOR, 2000). In practice it means that exists a interface to handle the resources of the server (i.e. using the HTTP methods). It is called addressability when in the system each resource can be identified uniquely by a URI (MUMBAIKAR et al., 2013).

2.9 Docker

Docker is an open system, whose goal is to help the development, deployment and execution of applications in isolated environments, called containers. A container is a set of processes that run isolated in a kernel. Docker is designed mainly to make an application accessible quickly, allowing the user to manage the application infrastructure, making it easy to create, maintain and update programs (GOMES, 2017). Docker containers are flexible, lightweight (share the host kernel), interchangeable (it is possible to deploy improvements on-the-fly), portable (creates a container locally and runs it wherever the user wants), scalable (it is possible to manage easily container replicas) and stackable (the user can stack services vertically) (DOCKER, 2019).

Docker provides a “language” to build files with descriptions of the essential infrastructure and how the application will be used in that environment (e.g. which the service port, which external volume data and so on). It also has a public cloud for supplying ready-made environments, such as an apache image, configuring just a few specific requirements to the users’ need, thereby building their custom environment (GOMES, 2017).

Figure 2.4 - Docker Architecture.



SOURCE: Docker (2019).

Docker uses virtualization at the operating system level as a method of isolation. This allows for numerous processes to run in isolation on a single host (i.e. operating system kernel). This isolation is done with the use of kernel namespaces that build isolated environments among containers. This causes that the processes that run within one container, not have access to the resources of other, unless this is explicitly released (GOMES, 2017). Figure 2.4 shows the Docker layers.

2.10 Conclusion

VGI is a version of crowdsourcing that uses a large number of untrained volunteers, to produce, gather and spread geographic information in web platforms, such as OpenStreetMap and Pauliceia 2.0. VGI contributions have a geographical location and several attributes, that describe the location. By default, VGI does not guarantee the quality in its data, because of that, it is necessary to provide at least a quality standard.

NMAs and CSCs use rigorous specifications that drive the collection of geographic data, while normally the VGI projects have lack of specifications. So, it is suggested the definition of a standard to ensure the VGI data quality, as a protocol. The definition of a protocol is important to aid and expand the reuse of VGI data, for objectives and applications different from what they were collected originally. Mooney et al. (2016) proposed a VGI Protocol to manage the VGI data collection. Its main steps are: initialisation, data collection, self-assessment and quality control, data submission and feedback to the community.

VGI data quality can be expressed in essence by: quality measures and quality indicators. Quality measures use authoritative data as a set of references, comparing them with the data generated by VGI, they are defined by: completeness, consistency, positional accuracy, thematic accuracy and temporal accuracy. Quality indicators are used when authoritative data are not available, expressing the data quality by other ways, such as purpose, usage and lineage. There are more abstract quality indicators, as: trustworthiness, experience, credibility, text content quality, vagueness, local knowledge, recognition and reputation. Goodchild and Li (2012) defined three approaches to guarantee the VGI quality: crowdsourcing, social approaches and geographic approaches.

Pauliceia 2.0 project has as objective to build an online computational system for collaborative management of historical data. VGI is used by researchers to create maps related to their own research and making the manual vectorization of the streets and buildings of the old maps. Pauliceia 2.0 architecture has two groups of web services: the first one are the geographic web services defined by the OGC and the second one are the specifics web services (i.e. VGI, spatiotemporal geocoding and spatiotemporal visualization).

In order to help to improve the quality of the collected historical data and to aid in the reuse of the Pauliceia 2.0 platform for other historical projects, it is proposed a VGI protocol for historical data, in the context of Pauliceia 2.0 project. In VGI protocol for historical data is described a crowdsourcing approach to ensure the data quality of contributions, which is the involvement of the platform members to assess the data through denunciations. For example: if it appears a data with inappropriate content, a user can do a denunciation.

Based on the defined VGI protocol, it was designed and built a VGI web service to manage the historical data. This service is part of the specific web services of Pauliceia 2.0 project. VGIMWS is a RESTful service, where one resource can be found through a unique URI. For example: in order to create a new layer is used a specific URL. Inside the Pauliceia 2.0 server, VGIMWS is run within a isolated Docker container, while the other Pauliceia 2.0 services run in another containers.

3 VGI PROTOCOL FOR HISTORICAL DATA

In order to guarantee the quality of the VGI data collected by Pauliceia 2.0 project, it was needed to define a VGI protocol that drives the data collection of vector data generated by the platform. This VGI protocol for historical data defines important issues that improve the comprehension of volunteers about the project, all its techniques and processes to gather and manage citizen-derived geographical data, and assisting the quality of these data.

The formalization to describe the VGI Protocol for historical data follows the standardization proposed by Mooney et al. (2016) through the description of the main stages of the VGI Protocol for vector data. Then, this section describes the VGI Protocol for historical data, applied in the context of Pauliceia 2.0 project.

3.1 Data types

Inside the Pauliceia 2.0 portal volunteers can insert or update vector geographical data, provided by their researches. These vector data may be points, lines or polygons, besides having textual and numerical properties associated to them, such as links to documents (e.g. photos or videos) that must be put in other repositories (e.g. Dropbox or YouTube). Until now, the platform does not supply upload or edition of raster data types.

In order to allow the citizens contribute with geographical data, Pauliceia 2.0 project expects to apply crowdsourcing and VGI methods for the vectorization of data from historical raster maps (e.g. old streets and buildings). Then, during the data collection process, the contributors can vectorize the same data, generating more than one feature that represents that data, such as a street. For that reason, it is intended to develop an algorithm that given various features that represent one data, such as a set of buildings, it will compute the better representation of that data, deriving the most accurate geometry from that data set, as the one suggested by Budig et al. (2016).

Events are organized with the purpose of promoting and instigating volunteers to vectorize streets and buildings of old maps of São Paulo city from 1870 to 1940, and historians distribute their historical data sets. These events are called HistMapathon (Historic Mapping Marathon) and they are similar to those made by Google Maps (TECH2, 2014) and OpenStreetMap (OPENSTREETMAP, 2017c). These events are organized by historians and their students, and they can be held at universities. They contain tutorials about the Pauliceia 2.0 platform, describing how to contribute. The main purpose of these events is to promote the mass contribution of historical geographical data to the Pauliceia 2.0 platform. A collaborator can also contribute individually without a HistMapathon. A HistMapathon organized by the Pauliceia 2.0 team is described in Chapter 5.

The inserted data in Pauliceia 2.0 platform must follow the spatiotemporal restriction of the project. The data are limited to scope of the project, that is the central area of São Paulo from 1870 to 1940.

3.2 Initialization

Pauliceia 2.0 is a web-based platform, so the history researchers or any citizens enter on portal using an online browser. In order to start the data collection, it is necessary to register a new user, logging the platform with credentials or use a social login, with Google or Facebook. Before starting the contribution, the user must read and accept a Use Term. A copy of the Pauliceia 2.0 Use Term is shown in Appendix A. In a nutshell, this Term describes that the system is not responsible for the contributions and that the data of the platform are public.

All data sets of the Pauliceia 2.0 project are available under the Creative Commons Attribution-ShareAlike 4.0 license (CC BY-SA)[1]. Basically this license allows the people freely copy, share, adapt and use the Pauliceia 2.0 information for any purpo, since the users credit Pauliceia 2.0 and its contributors. If the user downloads and modifies the Pauliceia 2.0 data, the user must use the same license for the results. A summary of the license can be described as: *"This license lets others remix, tweak, and build upon your work even for commercial purposes, as long as they credit you and license their new creations under the identical terms. This license is often compared to "copyleft" free and open source software licenses. All new works based on yours will carry the same license, so any derivatives will also allow commercial use. This is the license used by Wikipedia, and is recommended for materials that would benefit from incorporating content from Wikipedia and similarly licensed projects"* (CREATIVE COMMONS, 2018).

The platform provides the following specifications and manuals for volunteer initialization:

- a specification about the Pauliceia 2.0 project: this documentation will

[1] https://creativecommons.org/licenses/by-sa/4.0/

}

make the users more familiar with the platform, making them to understand why and how to collect the data. The users are encouraged to comment, ask questions or provide suggestions to the portal by sending an email to pauliceia_support@googlegroups.com;

- a tutorial of how to use the Pauliceia 2.0 platform: it is necessary that the volunteers read examples before starting the data collection. Then, they will be able to solve their main problems, to have a previous training and to become accustomed with the processes of data collection, making them to feel confident to insert the real data;
- the Pauliceia 2.0 database license: the volunteer must be sure that the collected data is of public domain. The data inserted into the platform cannot have any private license, because the Pauliceia 2.0 database is open data. A licence copy can be found in the follow web link: https://creativecommons.org/licenses/by-sa/4.0/.

3.2.1 How to cite?

If users want to use the data inside the Pauliceia 2.0 platform, or they need to cite the platform or the data in a study, for example, they may use a bibtex standard, as shown below:

```
@unpublished{ferreira2018platformvgi,
    title={A Platform for Collaborative Historical Research based on
    Volunteered Geographical Information},
    author={Karine R. Ferreira and Luis Ferla and Gilberto R. de
    Queiroz and Nandamudi L. Vijaykumar and Carlos A. Noronha and
    Rodrigo M. Mariano and Denis Taveira and Gabriel Sansigolo and
    Orlando Guarnieri and Thomas Rogers and Jeffrey Lesser and
    Michael Page and Fernando Atique and Daniela L. Musa and
    Janaina Y. Santos and Diego S. Morais and Cristiane R.
    Miyasaka and Cintia R. de Almeida and Luanna G. M. do
    Nascimento and Jaíne A. Diniz and Monaliza C. dos Santos},
    note = {Journal of Information and Data Management.
    (accepted for publication in 2019)},
    year = {N.D.}
}

@article{ferreira2017pauliceia,
    title={Pauliceia 2.0: A Computational Platform for Collaborative
    Historical Research},
    author={Karine R. Ferreira and Luis Ferla and Gilberto R. de
    Queiroz and Nandamudi L. Vijaykumar and Carlos A. Noronha and
    Rodrigo M. Mariano and Yasmin Wassef and Denis Taveira and
    Ivan B. Dardi and Gabriel Sansigolo and Orlando Guarnieri and
    Daniela L. Musa and Thomas Rogers and Jeffrey Lesser and
    Michael Page and Andrew G. Britt and Fernando Atique and
    Janaina Y. Santos and Diego S. Morais and Cristiane R.
    Miyasaka and Cintia R. de Almeida and Luanna G. M. do
    Nascimento and Jaíne A. Diniz and Monaliza C. dos Santos },
    journal={Proceedings XVIII GEOINFO, December 04th to 06th, 2017},
    address = {Salvador, BA, Brazil},
    pages={28--39},
    year={2017}
}
```

Other way of citing the Pauliceia 2.0 platform is using the ABNT standard, as shown below:

FERREIRA, K. R.; FERLA, L.; QUEIROZ, G. R. de; VIJAYKUMAR, N. L.; NORONHA, C. A.; MARIANO, R. M.; et al.. A platform for collaborative historical research based on volunteered geographical information. Journal of Information and Data Management. (accepted for publication in 2019)

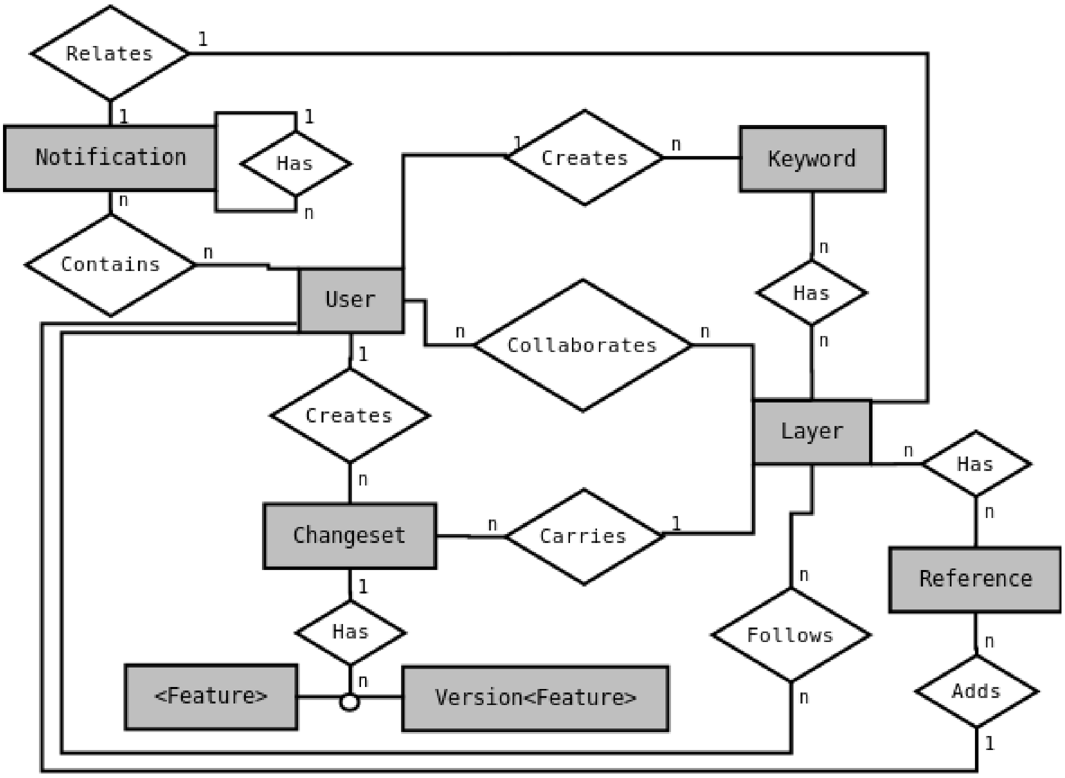
FERREIRA, K. R.; FERLA, L.; QUEIROZ, G. R. de; VIJAYKUMAR, N. L.; NORONHA, C. A.; MARIANO, R. M.; et al.. Pauliceia 2.0: A computational platform for collaborative historical research. Proceedings XVIII GEOINFO, December 04th to 06th, 2017, Salvador, BA, Brazil, p. 28–39, 2017.

It is necessary also to inform that the data is under the CC BY-SA license, putting a hyperlink that indicates the license, as: `https://creativecommons.org/licenses/by-sa/4.0/`.

3.3 Data Model

Figure 3.1 shows the conceptual data model of the Pauliceia 2.0 VGI protocol, using an entity–relationship diagram, that defines the main concepts of the Pauliceia 2.0 project and their relationships.

Figure 3.1 - Complete Entity–relationship Model of the Pauliceia 2.0 Spatiotemporal Database.



User: user_id, email, password, username, name, created_at, is_email_valid,
terms_agreed, login_date, is_the_admin, is_curator, receive_notification_by_email,
picture, social_id, social_account
Layer: layer_id, f_table_name, name, description, source_description, created_at,
Changeset: changeset_id, description, created_at, closed_at, user_id_creator, layer_id
<Feature>: id, geom, <attributes>, version, changeset_id
Version<Feature>: id, geom, <attributes>, version, removed_at, changeset_id
Notification: notification_id, description, created_at, is_denunciation, user_id_creator,
layer_id, notification_id_parent
Reference: reference_id, description, user_id_creator
Keyword: keyword_id, name, created_at, user_id_creator
Collaborates: user_id, layer_id, created_at, is_the_creator
Contains: user_id, notification_id
Follows: user_id, layer_id, created_at
Has (Layer-Keyword): layer_id, keyword_id
Has (Layer-Reference): layer_id, reference_id

SOURCE: Author's production.

There are five main concepts in the data model: User, Layer, Reference, Keyword and Notification. They will be described in the following sections.

3.3.1 Layer, Keyword, Reference and Notification

The users must organize their data in layers, as in GIS (Geographic Information System). A layer is a *"visual representation of a geographic dataset in any digital map environment. Conceptually, a layer is a slice or stratum of the geographic reality in a particular area, and is more or less equivalent to a legend item on a paper map. On a road map, for example, roads, national parks, political boundaries, and rivers might be considered different layers."* (ESRI, 2018). On Pauliceia 2.0 platform, the layers are data sets of vector data. They have a name and they may have a description. The historians can create how many layers they want and these data sets will be related with their research.

The layers on the Pauliceia 2.0 platform are organized by keywords and references. A keyword is a word or a phrase that describes the layer, such as: "public health", "cultural places", "crimes" and so on. Users are who that define the keywords or they can use predefined ones. A reference is a text that describes the origin of the data, whether it is from an article, book, dissertation or other. The reference must contain the bibliographic reference of the original document of the data and it must be described following the bibtex standard. The layer creator can add multiple keywords and references in his or her layer.

There are two user types associated with a layer: (1) "creator user": the user who created the layer on the Pauliceia 2.0 platform and who can add other users as collaborators of his or her layer; and (2) "associated users": users associated with a layer that can add, modify, delete features in the layer (i.e. volunteers can only manage layer data who they are part of). In addition, the information of the authors of a layer is stored, that are the people who generated the information of a layer from a research. The creator user of a layer may be different from its authors. Theoretically, the historian will add only in his or her layer trustworthy people, who has the ability of adding and evaluating the data correctly, generating a local quality control.

It is possible to have volunteers outside layers who discover problems in the data, for example, a bad vectorization of a street. These users may add comments, indicating any issues that may exist, through the notifications. The notifications are comments done in the platform that the users receive an associated alert. Thus, the layer members can evaluate if or not these comments are valid, deciding if or not the data are correct. The use of local knowledge of users related to a layer, is an indicator of quality.

A user can follow layers, receiving notifications about them. For example: if a user follows a layer, he or she will receive a notification when someone writes notifications in that specific layer.

3.3.2 Feature

A feature is a record that contains spatial and non-spatial attributes. The feature follows the standard defined by OGC, known as Simple Feature Access (HERRING, 2006). The set of features is represented by "<Feature>" entity and it has the same name of the layer, because of that the name is between the signals of less than and greater than. A feature may represent addresses, streets, buildings or other spatial data. The attributes of the features are defined by users, when he or she creates a new layer (i.e. when the user creates an empty layer, he or she defines explicitly the layer properties or when the user upload a file on Shapefile format, the properties are already inside the file). The features are the historical data sets of the researchers, that are vector data and it must have a start date and an end date, that are the temporal scope, and a geometry attribute (i.e. point, line or polygon).

The non-spatial attributes of a feature can be text, numerical, date or media file. Related to the media file, it may be historical photos, videos, testimonial audios and so on. These files must be stored in a repository whose public link is added to data. For example: videos are placed on YouTube, images on Google Photos and documents are added on Google Drive (or Dropbox); in the contribution is added the public URL as an attribute. It is advised for the user creates a project account in the repository that will be used, to avoid accidental deletes. It is possible to include photos directly in the features, such as historical images. In section 4.1 will be proposed a model to store the metadata of temporal attributes and media files attributes.

The vector data are the historical data with a geographical location and attributes, for example: the occurrence of a theft in 1930, at street São Bento, n. 3; a historical address; a building; the vectorization of old streets and so on.

As the Pauliceia 2.0 project is geared toward urban historical research of the São Paulo city from 1870 to 1940, the collected data should be restricted to this space and period of time. More precisely, the data should cover the central area of São Paulo, which is the defined as pilot area. The type of data that can be collected and inserted in the platform are vector data, not raster data.

In Figure 3.1 between "User" and "<Feature>" there is an entity called "Changeset". A changeset is a group of editions made, being a concept of Version Control System (REDMOND et al., 2008). OSM uses the changeset to manage the versions of the elements, as the additions and editions made (OPENSTREETMAP, 2018c). A changeset is used to control the version of the features, that are related to a layer. A user can edit the data, so a history of that is held, inside the "Version<Feature>" using the version attribute. "Version<Feature>" entity is generated together when the feature table is created and <Feature> is the same name of the feature table. A changeset must have a description, that describes the changes about the data, and the creation date.

3.3.3 Types of Users

Regarding the type of users, there are two states: the unlogged users and logged users. The unlogged users are collaborators who do not have a login. They can visualise the system information (e.g. layers and historical maps) and download them. In the platform, there are three types of logged users: normal, curator and administrator.

There are two main user groups that can contribute with historical data: historians and citizens. Even that these two theoretical groups exist, in the platform they are materialized as normal users, who are users who can contribute with historical data, whether referring to historical research or some personal historical data.

All users can view and access the data sets through the portal, but only normal ones can add or edit the data. They can: 1) produce layers related to their historical research associated with keywords or references. This can be done manually or by bulk import; 2) update layer information of layers created for them or other that they are part of; 3) delete layers that they created; 4) manage historical data sets in the layers that the users are part of; 5) add or remove volunteers in layers that they created; 6) write notifications in layers. If they find a trouble in data, the users may write comments and give suggestions of correction; 7) write general notifications in the system (e.g. announcements of events). 8) write denunciations when they discover an unappropriated data on the platform; and 9) follow layers, receiving notifications about the followed layer.

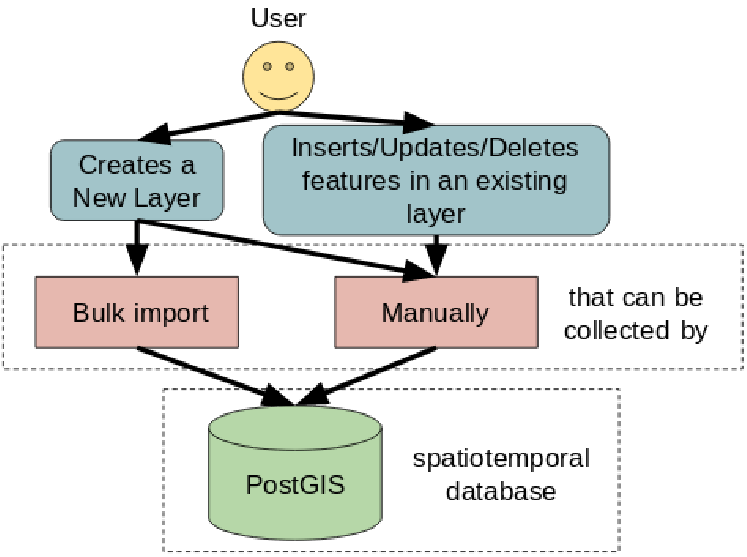
The curator users promote the collaboration of certain keywords and geographic space inside the platform. This type of user has also the same grants of a normal user. They can update the layer information of any user and organize them, creating new keywords for the layer. On “User” entity there is a flag attribute called “is_-curator” that indicates if a user is curator or not.

The administrator users are special users that have permission to insert, update and delete all entities of the Pauliceia 2.0 platform. This type of user can visualize and analyze the denunciations made by others, and remove a bad user, that inserted an unappropriated content on the portal, if it is needed. They have also the same grants of a curator user. On “User” entity there is a flag attribute called “is_the_admin” that indicates if a user is administrator or not.

3.4 Data Collection Methods

Figure 3.2 illustrates the data collection process, where the users create a new layer or contribute in an existing layer, inserting their research data, that can be collected manually or by bulk import. Finally, the contributions are saved inside the database.

Figure 3.2 - Data collection process.



SOURCE: Author’s production.

The manual contribution is done by users, where they can manually add the features by creating a new layer and vectorizing the features clicking on the historical maps displayed by the web platform. Besides, they can add or update the features’ attributes. The volunteer can just add, update or remove features and their attributes of layers that he or she is part of, as a creator of that layer or as a collaborator. In this type of contribution, it is needed that the volunteers inform the temporal metadada associated to the features. The procedure to create the data manually is to: (1) create a new layer; (2) inform the layer information, as name and description; (3) inform the columns of the layer; (4) save the layer; (5) open the edition page; (6) insert the data manually on the map, describing its attributes; and (7) save the modifications.

All data collected by manual vectorization are stored in the database using a Coordinate Reference System (CRS) called WGS 84 (EPSG:4326).

The bulk import refers to the uploading of geographical file data into the Pauliceia 2.0 platform. Collaborators can create a new layer, inform the layer information and upload a set of features saved in a Shapefile or GeoJSON, that are well-known file formats of vector geographical data. The metadada of these files are extracted automatically by the platform. The procedure to do a bulk import is to: (1) have the data in a vector geographical data format, saved in a zip file; (2) create a new layer; (3) inform the layer information, as name and description; (4) upload the zip file; and (5) save the layer.

Through the bulk import, the CRS is extracted from the files automatically. In the case of a Shapefile, EPSG is collected by the .prj file, while in the GeoJSON, EPSG is gathered by the geometry propertie called "crs" or if it does not exist, so the CRS is saved as EPSG:4326.

Through the Pauliceia 2.0 platform it is possible to generate a Shapefile to be used in the bulk import. In the initial page of the portal, the user can attach a CSV (Comma Separated Values) file to geolocate the addresses contained within it. Inside this CSV, each row is an address and the columns are the attributes, such as the street name, number and year. So the platform will produce a Shapefile from this CSV, that the user can download it. With this Shapefile, the user can upload it inside a new layer, through the bulk import. The default CRS of this Shapefile is EPSG:4326.

Examples of historical data collected by VGI using the two types of data collection are: (1) vectorization: the users can make the vectorization of the streets and buildings of old maps from 1870 to 1940; (2) research historical data: the users can contribute with historical data provided by their research (e.g. factories in Bom Retiro in 1900, hospitals in 1930 and so on); and (3) historical data with media: the users can insert their research data and attach in them media files, that are saved in repositories as Dropbox, Google Drive or Youtube, and a public link is added as an attribute. Some examples of media files are: historical photos, documents, videos, audios and etc.

3.5 Quality Control

As described in the section 2.5, quality measures use authoritative data sets for comparison. In the context of the Pauliceia 2.0 project, as far as it is known, there are not historical reference data for the São Paulo city for the period of interest. For that reason, this approach is not applicable. Therefore, quality indicators and quality approaches are used.

The users need to read this protocol before starting the data collection. This contributes to remove the main questions, avoiding that problems can happen, increasing the quality of the inserted data.

Goodchild and Li (2012) indicate the use of the Linus Law[2] to do the generated data to converge to the true. In a nutshell, the Linus Law says that, how many more volunteers contributing, higher will be the quality of the data. Haklay et al. (2010) studied the application of this method and concluded that really, how many more users contributing, more is the probability of having data quality. In addition, it was concluded based in the OSM data, where users are usually “normal” (usually they do not have depth study of the inserted data, only local knowledge). This type of observation for the Pauliceia 2.0 project is important, because the target audience are the historians, who have prior knowledge of the subject, so they are able to correct eventual problems and their data will have credibility. Goodchild and Li (2012) introduce a quality approach based on the Linus Law concept, it is known as crowdsourcing approach. This approach describes that is important to give the opportunity of a group of people validate and correct errors that may exist in the system, verifying if the contributed VGI data converge to the truth. In order to apply these resources, Pauliceia 2.0 project uses the concepts of notifications and denunciations.

As only users that belong to a layer (i.e. trusted people) can manipulate the data, a local quality is generated in the contributions. If external users find poor data quality, they may write comments and make observations about the data, through the notifications, indicating the problems that were found.

[2] “Given enough eyeballs, all bugs are shallow” (RAYMOND, 2017)

The user must not supply data with copyrights (e.g. owned by another researcher) or data with inappropriate content on the portal, as for example, pornography. Users must agree to the terms of use of the platform before they begin to contribute their data, so that they are warned of these questions. Because of that, as a security mechanism, the platform provides a resource of denunciation. A denunciation is a certain type of notification did to warn the administrators that a layer does not have proper data. If the users find an inappropriate content in the platform, they must report it through the denunciations. Then, an administrator user will receive them, verifying the denunciation and if it is confirmed that the content is really improper, so the data will be deleted, as well as the owner user.

Historians will contribute with layers based on their historical research, because of this, these data have already undergone some revision or evaluation of more experienced researchers, supervisors or professors in the area. For this reason, these data will have related to them the quality indicator called "credibility". Citizens will insert historical data that they contain, such as the insertion of historical documents or the mapping of old maps. The collection of these documents will be done by requesting the addition of metadata (e.g. these historical documents) through notifications in a layer. For this reason, these data will have the quality indicator called "local knowledge" related to them. Hence, the collaborative intelligence of volunteers (i.e. historians and citizens) can improve the quality of the data using of quality indicators.

For security reasons, the main actions of the user, such as manipulating historical data, are saved in a history in database, through the changeset concept. This history is stored in database using a version table that has the same schema of the original feature table (i.e. the table that contains all the saved features). How changeset is used on Pauliceia 2.0 platform is described in detail in the Appendix B. This may help to more easily track malicious users who want to damage the system. With this history, the volunteers will be able to follow the updates of the data.

It is advisable for the users diffuse the Pauliceia 2.0 project to people who they know, with the intention of attract new contributors. This will make the platform grows and improves frequently. How many more participants there are, more the system becomes rich in its data, consequently improving its quality (MOONEY et al., 2016).

3.6 Feedback to the Community

A collaborative project improves how many more users contribute to it, so it is interesting that the users provide feedback of their experience (MOONEY et al., 2016). The volunteers can express their comments, opinions and observations on the available channels of the Pauliceia 2.0 project, such as: mailing list[3], Facebook group[4] and Facebook page[5]. So the user can expound about the positive aspects and about what it is necessary to be made better on the platform. The channels are managed by the Pauliceia 2.0 team, as the researchers of INPE and UNIFESP. The user is encouraged to describe whether the data collection process was easy or difficult, and why. The volunteers can write the problems or unexpected situations that they find, suggesting improvements or changes (MOONEY et al., 2016). The user can explain precisely what happened, to facilitate the comprehension of the administrators and apply proper corrections. These observations are important to improve the Pauliceia 2.0 web portal.

The notifications and denunciations, that are resources for quality control, also are adopted as a support for the users provide feedback about their involvement on the portal. Historians can type notifications on available layers, supplying feedback about their situation, such as comments, opinions or tips about the data (e.g. announce a new reference or keyword for one layer), or disclose an unappropriated content through the denunciations. They can also write general notifications, such as event announcements or other announcements that will be received by all members of the Pauliceia 2.0 platform. One option that the user can have is of receiving the notifications by email. They just need to mark an option in the platform that indicates it, where they create a new account.

[3] pauliceia_support@googlegroups.com

[4] https://www.facebook.com/groups/pauliceia2.0/

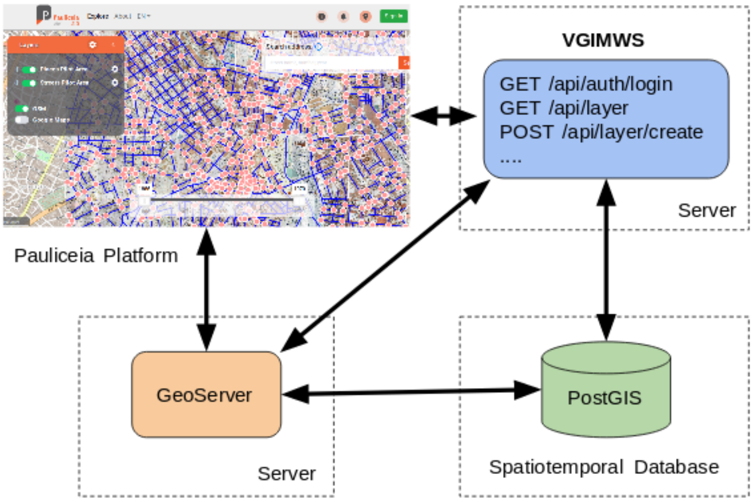
[5] fb.me/grupohimaco/

4 VGI MANAGEMENT WEB SERVICE FOR HISTORICAL DATA

Volunteered Geographic Information Management Web Service (VGIMWS) is a RESTful web service to handle spatiotemporal data. This API has been developed based on VGI protocol for historical data, described in section 3, in the context of Pauliceia 2.0 project. The default format of data exchange of the web service is the GeoJSON or JSON, to work with data with geographic information or without, respectively. It has been developed in Python language, using the web framework called Tornado. Each resource of the VGIMWS can be found through a unique URL.

Figure 4.1 shows the process of the use of the API. A user accesses the Pauliceia 2.0 portal through a browser. So, a front-end component inside the platform will call the VGIMWS to manage the data, through well-defined functions. This component has been built by another member of the Pauliceia 2.0 team. Finally, VGIMWS will store these information in a spatiotemporal database. For example: when a user creates a new layer, a specific function is called to store the information inside the database and the VGIMWS will publish this layer on GeoServer. Then, the Pauliceia 2.0 portal can plot the features on the map using the GeoServer, through the OpenLayers.

Figure 4.1 - Simplistic architecture of Pauliceia 2.0 project, related to VGIMWS.



SOURCE: Author's production.

VGIMWS attends the needs described in the VGI Protocol for historical data, supplying all essential functionalities for handling historical citizen-derived geographical information. VGIMWS works mainly with: (1) user control, managing its proper roles (e.g. users outside of a layer cannot edit its data); (2) spatiotemporal data management, manipulating historical data sets with their attributes, such as geographic and temporal information, and the attachment of historical documents; and (3) provide feedback from/to the community, through the notifications and denunciations.

Figure 4.2 displays a sequence diagram that illustrates the function of creating a new layer in the platform. The volunteer attempts to log in the platform and the web service returns a HTTP status, either success or error. If the contributor is capable to enter in the Pauliceia 2.0 platform, the user can get into the dashboard page, devising a new layer and associating to it any number of keywords and references. After that, the historian can build a blank layer, informing its attributes (e.g. feature, numerical and textual fields, and so on) or upload a Shapefile through the bulk import. In the end, the volunteer needs to describe the layer metadata (i.e. temporal columns, that inform the temporal bounding box and which columns are the temporal information).

Pauliceia 2.0 platform is open source. So, the source code can be found in the Github[1] of the project. VGIMWS is still on development and its source code can be found in its GitHub[2] repository. Inside it, there are described the main resources and how they are being used in the platform.

4.1 Feature Metadata Tables

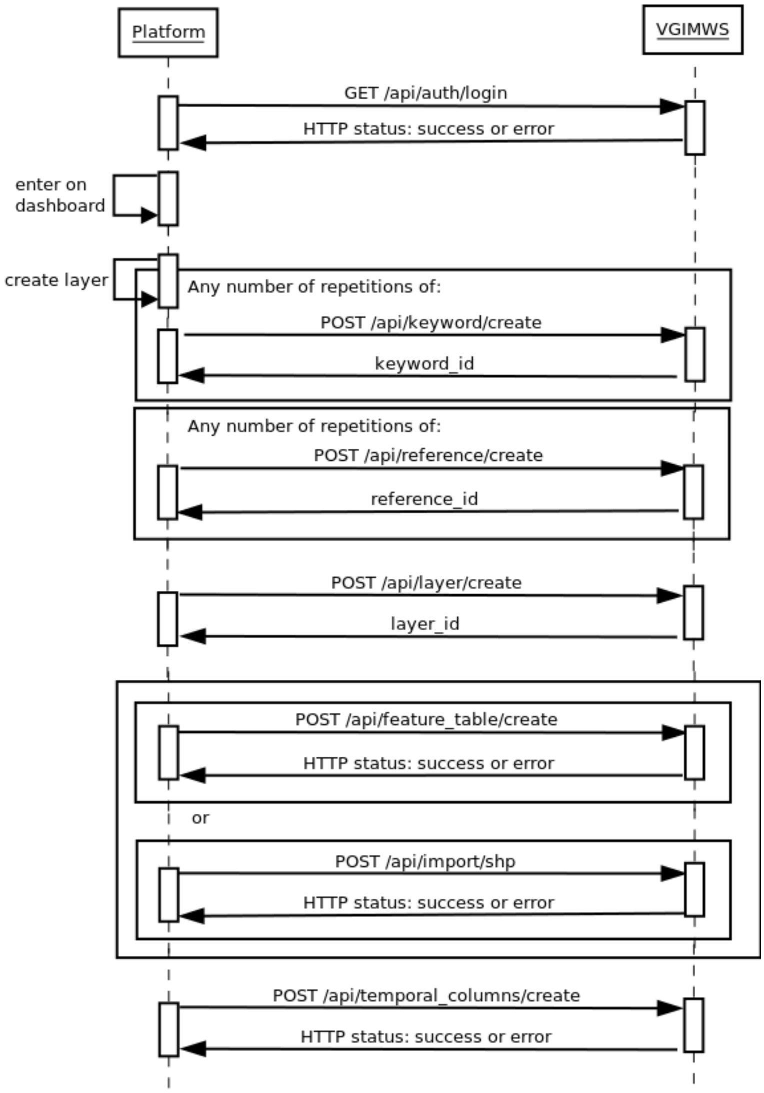
A model to handle the metadata of temporal attributes and media files attributes is proposed. This model is an extension of the standard proposed by Open Geospatial Consortium (OGC), called Simple Feature Access. OGC standard defines the implementation of geographic information. Figure 4.3 illustrates the model, where GEOMETRY_COLUMNS and FEATURE TABLE are entities defined by OGC, and MEDIA_COLUMNS, TEMPORAL_COLUMNS and MASK are tables introduced in this study.

- FEATURE TABLE is a table that saves a collection of features, where its columns represent the fields and the rows mean specific features (HERRING, 2006). Its attributes are: GEOMETRY COLUMN; MEDIA COLUMN,

[1] https://github.com/Pauliceia

[2] https://github.com/Pauliceia/vgiws

Figure 4.2 - Sequence diagram of adding a new layer.


Platform
VGIMWS
GET /api/auth/login
HTTP status: success or error
enter on dashboard
create layer
Any number of repetitions of:
POST /api/keyword/create
keyword_id
Any number of repetitions of:
POST /api/reference/create
reference_id
POST /api/layer/create
layer_id
POST /api/feature_table/create
HTTP status: success or error
or
POST /api/import/shp
HTTP status: success or error
POST /api/temporal_columns/create
HTTP status: success or error

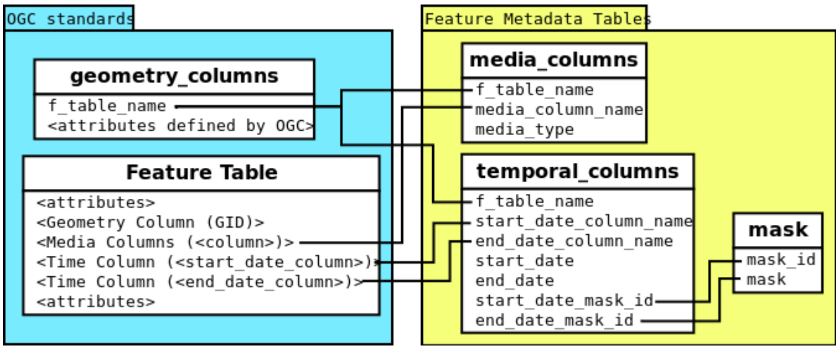
SOURCE: Author's production.

TEMPORAL COLUMN and other attributes defined by user.

- GEOMETRY_COLUMNS is a table that defines the accessible feature tables and their geometry fields (HERRING, 2006). Its attributes are: F_TABLE_NAME that is the feature table name and the other attributes defined by OGC standard, that were omitted in the model.

- MEDIA_COLUMNS is a table that describes the media column of a feature table, with its metadata. Its attributes are: F_TABLE_NAME, that

Figure 4.3 - Feature Metadata tables.



SOURCE: Author's production.

is the feature table name; COLUMN, that is the media column and the TYPE, that is the type of the inserted media (e.g. youtube, dropbox, google_drive and so on).

- TEMPORAL_COLUMNS is a table that informs the temporal information of the feature table, with its metadata. Its attributes are: F_TABLE_NAME, that is the feature table name; START_DATE_COLUMN, that is the column with the start date; END_DATE_COLUMN, that is the column with the end date; and START_DATE and END_DATE that are the temporal bounding box of the data, that is the temporal coverage of the layer.

- MASK is a table that stores the possible masks for the START_DATE_COLUMN and END_DATE_COLUMN columns. It has the MASK_ID that is generated automatically and the MASK, that is the mask (e.g. "YYYY-MM-DD")

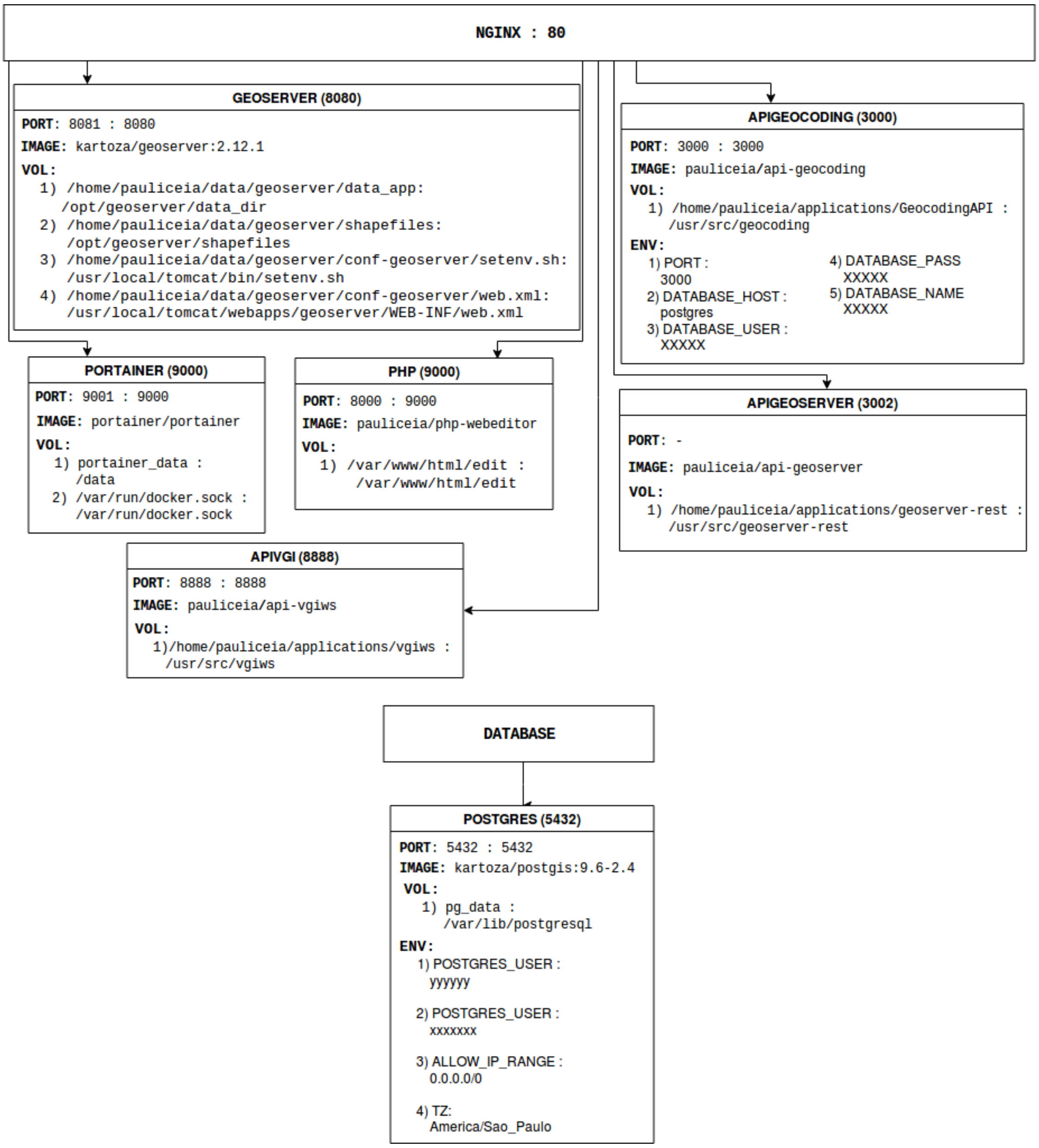
The F_TABLE_NAME attribute of the MEDIA_COLUMNS and TEMPORAL_COLUMNS tables is the same that exist in GEOMETRY_COLUMNS table, defined by OGC.

The <MEDIA COLUMNS <COLUMN>> attribute, in FEATURE TABLE, indicates the column with the web link to a media that is stored in other repository, such as Youtube or Dropbox. The name of this column is saved in MEDIA_COLUMN_NAME field in MEDIA_COLUMNS. The <TIME COLUMN <START_DATE_COLUMN>> and <TIME COLUMN <END_DATE_COLUMN>> attributes, in

FEATURE TABLE, express the column with the temporal field. The name of these columns are saved in START_DATE_COLUMN_NAME and END_DATE_COLUMN_NAME fields in TEMPORAL_COLUMNS table.

4.2 Docker Architecture of the Pauliceia 2.0 project

Figure 4.4 - Docker Architecture of Pauliceia 2.0 project.



```
GEOSERVER (8080)
PORT: 8081 : 8080
IMAGE: kartoza/geoserver:2.12.1
VOL:
  1) /home/pauliceia/data/geoserver/data_app:
     /opt/geoserver/data_dir
  2) /home/pauliceia/data/geoserver/shapefiles:
     /opt/geoserver/shapefiles
  3) /home/pauliceia/data/geoserver/conf-geoserver/setenv.sh:
     /usr/local/tomcat/bin/setenv.sh
  4) /home/pauliceia/data/geoserver/conf-geoserver/web.xml:
     /usr/local/tomcat/webapps/geoserver/WEB-INF/web.xml

APIGEOCODING (3000)
PORT: 3000 : 3000
IMAGE: pauliceia/api-geocoding
VOL:
  1) /home/pauliceia/applications/GeocodingAPI :
     /usr/src/geocoding
ENV:
  1) PORT :              4) DATABASE_PASS
     3000                   XXXXX
  2) DATABASE_HOST :     5) DATABASE_NAME
     postgres               XXXXX
  3) DATABASE_USER :
     XXXXX

PORTAINER (9000)
PORT: 9001 : 9000
IMAGE: portainer/portainer
VOL:
  1) portainer_data :
     /data
  2) /var/run/docker.sock :
     /var/run/docker.sock

PHP (9000)
PORT: 8000 : 9000
IMAGE: pauliceia/php-webeditor
VOL:
  1) /var/www/html/edit :
     /var/www/html/edit

APIGEOSERVER (3002)
PORT: -
IMAGE: pauliceia/api-geoserver
VOL:
  1) /home/pauliceia/applications/geoserver-rest :
     /usr/src/geoserver-rest

APIVGI (8888)
PORT: 8888 : 8888
IMAGE: pauliceia/api-vgiws
VOL:
  1)/home/pauliceia/applications/vgiws :
    /usr/src/vgiws

POSTGRES (5432)
PORT: 5432 : 5432
IMAGE: kartoza/postgis:9.6-2.4
VOL:
  1) pg_data :
     /var/lib/postgresql
ENV:
  1) POSTGRES_USER :
     yyyyyy
  2) POSTGRES_USER :
     xxxxxxx
  3) ALLOW_IP_RANGE :
     0.0.0.0/0
  4) TZ:
     America/Sao_Paulo
```

SOURCE: Noronha (2019).

Figure 4.4 illustrates the Pauliceia 2.0 server internal architecture using Docker containers. There is a reverse proxy server using the Nginx, that listens to the port 80 and it is not inside a container. Nginx guides the requests did by client to the suitable backend services, such as VGIMWS, GeoServer, Geocoding's API and so on. For example: a client list the available layers through the portal, then the platform does a request to the following URL: `http://www.pauliceia.dpi.inpe.br/api/vgi/api/layer`. Nginx gets this request and sends to the appropriate service, that is VGIMWS, that is running in port 8888. Each Docker container in Figure may have a name with the container port, a "PORT" property that defines the binded ports (host : container), a "IMAGE" attribute that is the used image, a "VOL" field that is the volume (host : container) and "ENV" metadata that lists the environment variables.

APIVGI is the Docker container name that runs VGIMWS. The exposed host port is the 8888 and the container port is the same (i.e. "8888 : 8888", both ports are binded), it means that the Nginx sends the requests to the port 8888 in host and it is redirect to the port 8888 in container. The Docker image used is the "pauliceia/api-vgiws". There is a volume (i.e. a folder inside the server that stores data produced by Docker containers) that uses the host folder `/home/pauliceia/applications/vgiws` binded with the container folder `/usr/src/vgiws`. The Pauliceia 2.0 Docker images are stored in Docker Hub[3], while the Dockerfiles are saved in Github[4].

The container GEOSERVER contains a GeoServer. GeoServer is an open source Java-based server for managing geographical data. It uses the standards defined by OGC, such as WMS, WFS and so on (GEOSERVER, 2019). GeoServer administers the geospatial data from Pauliceia 2.0 database, like ploting the historical maps on the platform.

PORTAINER is a container that consist of a Portainer application. Portainer is a web site to monitor Docker containers, volumes, images and networks (PORTAINER, 2019). Portainer handles the containers from Pauliceia 2.0 server, in a friendly user interface.

The container PHP has the first version of Pauliceia 2.0 portal, called Edit or Web Editor. APIGEOCODING is a container that includes the geocoding web service. This service does the geocoding of the historical addresses. The container APIGEOSERVER contains a middleware between the GeoServer and VGIMWS.

[3] https://hub.docker.com/u/pauliceia

[4] https://github.com/pauliceia/dockers-images

VGIMWS communicates to GeoServer through this service, in order to publish and unpublish a feature table (i.e. layer) from Pauliceia 2.0 database. POSTGRES is a container that stores the vector data from Pauliceia 2.0 platform using a PostgreSQL database with PostGIS extension.

4.3 Database model

The current model of the spatiotemporal database of the Pauliceia 2.0 project is shown in Figure 4.5 and it reflects the conceptual model of the Pauliceia 2.0 project, described in Figure 3.1 with the metadata tables proposed in section 4.1. The model informs the concepts of the Pauliceia 2.0 project, such as: user, layer, feature table, reference, keyword, changeset, notification, media, temporal information and followers. In order to store the historical data, it is used the PostgreSQL with the spatial extension called PostGIS.

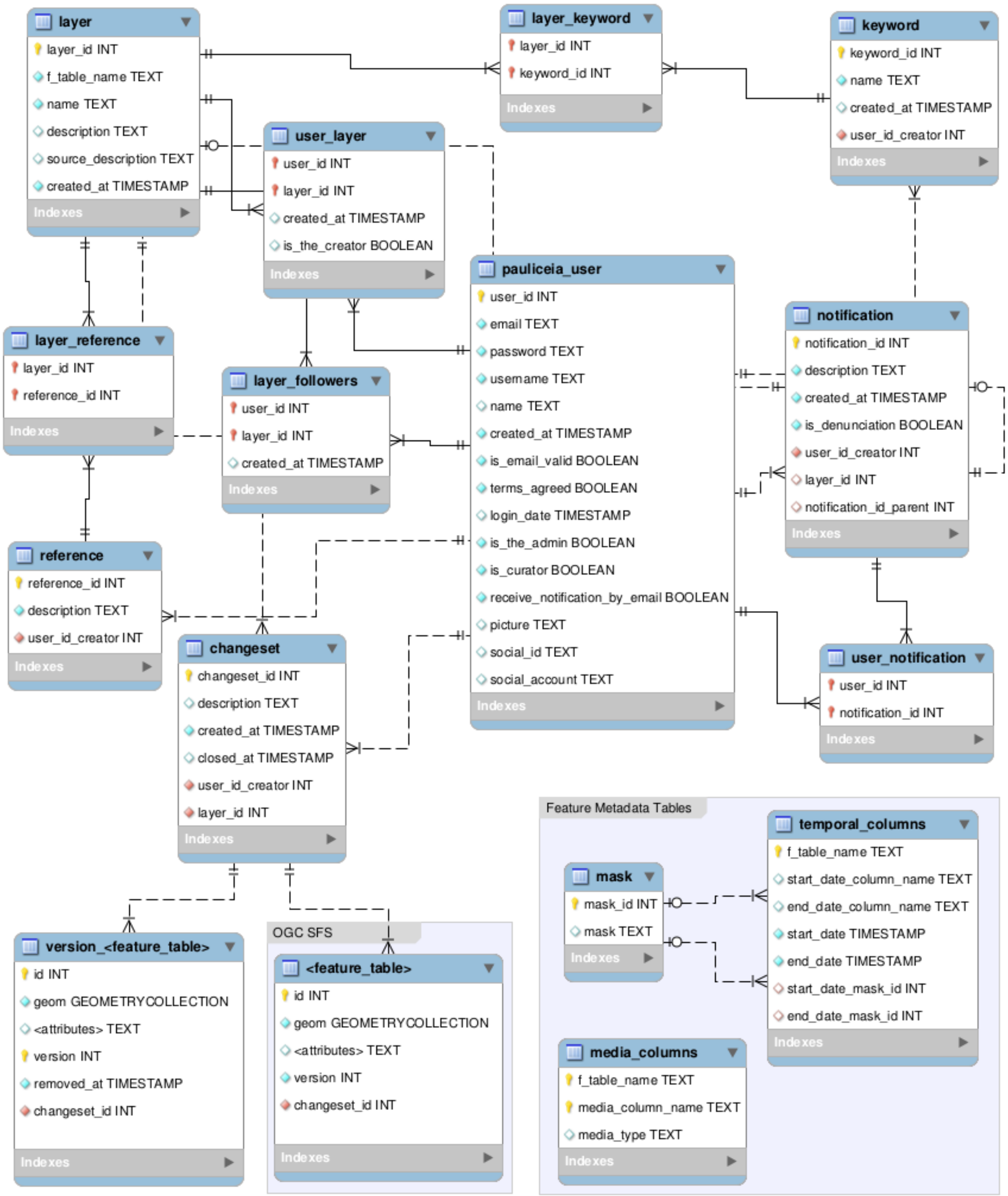
4.4 Describing the VGIMWS

The normal users are able to view data sets, but for data editing, it is necessary to do the log in first, which is the connection created for internal access to the system. This login can be done by the following functions: (1) GET `/auth/google/`, do the social login using a Google account; (2) GET `/auth/facebook/`, do the social login using a Facebook account; or (3) GET `/auth/login/`, do the login using an e-mail and password saved in Pauliceia 2.0 database. Figure 4.6 illustrates the login page.

When a user uses the social login with a Google or Facebook account, it appears a page to put the email and password of the user. If the credentials are correct, so it will redirect to the portal, otherwise, it will appear a message indicating that the email or password is(are) incorrect(s).

Regarding the normal login, with an email and password that are saved in database, the Pauliceia 2.0 platform needs to create a header in request that contains the credentials encrypted. First, the password is encrypted with a secure hash algorithm called SHA512. Then, the credentials are concatenated in one text and this text is encoded in a base64 string. After that, this string is added inside the header. Just now the request of the GET `/auth/login/` function is able to use. All this is done for security questions and theses steps are done by the portal, hence the final user (i.e. who uses the platform) will not see what is happening. The volunteers just insert their email and password inside the text fields of the login page. In Figure 4.5, the user is represented by the “pauliceia_user” table.

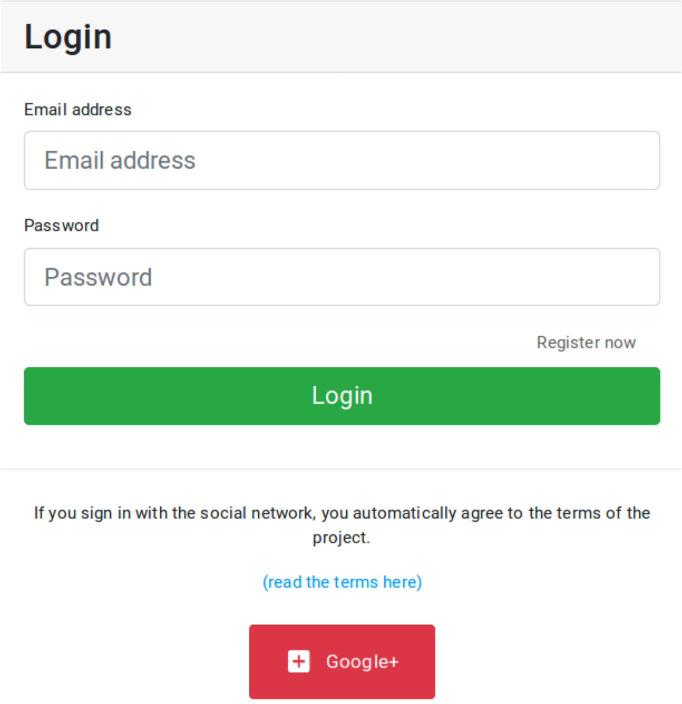
Figure 4.5 - Complete Model of the on going Spatiotemporal Database of Pauliceia 2.0.



SOURCE: Author's production.

In order to organize the historical data sets, they are inserted into layers. So, the user needs to create a new layer. For example: one layer could be the crimes of 1900; another layer, the crimes of 1910 and so on, until 1930. For the creation of a new layer, the function POST `/api/layer/create` is used. The layer can be

Figure 4.6 - Login page of Pauliceia 2.0 project.



Login

Email address

Email address

Password

Password

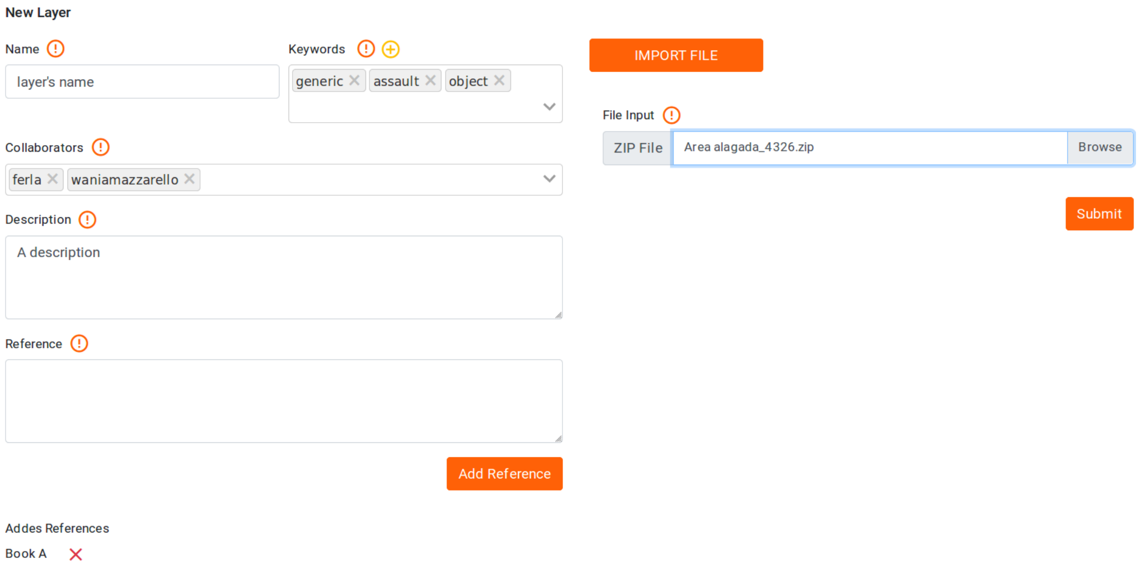
If you sign in with the social network, you automatically agree to the terms of the project.

SOURCE: Author's production.

organized by multiple keywords and references, that can be added through the POST `/api/keyword/create` and POST `/api/reference/create` functions. If the user wants to create an empty layer, the platform creates an empty feature table with the POST `/api/feature_table/create` function, but if the volunteer wants to upload a Shapefile, the POST `/api/import/shp/` function is used. After these processes, the user must inform the temporal metadata of the layer. These metadata are added by the POST `/api/temporal_columns/create` function. Figure 4.7 exposes the page of creating a new layer and the Figure 4.8 shows the page of inserting temporal columns. The users may list the available layers to show them on the map, so it is used the GET `/api/layer` function and it is illustrated by Figure 4.9. On dashboard page, the users may delete their layers, using the DELETE `/api/layer` function, being demonstrated by Figure 4.10. In Figure 4.5, the layer, keyword, reference, feature table and temporal columns are represented by the tables that have their names, where <feature_table> is the name of the created layer.

The concept of changesets is used in the Pauliceia 2.0 portal to control the history of the editions made about the spatiotemporal data. The changeset is represented by

Figure 4.7 - Page of creating a new layer on Pauliceia 2.0 platform.



New Layer

Name: layer's name

Keywords: generic, assault, object

Collaborators: ferla, waniamazzarello

Description: A description

Reference:

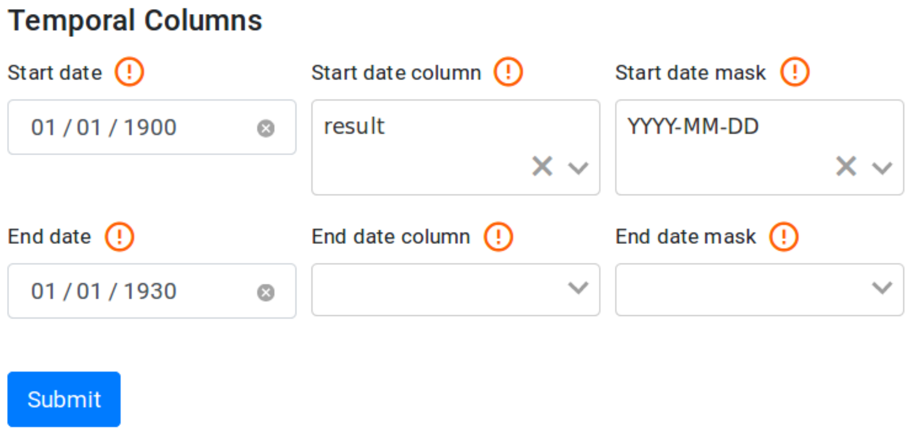
Addes References

Book A

File Input: ZIP File Area alagada_4326.zip

SOURCE: Author's production.

Figure 4.8 - Page of inserting temporal columns on Pauliceia 2.0 platform.



Temporal Columns

Start date: 01 / 01 / 1900

Start date column: result

Start date mask: YYYY-MM-DD

End date: 01 / 01 / 1930

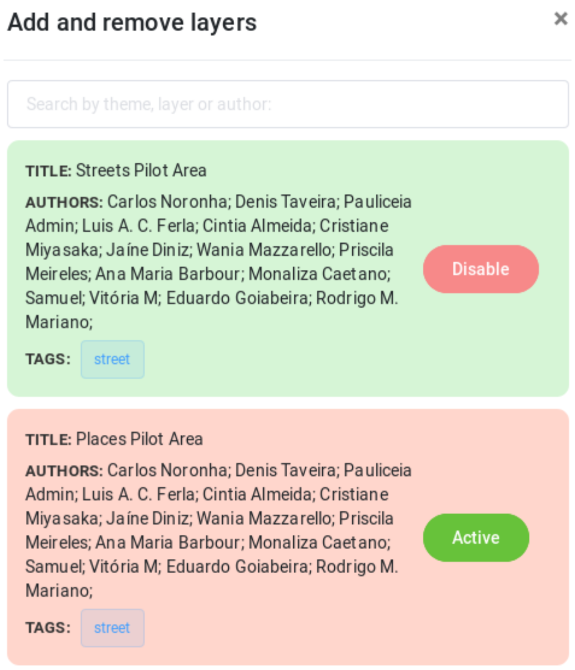
End date column:

End date mask:

SOURCE: Author's production.

the "changeset" table in Figure 4.5. There is not a user interface to the changesets, because it works internally. How it works inside the Pauliceia 2.0 portal is described in the Appendix B. After the layer is generated by the user, internally the system will open a changeset to save the changes that the user will make. The POST `/api/changeset/create` function is used to open a new changeset. So, the user is

Figure 4.9 - Listing available layers on Pauliceia 2.0 platform.



Add and remove layers ×

Search by theme, layer or author:

TITLE: Streets Pilot Area

AUTHORS: Carlos Noronha; Denis Taveira; Pauliceia Admin; Luis A. C. Ferla; Cintia Almeida; Cristiane Miyasaka; Jaíne Diniz; Wania Mazzarello; Priscila Meireles; Ana Maria Barbour; Monaliza Caetano; Samuel; Vitória M; Eduardo Goiabeira; Rodrigo M. Mariano;

TAGS: street

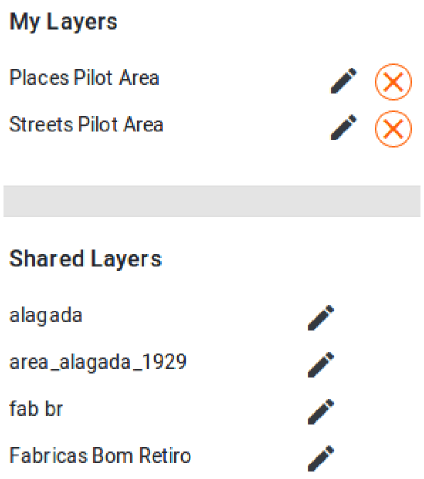
TITLE: Places Pilot Area

AUTHORS: Carlos Noronha; Denis Taveira; Pauliceia Admin; Luis A. C. Ferla; Cintia Almeida; Cristiane Miyasaka; Jaíne Diniz; Wania Mazzarello; Priscila Meireles; Ana Maria Barbour; Monaliza Caetano; Samuel; Vitória M; Eduardo Goiabeira; Rodrigo M. Mariano;

TAGS: street

SOURCE: Author's production.

Figure 4.10 - Page of deleting a layer on Pauliceia 2.0 platform.



My Layers

Places Pilot Area ✎ ⊗

Streets Pilot Area ✎ ⊗

Shared Layers

alagada ✎

area_alagada_1929 ✎

fab br ✎

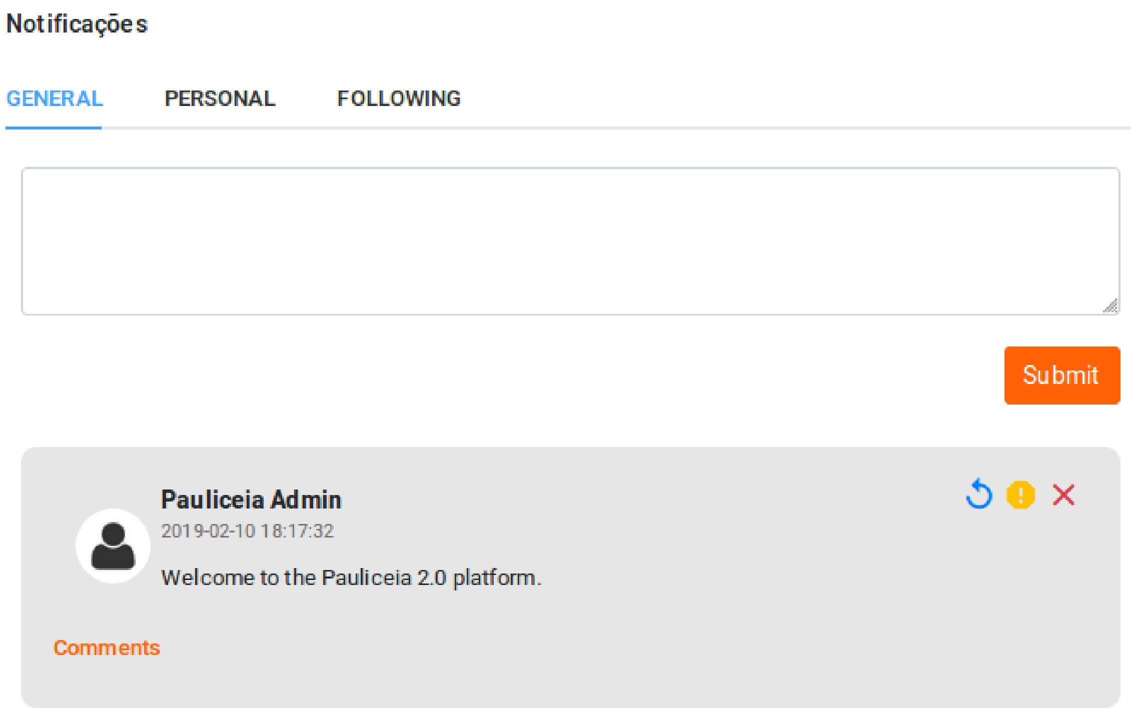
Fabricas Bom Retiro ✎

SOURCE: Author's production.

able to add or make updates in features. After the changes, the changeset needs to be closed. Then, the platform uses the POST `/api/changeset/close` function. A changeset is just deleted when the layer that it is related to is removed. The changes are stored in “version_<feature_table>” table in Figure 4.5.

A notification is a list of comments who a user will receive about the platform (e.g. feedbacks from the community related to the layers). The user can create a new notification about a layer or a general notification through the POST `/api/notification/create` function. The volunteers can delete their notifications using the DELETE `/api/notification` function. Figure 4.11 displays the notification page on dashboard, where the contributors can add or remove their notifications.

Figure 4.11 - Notification page on dashboard of Pauliceia 2.0 project.



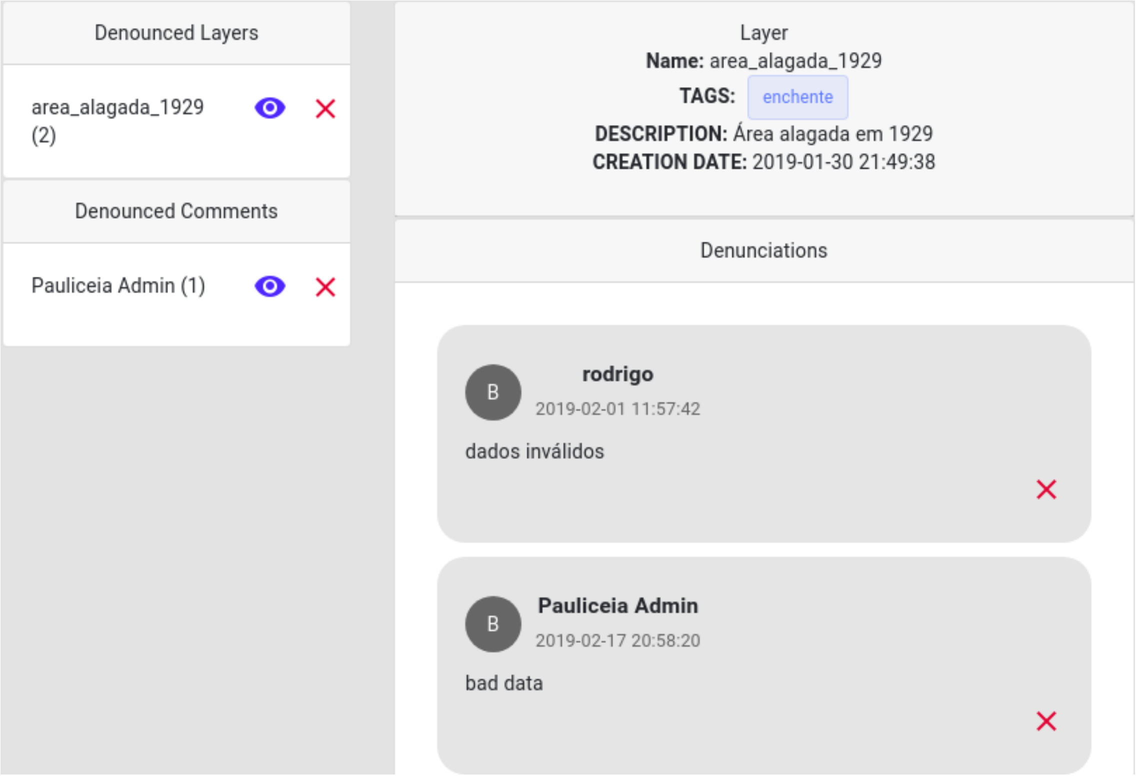
Pauliceia Admin
2019-02-10 18:17:32
Welcome to the Pauliceia 2.0 platform.

SOURCE: Author’s production.

If the users find out an unappropriated content in the platform, they must report it, through the denunciations. Denunciation is a type of notification, because of that, it can be created with the same function that is used to insert a notification. The difference is that the denunciation has a flag in the JSON that indicates that it is not a normal notification (i.e. the flag “is_denunciation” is marked with “True”).

Figure 4.12 shows the denunciation page on administrator dashboard, where the administrator can remove reported layers or notifications.

Figure 4.12 - Denunciation page on amdinistrator dashboard of Pauliceia 2.0 project.



Denounced Layers

area_alagada_1929 (2) ✕

Denounced Comments

Pauliceia Admin (1) ✕

Layer

Name: area_alagada_1929

TAGS: enchente

DESCRIPTION: Área alagada em 1929

CREATION DATE: 2019-01-30 21:49:38

Denunciations

B **rodrigo**
2019-02-01 11:57:42

dados inválidos

✕

B **Pauliceia Admin**
2019-02-17 20:58:20

bad data

✕

SOURCE: Author's production.

These and other functions are described in detail in API documentation inside the Github repository[5].

[5] https://www.github.com/pauliceia/vgiws/blob/master/doc/api/README.md

5 HISTMAPATHON AND PERIOD OF TESTS OF THE PAULICEIA 2.0 PLATFORM

The Pauliceia 2.0 project team held a HistMapathon event with volunteers at the Laboratório de Humanidades Digitais of UNIFESP Guarulhos - Lab.Hum. from 19 to 27 April 2018, with the objective of testing the VGI and crowdsousing concepts for assembling historical data sets for the Pauliceia 2.0 platform; and contributing with historical addresses to improve the historical geocoding. This HistMapathon had as objective to discuss the Pauliceia 2.0 project and its relation with the numbering of the addresses of the São Paulo city. Thus, obtaining collaboration in the construction of the historical address database, through the Pauliceia 2.0 platform based on the documentary sources usually used by the project numbering team.

The event was open to all people, even those who did not have computer or geotechnology skills. After all, one of the objectives was to enable those interested in collaborating with the project to provide an experience that facilitates the contact between people from Humanities to the digital world and the collaborative work, contributing to the platform data.

The dynamic and the development of the journey were carried out by the numbering team members, under the supervision of Professor Dr. Luís Ferla (coordinator of the Pauliceia 2.0 project). Ten vacancies were opened for this workshop, an amount established considering the available infrastructure and regarding more attention to each participant. Seven undergraduates and a public employee of the university attended in HistMapathon.

Each of the seven meetings lasted an hour and a half. The first one was dedicated to the presentation of the Pauliceia 2.0 project, the web platform and the historical geolocator, showing its importance within the project. On the second day was presented a tutorial of data collection in the platform and since then the participants had access to the historical documentary sources. The volunteers were able, with the assistance of the responsible team, to carry out the practical application of the concepts and methodologies discussed in the previous stages, selecting historical addresses to put them on the platform. In the other days, the contributors inserted and updated these historical addresses on the platform by clicking on the historical maps, provided by the web portal, to inform the spatial location of each historical address.

In a general evaluation, the results obtained with the workshops were quite positive.

Quantitatively, participants handled 146 streets, which represent about a third of the pilot area of the project. In a qualitative aspect, in addition to the dissemination of the project, collaboration links was established, where some participants were interested in continuing to contribute to the Pauliceia 2.0 project. Figure 5.1 shows people in this HistMapathon at UNIFESP Guarulhos.

Figure 5.1 - Volunteers at HistMapathon.

SOURCE: Ferla (2019)

During the period from 2 September to 10 October 2018, the manual tests of the Pauliceia 2.0 platform were made. The Pauliceia 2.0 numbering team spent these days monitoring and trying to find problems on the portal. In this period, problems were found regarding the historical address data entered during the HistMapathon. Afterwards, these errors have since been corrected. Also during this period, the Pauliceia 2.0 team indicated several points of improvement of the functionalities and the interface of the platform, as to leave more self-descriptive the error messages.

The errors or improvements reported by the team are described in the Pauliceia 2.0

project wiki: http://www.dpi.inpe.br/pauliceia/doku.php?id=testes. Fixed bugs are marked with an “OK” or a description of the result, while those that are pending are blank. The original errors reported can be found in the list of documents within the wiki.

After the trial period of the platform, its beta version was launched on October 30, 2018 in the Arquivo Público do Estado de São Paulo from 2 pm to 5 pm, with the participation of numerous people that given a lot of advices of how to improve the web portal, such as the possibility of a user upload a historical raster map.

The HistMapathon and the period of tests were crucial to identify issues in the web platform. The volunteers gave feedback about the portal stability, new functionalities and observations that were essential to improve the Pauliceia 2.0 platform and to discover VGI requirements for more events.

6 FINAL REMARKS AND CONCLUSION

Recently, to handle geographic information had become itself evident. Volunteers, that are often untrained, have started to devise, collect and diffuse geographic information through the internet, it is known as VGI. Even assembling geographic information rapidly, meticulous and with small cost, by default VGI does not assure the data quality. This imprecision is caused because the humans have vague concepts of geographical regions. In order to minimize inaccurate data, it is necessary to establish, at least, a quality standard and use some quality indicators, measures, approaches or the definition of a VGI Protocol.

Pauliceia 2.0 project has as objective to build an online computational system for collaborative research of historical data. VGI is used to assemble historical data resulted of research works provided by history researchers. The historians contribute with their spatiotemporal data, producing maps and views of their research. Some types of historical data collection can be through the manual vectorization of old maps (e.g. the streets and buildings), collection of ancient addresses and the acquisition of historical documents associated to places. Then, it is necessary to define quality mechanisms for these data, as the creation of a VGI protocol.

The absence of specification is one reason that may conduct into low data quality, because of that, the NMAs and the CSCs use strict protocols that drive the geographic data collection. Then, it is important to define a VGI protocol to assure the data quality.

Mooney et al. (2016) introduce a generic protocol, to be used to various VGI projects, creating a standardization to handle collaborative geographic data. This protocol improves the data quality of the VGI contributions. For that reason, a VGI protocol was developed to work with historical data, in the context of Pauliceia 2.0 project. VGI protocol, defined by Mooney et al. (2016) have five main steps: initialisation, data collection, self-assessment and quality control, data submission and feedback.

VGI data quality can be characterized by: quality measures, indicators and approaches. Quality measures use authoritative data, as reference, comparing them with the data provided by VGI. Quality indicators are adopted when authoritative data are not accessible and they are qualitative. There are also the quality approaches to help to assure the VGI quality, being defined by Goodchild and Li (2012) as: crowdsourcing, social and geographic approach.

Quality indicators and approaches are used to ensure the VGI data quality of the historical data of the Pauliceia 2.0 project. This is done because the quality measures apply authoritative data sets as reference, what it does not exist for the Pauliceia 2.0 project, at the moment.

The first method to ensure the data quality of the Pauliceia 2.0 data is the definition of a VGI protocol. This protocol describes the topics that the volunteers can do and how they should follow them. This provides a standardization to avoid data heterogeneity, leading to an improvement in quality. This VGI protocol has the main stages: (1) the platform provides a tutorial of how to use the system, so the users need to read it to solve the main questions; (2) the volunteers may create layers about their research data, manually or by bulk import; (3) the collaborators may add trustworthy people in their layers to help them in data collection, generating a local quality; (4) users outside layers, that find problems in data, may do comments and observations through the notifications; (5) historians may promote HistMapathons for large data collection of vectorization of historical maps; (6) the user may receive relevant notifications of the system; (7) if the volunteers discover inappropriate data on the platform, they must report it though denunciations; and (8) the volunteers can provide feedback of their experience on the available channels of the project. The VGI protocol for historical data applied to the Pauliceia 2.0 project, developed here, can be applied to other existing historical VGI projects that does not have authoritative data for comparison. It is possible to use the techniques of quality indicators and approaches described here and to apply them to other VGI projects.

Other quality mechanism is the crowdsourcing approach using the Linux Law, where a group of people is used to validate the data quality through the notifications and denunciations. If the volunteers discover a problem inside a specific layer (e.g. a bad vectorization), they can warn the owners through the notifications. Whether the contributors find out inappropriate content, they must report it through the denunciations.

A web service was developed based on the VGI protocol for historical data, in the context of the Pauliceia 2.0 project, being called VGIMWS. The decision of developing a web service, with well-defined and modularized functions, gives the opportunity of creating a system with the facility and extension of the reuse of the VGI for several types of applications. This provides support to help existing and new VGI-related projects to improve their quality. This service manages the main concepts of Pauliceia 2.0 platform, such as: users, layers, keywords, references, notifications and features, being each resource found through a unique URL. For example: in order to log a user in the system, the portal uses the GET `/auth/login/` function; to create a new layer, the platform uses the POST `/api/layer/create` function and so on.

Changing the VGIMWS configuration files, the developer can apply it to another historical VGI project that manages spatiotemporal data organized in features tables. The configuration files are inside the "settings" folder in VGIMWS source code. The "accounts.py" file has the configuration about social accounts (i.e. Facebook and Google) and the cookie secret. The "db_settings.py" file has the configuration about PostgreSQL connection. Changing these files, it is possible to use the VGIMWS in other server. Remind that is necessary to create a new database in PostgreSQL with the structure described in Figure 4.5. For that, it is possible to use the "clean_db.py" file, that is in "tests/util" folder. This file creates the tables and relationships described in Figure 4.5, being just necessary edit the "db_settings.py" file. All these processes can be read in detail in the API's documentation in Github repository[1].

A HistMapathon event was carried out by the Pauliceia 2.0 project team from 19 to 27 April 2018, at the Laboratório de Humanidades Digitais of UNIFESP Guarulhos - Lab.Hum. The aim of this event was to test the VGI concepts of Pauliceia 2.0 platform. This event was open to the citizens, where the event goals and the web platform were described. During the days, the volunteers received historical documents that contain historical addresses and they added them to the web platform by clicking on the map to inform their spatial location.

The feedbacks that the team received were very positive. The contributors selected 146 streets that represent a third of the project pilot area, and they gave comments and observations related to improve the web portal.

Manual tests were did by the Pauliceia 2.0 numbering team during 2 September to 10 October 2018. During this period, they found some bugs on the platform and they described some points to improve the system, reporting them through the e-mail. Then, these errors were stored in a Wiki and the Pauliceia 2.0 developers fixed them, making the web portal better.

While the contributors were using the Pauliceia 2.0 platform, it worked fine, just a few issues were discovered. The operations of management, related to the Pauliceia 2.0 entities, were did using the VGIMWS. For example: when the users inserted a

[1] https://github.com/Pauliceia/vgiws/blob/master/doc/server/README.md

new layer, the platform called the VGIMWS to add it in the database.

Finally, the beta version of the Pauliceia 2.0 platform was launched for the history community in 30 October 2018.

As future work, it is intended to :

- extend the VGI Protocol for historical data to handle raster data, describing how a user can contribute with historical maps and detailing quality control methods for these data;
- create methods to analyse the historical data provided on the platform. For example: to develop mechanisms for a user to build graphics based on the data or to analyse a specific historical object in the time based on feature tables;
- make avaiable the historical map metadata of Pauliceia 2.0 platform easily. For that, it is needed to evaluate the use of services to publicize metadata of spatial resources, such as GeoNetwork, GeoBlacklight or Pycsw. These metadata will be available using the National Spatial Data Infrastructure of Brazil (INDE) standard. This standard is called Brazilian Geospatial Metadata Profile (MGB profile) and it is based on ISO 19115. Currently, the platform avaiable layers are published on a Geoserver, that serves them using the OGC standards.
- finalize the construction of the algorithm to find the best consensus of several geometries (i.e. lines and polygons). During a HistMapathon, citizens will be able to vectorize old maps, however they may vectorize the same object accidentally. For that reason, it is planned to finish the development of the algorithm that given several representations of the same vector object/feature (e.g. street or building), it will obtain the best polygonal or linear representation (i.e. the best consensus). A similar algorithm is proposed by Budig et al. (2016). It is intended to apply it as a microservice or a specific function of VGIMWS.

REFERENCES

ADAMCZYK, P.; SMITH, P. H.; JOHNSON, R. E.; HAFIZ, M. Rest and web services: in theory and in practice. In: WILDE E.; PAUTASSO, C. E. (Ed.). **REST: from research to practice**. [S.l.]: Springer, 2011. p. 35–57. 26

ANTONIOU, V.; SKOPELITI, A. Measures and indicators of vgi quality: an overview. **ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences**, 2015. 18

BOX, D.; EHNEBUSKE, D.; KAKIVAYA, G.; LAYMAN, A.; MENDELSOHN, N.; NIELSEN, H. F.; THATTE, S.; WINER, D. **Simple Object Access Protocol (SOAP) 1.1**. 2000. Available from: https://www.w3.org/TR/soap11/. Access in: 12 Mar. 2018. 27

BUDIG, B.; DIJK, T. C. van; FEITSCH, F.; ARTEAGA, M. G. Polygon consensus: smart crowdsourcing for extracting building footprints from historical maps. In: ACM SIGSPATIAL INTERNATIONAL CONFERENCE ON ADVANCES IN GEOGRAPHIC INFORMATION SYSTEMS, 24., 2016. **Proceedings...** [S.l.]: ACM, 2016. p. 66. 23, 31, 66

CARVALHO, D. F. **Café, ferrovias e crescimento populacional: o florescimento da região noroeste paulista**. 2007. Available from: http://www.historica.arquivoestado.sp.gov.br/materias/anteriores/edicao27/materia02/. Access in: 04 Apr. 2017. 11

CECHANOWICZ, J.; GUTWIN, C.; BROWNELL, B.; GOODFELLOW, L. Effects of gamification on participation and data quality in a real-world market research domain. In: INTERNATIONAL CONFERENCE ON GAMEFUL DESIGN, RESEARCH AND APPLICATIONS, 1., 2013. **Proceedings...** [S.l.]: ACM, 2016. p. 58–65. 20

CHASE, N. **Understanding web services specifications, Part 1**. 2006. Available from: https://www.ibm.com/developerworks/webservices/tutorials/ws-understand-web-services1/ws-understand-web-services1.html. Access in: 12 Mar. 2018. 26

CHINNICI, R.; MOREAU, J.-J.; RYMAN, A.; WEERAWARANA, S. **Web Services Description Language (WSDL) Version 2.0 Part 1: Core**

Language. 2007. Available from: https://www.w3.org/TR/2007/REC-wsdl20-20070626/. Access in: 12 Mar. 2018. 27

CREATIVE COMMONS. **About the licenses**. 2018. Available from: https://creativecommons.org/licenses/. Access in: 13 May 2018. 32

DIGITAL HARLEM BLOG. **The project**. 2017. Available from: https://digitalharlemblog.wordpress.com/about/the-project/. Access in: 18 Sept. 2017. 21

DOCKER. **Get started, part 1: orientation and setup**. 2019. Available from: https://docs.docker.com/get-started/. Access in: 16 Jan. 2019. 28

ESRI. **GIS Dictionary**. 2018. Available from: https://support.esri.com/en/other-resources/gis-dictionary/term/layer. Access in: 13 May 2018. 36

ESTELLÉS-AROLAS, E.; GONZÁLEZ-LADRÓN-DE-GUEVARA, F. Towards an integrated crowdsourcing definition. **Journal of Information Science**, v. 38, n. 2, p. 189–200, 2012. 1

FERLA, L. A. C. **Re: Autorização de foto do Mapathon na dissertação**. 2019. [mensagem pessoal]. Mensagem recebida por ferla@unifesp.br em 18 abr. 2019. 60

FERREIRA, K. R.; FERLA, L.; QUEIROZ, G. R. de; VIJAYKUMAR, N. L.; NORONHA, C. A.; MARIANO, R. M.; WASSEF, Y.; TAVEIRA, D.; DARDI, I. B.; SANSIGOLO, G.; GUARNIERI, O.; MUSA, D. L.; ROGERS, T.; LESSER, J.; PAGE, M.; BRITT, A. G.; ATIQUE, F.; SANTOS, J. Y.; MORAIS, D. S.; MIYASAKA, C. R.; ALMEIDA, C. R. de; NASCIMENTO, L. G. M. do; DINIZ, J. A.; SANTOS, M. C. dos. Pauliceia 2.0: a computational platform for collaborative historical research. In: GEOINFO, 18., 2017, Salvador, BA. **Anais...** [S.l.], 2017. p. 28–39. 12, 13

FERREIRA, K. R.; FERLA, L.; QUEIROZ, G. R. de; VIJAYKUMAR, N. L.; NORONHA, C. A.; MARIANO, R. M.; TAVEIRA, D.; SANSIGOLO, G.; GUARNIERI, O.; ROGERS, T.; LESSER, J.; PAGE, M.; ATIQUE, F.; MUSA, D. L.; SANTOS, J. Y.; MORAIS, D. S.; MIYASAKA, C. R.; ALMEIDA, C. R. de; NASCIMENTO, L. G. M. do; DINIZ, J. A.; SANTOS, M. C. dos. A platform for collaborative historical research based on volunteered geographical information. **Journal of Information and Data Management**, v. 8, n. 1, 2018. 12

FIELDING, R. T.; TAYLOR, R. N. **Architectural styles and the design of network-based software architectures**. 2000. 180 p. Thesis (Doctor in Computer Science) — University of California, Irvine, 2000. 27

FLANAGIN, A. J.; METZGER, M. J. The credibility of volunteered geographic information. **GeoJournal**, v. 72, n. 3-4, p. 137–148, 2008. 19

GEOSERVER. **What is Geoserver?** 2019. Available from: `http://geoserver.org/about/`. Access in: 16 Jan. 2019. 50

GIRRES, J.-F.; TOUYA, G. Quality assessment of the french openstreetmap dataset. **Transactions in GIS**, v. 14, n. 4, p. 435–459, 2010. 8

GOMES, R. **Docker para desenvolvedores**. [S.l.]: Leanpub, 2017. 28

GOODCHILD, M. F. Citizens as sensors: the world of volunteered geography. **GeoJournal**, v. 69, n. 4, p. 211–221, 2007. 1, 7

GOODCHILD, M. F.; LI, L. Assuring the quality of volunteered geographic information. **Spatial Statistics**, v. 1, p. 110–120, 2012. 1, 2, 7, 18, 20, 29, 41, 63

GOOGLE. **Google map maker has closed**. 2018. Available from: `https://support.esri.com/en/other-resources/gis-dictionary/term/layer`. Access in: 06 June 2018. 2

GROSSO, E.; PLUMEJEAUD, C.; PARENT, B. Geopeuple project: using restful web api to disseminate geohistorical database as open data. In: OPEN SOURCE GEOSPATIAL RESEARCH AND EDUCATION SYMPOSIUM, 2012, Yverdon-les-Bains, Switzerland. **Proceedings...** [S.l.], 2012. p. 222–228. 22

HAKLAY, M.; BASIOUKA, S.; ANTONIOU, V.; ATHER, A. How many volunteers does it take to map an area well? the validity of linus' law to volunteered geographic information. **The Cartographic Journal**, v. 47, n. 4, p. 315–322, 2010. 41

HAMARI, J.; KOIVISTO, J.; SARSA, H. Does gamification work?–a literature review of empirical studies on gamification. In: HAWAII INTERNATIONAL CONFERENCE ON SYSTEMS SCIENCES (HICSS), 47., 2014. **Proceedings...** [S.l.]: IEEE, 2014. p. 3025–3034. 20, 21

HERRING, J. Opengis implementation specification for geographic information-simple feature access-part 2: Sql option. **Open Geospatial Consortium**, 2006. 37, 46, 47

HOLLENSTEIN, L.; PURVES, R. Exploring place through user-generated content: using flickr tags to describe city cores. **Journal of Spatial Information Science**, v. 2010, n. 1, p. 21–48, 2010. 8

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION (ISO). **ISO/TC 211 Geographic information/Geomatics**. Vernier, Geneva, Switzerland, june 2009. 53 p. 18

JAKOBSSON, A.; GIVERSEN, J. **Guidelines for implementing the ISO 19100 geographic information quality standards in national mapping and cadastral agencies**. [S.l.]: Eurogeographics Expert Group on Quality, 2007. 18

KATRAGADDA, L.; JAIN, S. **Making your mark on the world**. 2008. Available from: https://maps.googleblog.com/2008/06/making-your-mark-on-world.html. Access in: 06 June 2018. 2

KOWAL, K. C.; IDAL, P. P. Online georeferencing for libraries: the british library implementation of georeferencer for spatial metadata enhancement and public engagement. **Journal of Map & Geography Libraries**, v. 8, n. 3, p. 276–289, 2012. 22

LIMA, P. de; SANTOS, A. d. P. dos; LISBOA FILHO, J. Estudo sobre infraestrutura de dados espaciais para embasar a proposta de desenvolvimento de uma ide para a universidade federal de viçosa. **Revista Eletrônica: Tempo - Técnica - Território**, v. 9, n. 2, p. 7–27, August 2018. 10, 13

MOONEY, P.; MINGHINI, M.; LAAKSO, M.; ANTONIOU, V.; OLTEANU-RAIMOND, A.-M.; SKOPELITI, A. Towards a protocol for the collection of vgi vector data. **ISPRS International Journal of Geo-Information**, v. 5, n. 11, p. 217, 2016. 2, 8, 15, 16, 17, 29, 31, 42, 43, 63

MOTA, P. d. B. et al. **A cidade de São Paulo de 1870 a 1930: café, imigrantes, ferrovia, indústria**. 2007. 181 p. Dissertação (Mestrado em Urbanismo) — Pontifícia Universidade Católica de Campinas, Campinas, 2007. 11

MULLIGAN, G.; GRAČANIN, D. A comparison of soap and rest implementations of a service based interaction independence middleware framework. In: WINTER SIMULATION CONFERENCE, 2009. **Proceedings...** [S.l.], 2009. p. 1423–1432. 27

MUMBAIKAR, S. et al. Web services based on soap and rest principles. **International Journal of Scientific and Research Publications**, v. 3, n. 5, p. 1–4, 2013. 26, 27

NORONHA, C. A. **Re: Autorização para inserção de imagem da arquitetura do Docker**. 2019. [mensagem pessoal]. Mensagem recebida por beto_noronha@live.com em 20 mar. 2019. 49

OPEN GEOSPATIAL CONSORTIUM. **OGC® standards and supporting documents**. 2019. Available from: https://www.opengeospatial.org/standards/. Access in: 20 Jan. 2019. 13

OPENSTREETMAP. **About**. 2017. Available from: https://www.openstreetmap.org/about. Access in: 02 Sept. 2017. 2, 24

______. **About OpenStreetMap**. 2017. Available from: http://wiki.openstreetmap.org/wiki/About_OpenStreetMap. Access in: 01 Sept. 2017. 2

______. **Mapathon**. 2017. Available from: http://wiki.openstreetmap.org/wiki/Mapathon. Access in: 18 Sept. 2017. 31

______. **Quality assurance**. 2017. Available from: http://wiki.openstreetmap.org/wiki/Quality_assurance. Access in: 25 Sept. 2017. 24

______. **API**. 2018. Available from: https://wiki.openstreetmap.org/wiki/API. Access in: 20 Jan. 2018. 3, 23, 24

______. **API v0.6**. 2018. Available from: https://wiki.openstreetmap.org/wiki/API_v0.6. Access in: 20 Jan. 2018. 3, 23, 24

______. **Changeset**. 2018. Available from: https://wiki.openstreetmap.org/wiki/Changeset. Access in: 16 Jan. 2018. 38, 79

______. **Open historical map**. 2018. Available from: https://wiki.openstreetmap.org/wiki/Open_Historical_Map. Access in: 05 Mar. 2018. 3, 23

OVER, M.; SCHILLING, A.; NEUBAUER, S.; ZIPF, A. Generating web-based 3d city models from openstreetmap: the current situation in Germany. **Computers, Environment and Urban Systems**, v. 34, n. 6, p. 496–507, 2010. 18

PORTAINER. **Portainer documentation**. 2019. Available from: https://portainer.readthedocs.io/en/stable/. Access in: 16 Jan. 2019. 50

RAYMOND, E. S. **Release early, release often**. 2017. Available from: http://www.catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/ar01s04.html. Access in: 01 Sept. 2017. 41

REDMOND, T.; SMITH, M.; DRUMMOND, N.; TUDORACHE, T. Managing change: an ontology version control system. In: OWL: EXPERIENCES AND DIRECTIONS INTERNATIONAL WORKSHOP, 5., 2008. **Proceedings...** [S.l.], 2008. 38

ROBERTSON, S. Digital mapping as a research tool: digital harlem: everyday life, 1915–1930. **The American Historical Review**, v. 121, n. 1, p. 156–166, 2016. 21

SCHIERMEIER, Q. **Data management made simple**. 2018. Available from: https://www.nature.com/articles/d41586-018-03071-1. Access in: 08 Aug. 2018. 9

SEE, L. et al. Crowdsourcing, citizen science or volunteered geographic information? the current state of crowdsourced geographic information. **ISPRS International Journal of Geo-Information**, v. 5, n. 5, p. 55, 2016. 1, 7

SENARATNE, H.; MOBASHERI, A.; ALI, A. L.; CAPINERI, C.; HAKLAY, M. A review of volunteered geographic information quality assessment methods. **International Journal of Geographical Information Science**, v. 31, n. 1, p. 139–167, 2017. 2, 7, 8, 18, 19

SIGURBJÖRNSSON, B.; ZWOL, R. V. Flickr tag recommendation based on collective knowledge. In: INTERNATIONAL CONFERENCE ON WORLD WIDE WEB, 17., 2008. **Proceedings...** [S.l.]: ACM, 2008. p. 327–336. 2

SÃO PAULO. SECRETARIA MUNICIPAL DE URBANISMO E LICENCIAMENTO. **História demográfica do município de São Paulo**. 2017. Available from: http://smul.prefeitura.sp.gov.br/historico_demografico/tabelas/pop_brasil.php. Access in: 04 Apr. 2017. 11

TECH2. **Why is Google's Mapathon in hot waters in India? all you need to know**. 2014. Available from: http://www.firstpost.com/tech/news-analysis/why-is-googles-mapathon-in-hot-waters-in-india-all-you-need-to-know-3655197.html. Access in: 18 Sept. 2017. 31

TOBLER, W. On the first law of geography: a reply. **Annals of the Association of American Geographers**, v. 94, n. 2, p. 304–310, 2004. 20

WHITE, J. W.; GILBERT, H. **Laying the Foundation**. [S.l.]: Purdue University Press, 2016. 4, 21

WIKIMAPIA. **About Wikimapia**. 2018. Available from: http://wikimapia.org/about/. Access in: 06 June 2018. 2

WILKINSON, M. D. et al. The fair guiding principles for scientific data management and stewardship. **Scientific Data**, v. 3, n. 160018, 2016. Available from: <https://www.nature.com/articles/sdata201618>. Access in: 15 Mar. 2019. 9, 10

APPENDIX A - USE TERMS

The appendix shows the Pauliceia 2.0 Use Term available on the online platform in http://www.pauliceia.dpi.inpe.br/portal/login in "(read the terms here)". The following lines describe in full the Term:

"We of the Pauliceia team made a brief and clear Term to facilitate its reading and understanding.

SECTION 1 - WHAT DO WE DO WITH YOUR INFORMATION?

When you register on the platform, we store your name and email.

When you browse our portal, we obtain information about your browser or operating system.

All data you upload by platform is stored in the project database and any other user can view it later.

Email notifications are sent with your permission. We can send you system notification emails, such as general notifications, related to your layers or layers that you follow.

SECTION 2 - INFORMATION

The information of the historical data stored by the users of the platform, notifications, denunciations, descriptions, among other, do not necessarily express the opinion of the Pauliceia 2.0 team. They are only of responsibility of the users who have entered this information into the system.

No data with copyright should be inserted in the platform or data with inappropriate content. In this case, the project team will delete the data from the platform and the user will lose access. For this reason, the portal will provide a complaint feature. If any user finds inappropriate content, he or she should inform it through the denunciations on the platform.

In relation to layers, they may have multiple keywords associated with it. A curator user can add or remove keywords, or edit information in a layer, to improve or organize it.

SECTION 3 - CONSENT

From the moment you provide us with your personal information to access the portal (name and email) and enter information on the platform, such as layers of historical data, notifications or denunciations, you are agreeing to the terms described here.

If you wish to revoke your consent, for any reason, after providing your data; you can remove them at any time. The platform provides tools for a user deletes at any time all the elements that have been entered.

SECTION 4 - DISCLOSURE

We will only share your personal information if we are, for any reason, required by law to do it.

SECTION 5 - THIRD PARTY SERVICES

If you choose to enter the system with a social login, you will have to give us permission to access your social account in order to gain access to our platform by this method.

SECTION 6 - SAFETY

To protect your personal information, we take precautions and follow practices to make sure they will not lost, accessed or disclosed.

Your password is encrypted to prevent malicious software on the network from discovering it. While no Internet method is 100% secure, we follow practices to avoid any problems.

The platform may use cookies to store some information in your browser, so that they can assist us in the procedures we may need to do, such as validating a user within the system.

SECTION 7 - PUBLIC DATA

Platform data are public. Soon, anyone can view or download the data. The only restriction is regarding editing the data. Only the creator and collaborators of a historical data layer can edit their content.

All data for the Pauliceia project are available under the Creative Commons Attribution-ShareAlike 4.0 license (https://creativecommons.org/licenses/by-sa/4.0/), which basically allows people to copy, distribute, modify and use the platform information for any proposal, as long as you give credits to the Pauliceia project and its collaborators. If the person downloads and updates the system data, he or she must use the same license. A license summary can be described as: “This license lets others remix, tweak, and build upon your work even for commercial purposes, as long as they credit you and license their new creations under the identical terms. This license is often compared to “copyleft” free and open source software licenses. All new works based on yours will carry the same license, so any derivatives will also allow commercial use. This is the license used by Wikipedia, and is recommended for materials that would benefit from incorporating content from Wikipedia and similarly licensed projects”

SECTION 8 - CHANGES IN TERMS

We have the right to update these terms at any time. For this reason, read it frequently. If we do any editing on these terms, we will warn you of this. So, you will know what information we collect and how we use it.

SECTION 9 - CONTACT

If you have any questions regarding the terms, use of the platform, information, complaints or compliments, you may contact us at the following email: “pauliceia_support@googlegroups.com” ”

APPENDIX B - CHANGESET

A changeset is a set of changes did by a user in the platform (OPENSTREETMAP, 2018c). A changeset can include adding, updating and removing of features related to a layer in the Pauliceia 2.0 portal. This appendix will describe how a changeset works internally in the server.

In order to exemplify, it supposes that a user creates an empty layer called "Street", so it is generated a feature table with the same name ("Street") and a version feature table called "Street Version". A feature table is a special table that stores the layer features. The version feature table saves the history of a feature. The user added just one column in his or her layer, the column "name". With that created layer, internally the system add three more columns to control, the "id", "changeset_-id" and "version". The "ìd" column is the primary key of the table, a column that identifies the feature, it is automatically generated. The "changeset_id" stores the id of the changeset created to save the changes of the feature. The "version" column conserves the feature version. The version feature table contains a "removed" column that indicates if the feature was removed or not. The geometry column in both tables was omitted to better exemplify, but the modifications that will be illustrated can be applied to it as well.

The tables B.1 and B.2 show the first state of the feature tables after a layer is created. Hence, they are empty.

Table B.1 - Empty Street Table.

id	**name**	**changeset_id**	**version**

Table B.2 - Empty Street Version Table

id	**name**	**changeset_id**	**version**	**removed**

Now, it is supposed that the user includes a new feature in his or her layer, with the name "street são francisco". One record will be added in the feature table and the version feature table will be empty. The adding is done using the changeset 1 and how he or she just included the new feature, its version is 1. The tables B.3 and B.4

illustrate this adding.

Table B.3 - Adding one feature in Street Table.

id	name	changeset_id	version	
1	street são francisco	1	1	**current version**

Table B.4 - Empty Street Version Table

id	name	changeset_id	version	removed

Then, it is deemed that the user updates that record, changing the name to "street santa edwiges". For that, the user uses a new changeset (e.g. number 2). The version column is incremented by 1. The old feature is sent to the version feature table. The tables B.5 and B.6 show this modification.

Table B.5 - Updating a feature in Street Table.

id	name	changeset_id	version	
1	street santa edwiges	2	2	**current version**

Table B.6 - Put the old version in Street Version Table

id	name	changeset_id	version	removed	
1	street são francisco	1	1	false	**old version**

It is supposed that the user changes the name again, now to "street santa mônica". He or she uses a new changeset (e.g. number 3) and the version column is incremented by 1. The old feature is sent to the version feature table. The tables B.7 and B.8 illustrate this updating.

Now the user wants to delete his or her feature using the changeset number 4. The old feature is sent to the version feature table and a new record is stored in this table

Table B.7 - Updating a feature in Street Table.

id	name	changeset_id	version	
1	street santa mônica	3	3	**current version**

Table B.8 - Put the old version in Street Version Table

id	name	changeset_id	version	removed	
1	street são francisco	1	1	false	
1	street santa edwiges	2	2	false	**old version**

as well. The new record contains the current state of the feature, that is deleted, using the new changeset and with the version incremented by 1. The removed column of the feature is true now, because it is its current state. The tables B.9 and B.10 show this action.

Table B.9 - Empty Street Table.

id	name	changeset_id	version

Table B.10 - Last state of Street Version Table

id	name	changeset_id	version	removed	
1	street são francisco	1	1	false	
1	street santa edwiges	2	2	false	
1	street santa mônica	3	3	false	**old version**
1	street santa mônica	4	4	true	**current version**

The version feature table stores all the history of the feature table (i.e. layer), with all the changes, since the first adding, until the final deleting.