



Universidade do Minho

Escola de Engenharia

Departamento de Informática

Vitor Duarte Loureiro Afonso da Rocha Armada

Interactive Tagging of Photographs in Cooperative Systems

February 2016



Universidade do Minho

Escola de Engenharia

Departamento de Informática

Vitor Duarte Loureiro Afonso da Rocha Armada

Interactive Tagging of Photographs in Cooperative Systems

Master dissertation

Master Degree in Software Engineering

Dissertation supervised by

Pedro Rangel Henriques

Nuno Oliveira

February 2016

ABSTRACT

This document presents a dissertation for a master thesis in Software Engineering, in the area of *Interactive Tagging of Photographies in Cooperative Systems*.

This thesis is a component of the second year of the Masters degree and will be achieved at the Universidade do Minho in Braga, Portugal.

The main purpose of this thesis is to study a dynamic collaborative system that can help a community to identify entities in photographs. The thesis aims at applying the newly Web 2.0 technologies [1] in an online cooperative platform. This allows users to interact and collaborate with each other in a social media as creators of user-generated content, in contrast to Web sites where people are limited to the passive viewing of content.

CONTENTS

Contents i

1	INTRODUCTION	1
1.1	Context	1
1.2	Motivation and Objectives	2
1.3	Research Hypothesis	2
1.4	Contributions	3
1.5	Document Structure	3
2	STATE OF THE ART	4
2.1	Semantic Web	4
2.2	Crowd sourcing on the Web	5
2.3	Online Social Networks and Interactive Tagging	7
2.4	Interactive Tagging in Cooperative Systems	8
2.5	Identification of Entities	9
2.6	Ontologies In Cooperative Systems	11
2.7	Existing Solutions	13
2.8	Summary	14
3	FOTOBORA, PROPOSED SOLUTION	15
3.1	Requirements	15
3.2	Proposed Architecture	18
3.3	Data Model	19
3.4	Ontologies Analysis	22
3.4.1	DBPedia	22
3.4.2	Protégé Ontologies	24
3.4.3	Summary	26
4	FOTOBORA IMPLEMENTATION	27
4.1	Application Framework	27
4.2	Social Collaborative System	29
4.3	Identification of Entities in Photographs	32
4.3.1	Tagging of Entities	32
4.3.2	Identification	34
4.4	Ontology and Entity Types	35
4.5	Poll and Voting	37
4.6	Exporting the Photo Information as PDF Document	38

Contents

4.7	Import Annotations	39
4.8	Genealogical Trees	42
5	CASE STUDIES	45
5.1	Case Study 1	45
5.2	Case Study 2	57
6	CONCLUSION	66
A	FOTOBORA FEATURES	73
A.1	Photograph Display	73
A.2	Tagging of entities	76
A.3	Voting on the entity type and their properties	79
A.4	Polling of new values to an entity's property	84
A.5	Contribute with stories and comments	86
A.6	Exporting the information identified in the Photograph	88
A.7	Associate different ontologies	93
A.8	Import Annotations	94
A.9	Support Genealogical Trees	95

LIST OF FIGURES

Figure 1	Tagging of a Person on a social media website.	7
Figure 2	Schema of a Cooperative tagging system.	8
Figure 3	A fragment of a fruit ontology.	9
Figure 4	Example of an Academic Ontology. [22]	11
Figure 5	Example of tagging of an Animal.	12
Figure 6	Fotobora Use Case Model.	17
Figure 7	Fotobora Architecture (block diagram)	19
Figure 8	Data Model.	21
Figure 9	Example of a branch of Animal in the DBpedia Ontology.	22
Figure 10	Example of a branch of Person in the DBpedia Ontology.	23
Figure 11	Example of a branch of Place in the DBpedia Ontology.	23
Figure 12	MVC Process	28
Figure 13	Sign In Page	30
Figure 14	Sign In Page	31
Figure 15	Search Results	32
Figure 16	Tagging	33
Figure 17	Annotations Schema	41
Figure 18	Vitor's Family Picture	46
Figure 19	Vitor's Family Picture Photograph Page	46
Figure 20	Example of Amadeu using Fotobora's search engine	47
Figure 21	Vitor's Family Picture Photograph Page (Part 1)	48
Figure 22	Vitor's Family Picture Photograph Page (Part 2)	49
Figure 23	Vitor's Family Picture Photograph Page	50
Figure 24	Vitor's Family Picture Photograph Page	51
Figure 25	Vitor's Family Picture Photograph Page	52
Figure 26	Vitor's Family Picture Photograph Page	52
Figure 27	Vitor's Family Picture Photograph Page	53
Figure 28	Vitor's Family Picture Photograph Page	54
Figure 29	Vitor's Family Photograph Report (Part 1)	55
Figure 30	Vitor's Family Photograph Report (Part 2)	56
Figure 31	Vitor's Family Photograph Report (Part 3)	57
Figure 32	Microscopic picture of a blood sample	58
Figure 33	Blood sample upload	59

List of Figures

Figure 34	Tagging of a stem cell	60
Figure 35	Property fill of the stem cell	60
Figure 36	Initial display of property poll of the stem cell	61
Figure 37	Polling of the stem cell's properties	62
Figure 38	Polling of the cancer stem cell's properties	63
Figure 39	New property of the cancer stem cell	64
Figure 40	Story sharing and comment dispute	65
Figure 41	Photo creation page	74
Figure 42	Photograph display page	75
Figure 43	Edit Photo	76
Figure 44	Identify Entity Page	77
Figure 45	Identify Entity : selecting the entity Type	77
Figure 46	Identify Entity Page	78
Figure 47	Identify Entity Page	79
Figure 48	New Entity Type Page	80
Figure 49	New Entity Type example	82
Figure 50	New Entity type submission	83
Figure 51	Score example on poll	84
Figure 52	Upvote or Downvote on Poll	85
Figure 53	Voting Page	86
Figure 54	Voting Page	87
Figure 55	New Value submission	88
Figure 56	Example of Poll Reordering	89
Figure 60	Example of the Convert to PDF Functionality	89
Figure 57	Story Page	90
Figure 58	Story Page	90
Figure 59	Story Page	91
Figure 61	Example of the Convert to PDF Functionality	91
Figure 62	Example of the Convert to PDF Functionality	92
Figure 63	Import Annotations Page	94
Figure 64	Import Annotations - Example of a Page Error	95
Figure 65	Family Tree Page	96
Figure 66	Family Tree Error Example	97
Figure 67	Family Tree Example	97

LIST OF TABLES

Table 1	Properties used on Fotobora	36
---------	-----------------------------	----

LIST OF LISTINGS

4.1	Grammar Context Independant of the Genealogical Tree	42
4.2	JSON example of the Genealogical Tree	42
A.1	Fotobora's Ontology in a Ruby Structured Seed File	93

INTRODUCTION

This document introduces a master work, in the area of *Interactive Tagging of Photographies in Cooperative Systems*. The development of a thesis and its presentation in the format of a dissertation is a component of the second year of the Master Degree in Software Engineering that is held at University of Minho in Braga, Portugal.

1.1 CONTEXT

With the dawn of the Semantic Web and Web 2.0, Crowdsourcing [2] has become one of the ways in which the Social Web [7] has been used for collaborative efforts, particularly in the last few years [4]. Modern web applications [3] have the capabilities for crowdsourcing techniques, and consequently the term is now used exclusively for web-based activity. In this particular case, the term branches into a more specific part of Crowdsourcing, labeled as Crowdvoting [6].

With the evolution of the Web, social web has become a bigger part of most people's lives, and so people have become more reliant on using these systems. People use these systems to share most of their common activities, accomplishments and even photographies, and hardening in order to create a social network. Though most of these communities share content that the original user knows about [5], the need however comes when the user knows little to nothing about the pictures he has.

This requirement is sometimes is satisfied through the use of different social media websites or through Forum posting, in which different users input information regarding the picture that has been submitted by the original poster.

In an attempt to tag entities in a photograph, different companies such as Facebook do provide with a way to identify people and associate them with the user that is tagging friend's list. This is not only restricted to the people that have an account on Facebook, but also with the group that the user is associated to.

There is also not any crowdsourcing and not any level of certainty associated with who tags what, and there is also no real voting towards any of the tagged entities in most other applications.

1.2. Motivation and Objectives

Furthermore most of the entities that are tagged in photographs are limited to be a Person and not necessarily anything else. An Ontology does come in need to associate with the names and properties of these entities, which allows a much extended utility when it comes to tagging entities in photographs.

1.2 MOTIVATION AND OBJECTIVES

Currently a bigger part of the society relies on the use of Crowd sourcing to obtain information regarding photographs. This information however is generally obtained through a variety of comments from different users in a less controlled environment.

With that in mind, this master work consists on the development of a platform that allows users to load and publish a photo. The photo is required to have some contextual information so that the other users can identify (by tagging) a person or an object like a monument, a pet, an artifact or any other entity classifiable.

However such user added information can be incorrect. This raises the need for a Crowdvoting system where users collaborate or contest information with a certain level of assurance. Altogether this provides a more consensual vision about that entity.

Furthermore the project also aims at including a feature to allow the creation of stories related to each entity or the photo itself, so that the users can express and share and even agree or disagree with the information that is being shared. The association of entities identified in the photo with an ontology is also another planned feature. Additional features like a genealogical tree, are also in order, so that users can collaborate by identifying the correlation between people tagged in, for example, a very old family picture.

In this context, the main propose of this thesis is to study the applicability of the Web 2.0 Technologies in order to develop an application that allows users to tag and collaborate about entities identified in pictures. This application will be developed in Ruby on Rails.

1.3 RESEARCH HYPOTHESIS

In the context of this master thesis it has been proved through research and development that the current web technologies can be useful to construct a collaborative system of entity identification in photographs. With this system it has been also possible to allow its integration with an ontology, expanding its possibilities and allowing a more advanced and generic method of identification.

1.4. Contributions

1.4 CONTRIBUTIONS

This dissertation contributed with the development of a collaborative tool named as Fotobora. It is available publicly for any user that wishes to share photographs or contribute with their input in the identification of entities of other photographs submitted by other users. Its source code is also available publicly for the scientific community or for further future development.

1.5 DOCUMENT STRUCTURE

This document is structured in different chapters.

This chapter was an introduction to the document, describing the purposes and objectives of this master work. In chapter 2 , it is introduced the area of Semantic Web, Cooperative Systems and Crowdsourcing. The second section of the chapter describes more in detail the existing tools and methods that are being used. The chapter ends with an overview of the tools being used in order to achieve the development of an improved application which will be the object of study in this thesis. The methodology related to the development of the application will be described in Chapter 4.

STATE OF THE ART

In this chapter it is introduced the area of Semantic Web and Cooperative Systems, it will demonstrate why the semantic web and the Web 2.0 are the foundations of this work and also to describe the different solutions to the problem of collaborative photo tagging as well as the current state of Crowdsourcing.

2.1 SEMANTIC WEB

"The Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation."[18]

The second generation of the World Wide Web is focused on the ability for people to collaborate and share information online. Where the Web contains static HTML pages, Web 2.0 is dynamic, in that it serves applications to users and offers open communications with an emphasis on Web-based communities. Web 2.0, because it focuses on people and communications, encompasses a large number of technologies and standards.

Where Web 2.0 is focused on people, the Semantic Web is focused on machines. The Web requires a human operator, using computer systems to perform the tasks required to find, search and aggregate its information. Its impossible for a computer to do these tasks without human guidance because Web pages are specifically designed for human readers. The Semantic Web is a project that aims to change that by presenting Web page data in such a way that it is understood by computers, enabling machines to do the searching, aggregating and combining of the Web's information — without a human operator.

According to the World Wide Web Consortium (W3C), the Web can reach its full potential only if it becomes a place where data can be shared, processed, and understood by automated tools as well as by people. The Semantic Web is an extension of the current World Wide Web, not a separate set of new and distinct websites. It builds on the current World Wide Web constructs and topology, but adds further capabilities by defining machine-processable

2.2. Crowd sourcing on the Web

data and relationship standards along with richer semantic associations.

With it, the use of semantic technologies makes it possible to describe the logical nature and context of the information being exchanged, while allowing for maximum independence among communicating parties. The results are greater transparency and more dynamic communication among information domains irrespective of business logic, processes, and workflows [19]. With the evolution of such, it has been possible to create systems on which the information in it could be achieved via collaboration, instead of the standard way of uploading information by any user, and assuming the truthfulness of every detail being introduced by the users of an application.

And so, Crowd sourcing emerged, with it it was possible to obtain needed services, ideas, or content by soliciting contributions from a large group of people, and especially from an on line community.

2.2 CROWD SOURCING ON THE WEB

One of the definitions of crowd sourcing has been as follows: *"Crowd sourcing is a type of participative on line 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 that which the user has brought to the venture, whose form will depend on the type of activity undertaken."* [9]

In a February 1, 2008 article, Daren C. Brabham, "the first person to publish scholarly research using the word crowdsourcing" and writer of the 2013 book, *Crowdsourcing*, defined it as an "online, distributed problem-solving and production model".

Despite the multiple definitions of crowdsourcing, one constant has been the broadcasting of problems to the public, and an open call for contributions to solving the problem.

A recent example of the application of these in the emerging web technologies of crowdsourcing would be NASA's photo organizing project which asks internet users to browse photos taken from space and try to identify the location the picture is documenting.

2.2. Crowd sourcing on the Web

Other companies however, managed to achieve a social network through image tagging, on which the users collaborate into identifying and tagging people on pictures. The people tagged in though, could either be associated with a profile or just be associated with normal text. However though, the pictures depicted in these projects are not being deployed by the normal users, and the information that they are voting for is not dynamic, the users can only give feedback based on the location of the pictures, they cannot however identify other entities that could be identified, and neither such associate them with new properties. It would be even harder, to associate an entity with the same semantic meaning.

2.3. Online Social Networks and Interactive Tagging

2.3 ONLINE SOCIAL NETWORKS AND INTERACTIVE TAGGING

In the most recent years websites such as *Facebook* became one of the most widely used photograph tagging social media network.

Tagging, which is one of the defining characteristics of Web 2.0 services, allows users to collectively classify and find information, also known as collaborative tagging or folksonomy.

Through their website when you tag someone, you create a link to their profile. The post you tag the person in may also be added to that person's Timeline. For example, you can tag a photo to show who's in the photo or post a status update and say who you're with.

This is essential for the identification of people through collaboration, you can allow other users to tag your own photographs associate them with different users that are registered on the website.



Figure 1.: Tagging of a Person on a social media website.

However most of these websites have a limitation, specifically when it comes to who they are tagging. The application will only allow a restricted tagging to a resource that is already on their website, more specifically another user's profile.

Furthermore there is no way to disagree or to prevent other users from tagging wrongly.

2.4. Interactive Tagging in Cooperative Systems

2.4 INTERACTIVE TAGGING IN COOPERATIVE SYSTEMS

There are broad folksonomies [15] which have many users contributing to the creation of tags and narrow folksonomies where only a few users are tagging particular items. A broad folksonomy [16] allows many people to tag the same resources and any user can tag a resource using their own vocabulary. In a narrow folksonomy [16], only a few people are able to create tags and these tags are used by other users to locate resources. Unlike broad folksonomies, narrow folksonomies are not very common. An example of a broad folksonomy is del.icio.us [17], this is a website where users can tag any online resource they find relevant with their own personal tags. An example of how a collaborative tagging system would be demonstrated in Figure 2.

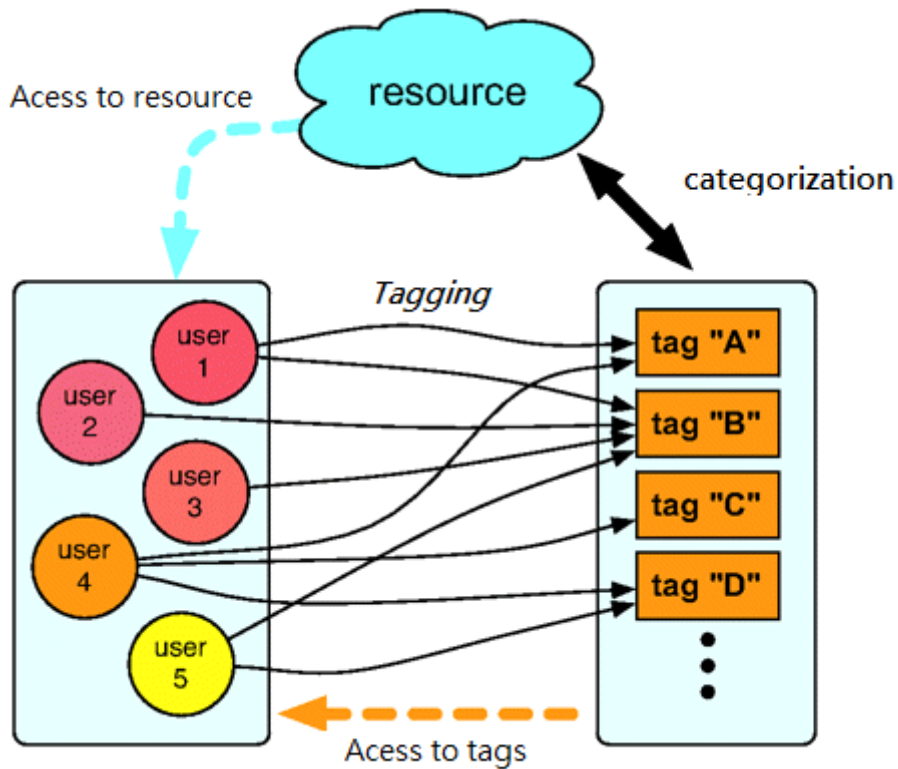


Figure 2.: Schema of a Cooperative tagging system.

Figure 2 is a block diagram that depicts a collaborative tagging process: web users access a web resource and freely associate tags with it. Their interaction with the system also allows them to access tags previously entered by themselves and by other users. The aggregated activity of users leads to an emergent categorization of resources in terms of tags shared by a community.

With the development of such an architecture, some issues arise. First, when people make

2.5. Identification of Entities

mistakes while tagging, such as spelling mistakes, or accidental tagging with the wrong tag. Second, synonyms or different languages can only be handled by explicit tagging. Third, there is no current solution to handle homonymy, i.e. to distinguish different meanings of a word.

The proposed solution will address these matters with the help of a system that can allow a community to collaborate and vote on addressing wrong misspelled tags, explicit tagging or different meanings of a word with a system of voting.

2.5 IDENTIFICATION OF ENTITIES

In the proposed system, every user is able to identify any entity in a photograph. Every entity can either be a Person, a monument or perhaps an animal. There are a large number of possible things that can be identified, and each of which can have several different properties, this is usually defined as an Ontology.

The system can support a structure that provides the possibility to integrate it with an ontology. With this any user is able to access several possible entity types and use their properties to define values of what they could be attempting to identify.

In Figure 3 we have an example of a branch of an ontology. It is used to describe the relationship between several entities, in this case a Fruit, however depending on the level of detail, the user could potentially go further in the graph of the fruit subclass e's and attempt to identify something that could either be an Apple or a Pear and so on.

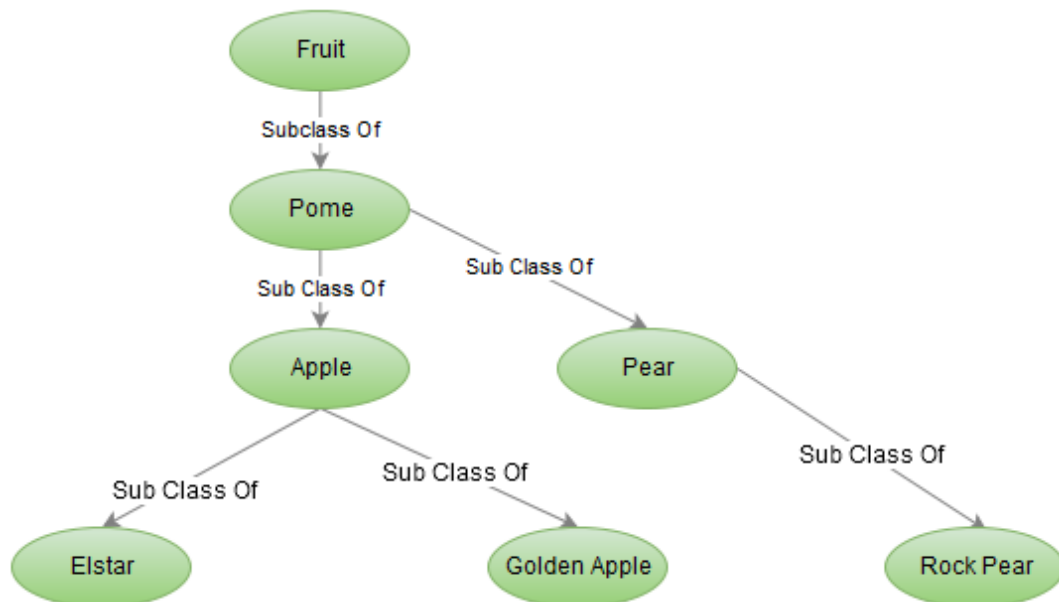


Figure 3.: A fragment of a fruit ontology.

2.5. Identification of Entities

The system will support the possibility to integrate fully or partially an ontology allowing an administrator user to edit the existing types of entities and their properties based on the ontology that has been applied. With this the system should be generic enough to allow a vast variety of possible ontologies, which could be adapted based on the context of what is intended to be identified in the system.

2.6. Ontologies In Cooperative Systems

2.6 ONTOLOGIES IN COOPERATIVE SYSTEMS

"Ontologies are often equated with taxonomic hierarchies of classes, class definitions, and the subsumption relation, but ontologies need not be limited to these forms. Ontologies are also not limited to conservative definitions — that is, definitions in the traditional logic sense that only introduce terminology and do not add any knowledge about the world. To specify a conceptualization, one needs to state axioms that do constrain the possible interpretations for the defined terms." [23]

In the semantic web, an Ontology is a description of the concepts and relationships that can exist for an agent or a community of agents. This definition is consistent with the usage of ontology as set-of-concept-definitions, but more general. [20]

Ontologies are used to classify the terms that can be used in a particular application, characterize possible relationships, and define possible constraints on using those terms. In practice, Ontologies can be very complex (with several thousands of terms) or very simple (describing one or two concepts only).

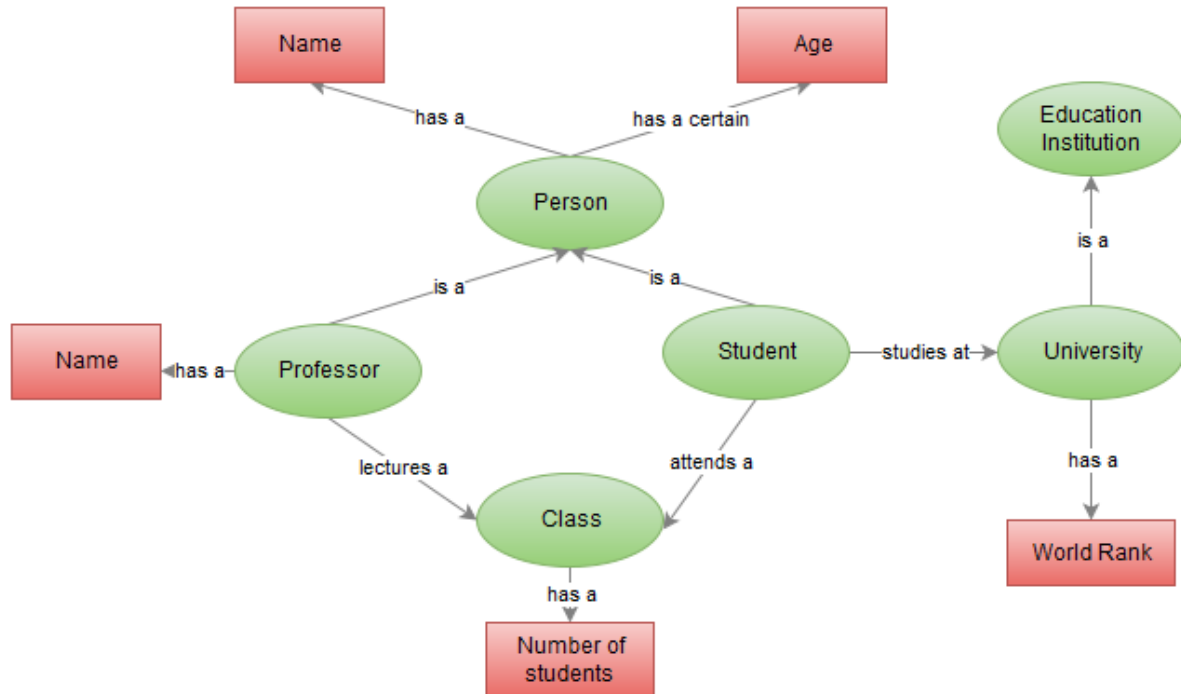


Figure 4.: Example of an Academic Ontology. [22]

2.6. Ontologies In Cooperative Systems

The example above demonstrates a simplified part of an Ontology that represents an Animal, a Person and the relationship between them, as well as the properties and sub properties of each other.

For the purpose of notion the RDF sentence is used in this context as the basic unit of summarization.

Classes in an ontology can be used to represent entities in photographs, they can be more in depth in detail or less depending on the grade of certainty. For instance on the image below a user can identify the tag by indicating that the entity is in fact an animal, however since an ontology has different other classes which are subclasses of animal, such as Domestic Animal or simply Horse, a user can go more in detail and use that type of entity instead. Taking the opportunity to extract properties of that instead.

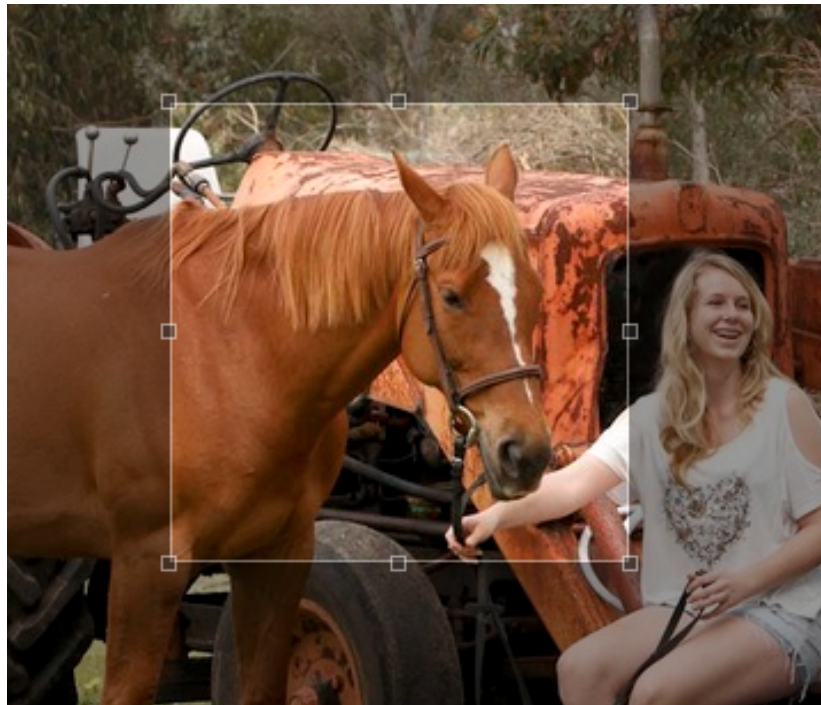


Figure 5.: Example of tagging of an Animal.

In this case, the entity Person is a subclass of the entity Animal. Man and Woman are subclasses of the entity Person and Animal, these are represented through a relationship of `rdf:SubclassOf` that is used to state that all the instances of one class are instances of another.

In the context of Cooperative Systems ontologies can provide a way to allow users to tag anything they wish, to identify without having to input using their own words, this is unless it is the value of a property from an entity they tagged.

2.7. Existing Solutions

2.7 EXISTING SOLUTIONS

Europeana 1914-1918 [10] is a collaboration led by Europeana with support from the The Great War Archive [11] team at the University of Oxford. A website is used by the project to encourage the public from all European Union countries to contribute information about the First World War, especially their family's stories and digitized photographs of their artefact's. In addition to the website, since 2011 First World War stories have been collected directly from family members during roadshow events held in Germany, Luxembourg, Ireland, Slovenia, Denmark, with further events planned in Cyprus, Belgium, Italy and more. The items collected by the project are released on the Internet for use under a Creative Commons license.

Google Image Labeler is a sort of game where users are asked to label pictures in order to improve images search results [12].

The Vancouver Police Department (VPD) has put up a website entitled Hockey Riot 2011 [13], informing people about the VPD's investigations into the 2011 Stanley Cup Riot. It also asks people to contribute any pictures or video that they may have taken during the riot, with the goal of identifying people who may have participated in the rioting. As of July 1, 2011, 101 arrests have been made as a result of this project.

The Gateway to Astronaut Photography of Earth - Image Detective" is an interactive citizen science hunt for the Earth location of images taken from space by astronauts since the 1960s. Reviewing 1.8 million photos, individuals submit what they believe to be the location of a given photo, and thus accumulate "points" and "badges" on part of the NASA website. [14]

In 2010 the Library of Congress received, as a donation, the Liljenquist family's collection of photographs of the American Civil War. With most of the people and photographers unidentified, the Library posted the images on Flickr and sought the public's help in providing information: "Please let us know if you recognize a face from your family, a regiment, or a photographer's painted studio backdrop!" [15]

Metadata Games is a free and open source gaming platform for crowd sourcing keyword metadata for photo, audio, and moving image artifacts. The platform motivates players to visit archives and explore humanities content while contributing to vital records.

Sightmap is a sightseeing popularity heat map overlaid on Google Maps, based on crowd sourcing: the number of Panoramic photos at each place in the world.

2.8. Summary

2.8 SUMMARY

The different projects stated in the previous chapter are a result of a need for information that can't be obtained without the collaboration of several different users. Most of them are very specific regarding the needs of what needs to be identified such as the Europeana project led by the University of Oxford, in which they attempted to gather stories of different photographs.

Other ones such as the Library of Congress and the Vancouver Police Department attempted at identifying people in photographs through collaboration. However all of these projects had a very specific goal and were limited to what needed to be identified, limited to what the users could input through their applications.

Most of these attempts however relied on a very detailed human inspection of what every other user contributed on their project, either that or what needed to be specified was limited to what they were looking for, such as with the project The Gateway to Astronaut Photography of Earth in which individuals could only submit what they suspected to be the location of the photos that were provided by NASA.

Not only this had to be determined based on what the users had to say, there was also no real measure of how certain they were regarding their input, neither could they vote if any other user information was correct or incorrect.

These projects were also never generic in a way that allowed their users to identify other entities, entities of which could be useful to extract additional information from the photograph.

This lack of standardized data prevents the use of an automated mechanism that analyzes the user's input, such as that it ends up with having most projects relying on a human being to interpret most of the information that has been submitted by the other users.

The use of an ontology could potentially fill this gap and allow a more controlled system which can provide a better understanding of what is being identified in the photograph.

FOTOBORA, PROPOSED SOLUTION

Previously stated in chapter 2, the existing Crowd sourcing projects allow different users to identify specific entities depending on the goal in mind; they however do use any generic identification and are not capable of allowing users to add additional relevant information that can be interpreted without the aid of another human being.

To overcome these issues, we propose a solution by building a system which enables users to publish photographs and allow a community to aid in the identification of the entities proposed in the photo through collaboration. The system has been called Fotobora which stands for a conjunction of the Portuguese words Fotografia (Photograph) and Colabora (Collaborate).

In this chapter the proposed solution will be described, depicting some aspects such as its architecture and data model. Also the requirements it must fulfill, the methodology that will be discussed and its features.

3.1 REQUIREMENTS

The system proposed is designed to allow users to publish their own photographs as well as to allow them to contribute to the identification of entities in other photographs published by other users. The system shall provide the following features:

- **Upload photographs:** the system shall allow any user that is authenticated to upload any photograph. This photograph will contain a name and a date.
- **Tag entities in the photograph:** users should be able to identify an entity in any photograph through the aid of a tagging tool. The tag should also be complemented with a type of entity and possible initial values for the properties of that entity type.
- **Support ontologies for the types of entities and their properties :** each entity on the system has one or more types (for example: Person and Statue) and each of those

3.1. Requirements

types has properties (for example: name, color). As such the system has the required structures to contain this information, which are compatible with the concept of an ontology and therefore this information can be transposed from most ontology formats.

- **Allow Multiple possible entity types for each entity identified:** since different tags could have debatable identifications, different users should be able to contribute with more types of entities.
- **Vote on the type of entity identified:** each type of entity identified in a photograph should be subject to a level of agreement from the community of users that are contributing.
- **Vote on an entity's properties:** users might agree or disagree with the values identified on each property of an entity. For this reason the system should allow them to express their opinion with a specific level of certainty.
- **Add new values to an entity's properties:** any identification could have some level of doubt, as it could diverse from something a user is completely sure of to something he isn't sure at all but his information might prove useful. Because of this the system should allow a certainty grading scale to each value of a property identified.
- **Add stories to a photograph and/or their entities:** every photograph or entity identified could have a background story related to give further context and an incite discussion among the users.
- **Export the information identified in a photograph:** the system should be able to export all the existing information in the context of a photograph in PDF and XHTML format.
- **Import annotations:** the system should be able to import annotations of a photograph through a XML Schema.
- **Support Genealogical Trees:** the system will also support the association of entities of the type Person to be related with eachother.

The representation of what a user and an administrator can do in the system are expressed in Figure 6 in the form of a Use case diagram.

3.1. Requirements

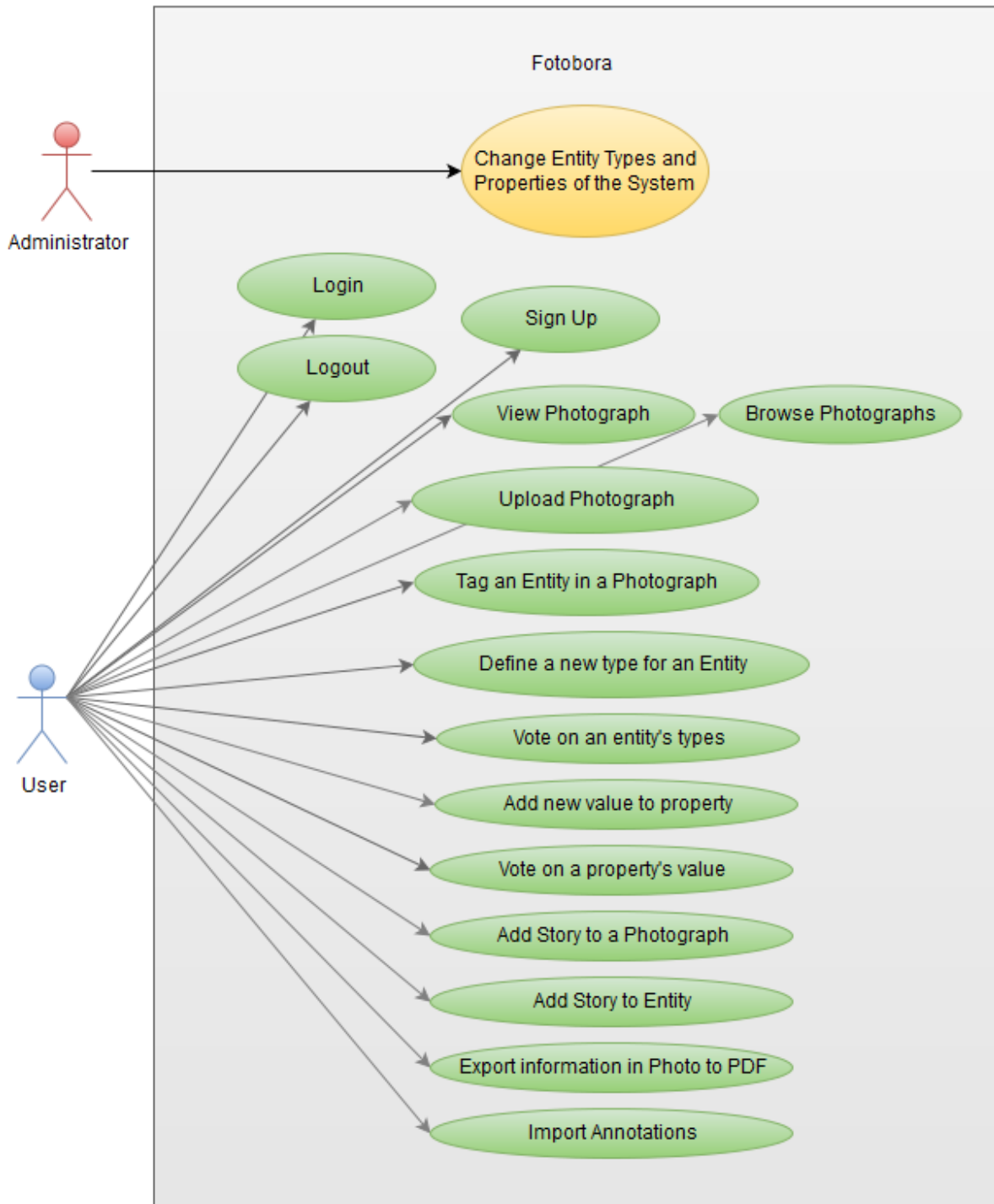


Figure 6.: Fotobora Use Case Model.

3.2. Proposed Architecture

3.2 PROPOSED ARCHITECTURE

From the analysis done, it was possible to identify several important elements of the system to be implemented. In the block diagram of Figure 7 the main components of the system are identified, along with the way they interact with each other. Externally the Computer Client interacts with the system.

Besides the basic components such as the web server, dynamic web pages generator, data base server and engine that the system should have, the system should also contemplate the following components:

- **Photograph Manager:** Manages each photograph in the system, their stories and provides a search engine for their accessibility.
- **Entity Manager:** Manages each entity in the system, their relationship towards each photo, the property types associated as well as the values of the properties in each one of them.
- **User Manager:** Manages each user in the system, their relationship between each photograph and their contribution towards other photographs.
- **Entity Type Manager:** Manages the existing property types in the system as well as their properties. Also manages the hierarchy and the relationships between each property type.
- **XML Processor:** Processes an uploaded XML file containing annotations that describe a photograph; these should be following a specific schema.
- **Document Generator:** Generates a document in PDF format based on the contextual user input information of a photograph.

3.3. Data Model

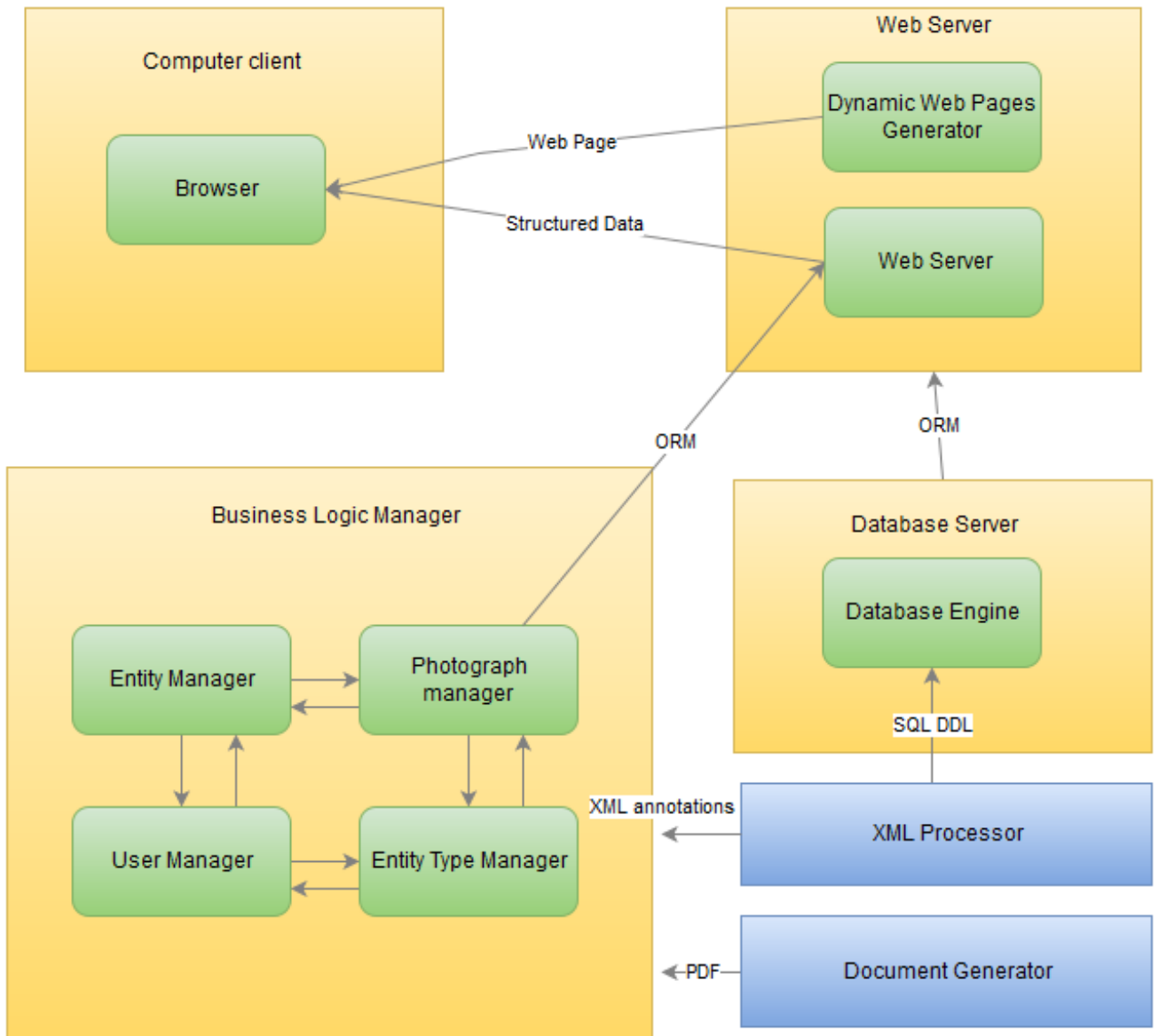


Figure 7.: Fotobora Architecture (block diagram)

3.3 DATA MODEL

The proposed data model is represented in figure 8.

Every user has an account in the system and has the possibility to upload/create photos an infinite number of times.

The Photos created in the system have a date of submission and a name associated with only a relationship to the user that created it, unless it has been submitted via the XML Importer with already some entity relationships.

3.3. Data Model

Upon submission of a photo any user has the possibility to tag an entity in it via a flexible rectangular selector through the interface. It will delimit the section of the photo that the user wishes to identify, allowing it to identify with an initial contextual information.

This information has a mandatory field which is its entity type, which comes with several associated properties. These properties have an associated first possible value as well as its level of certainty, however these are not mandatory if the user does not wish to complete them at this stage.

These entity types, their properties and relationships between them are managed via the administrator of the server. They follow the structure of components of an ontology, this is explained in further detail in the next section.

Furthermore the system allows the creation of a story for every photograph and comments associated for that story; as well as a story and a commentary session for each entity identified.

In addition, the system supports a genealogical tree, allowing the association of entities of the type Person to be related with each other.

3.3. Data Model

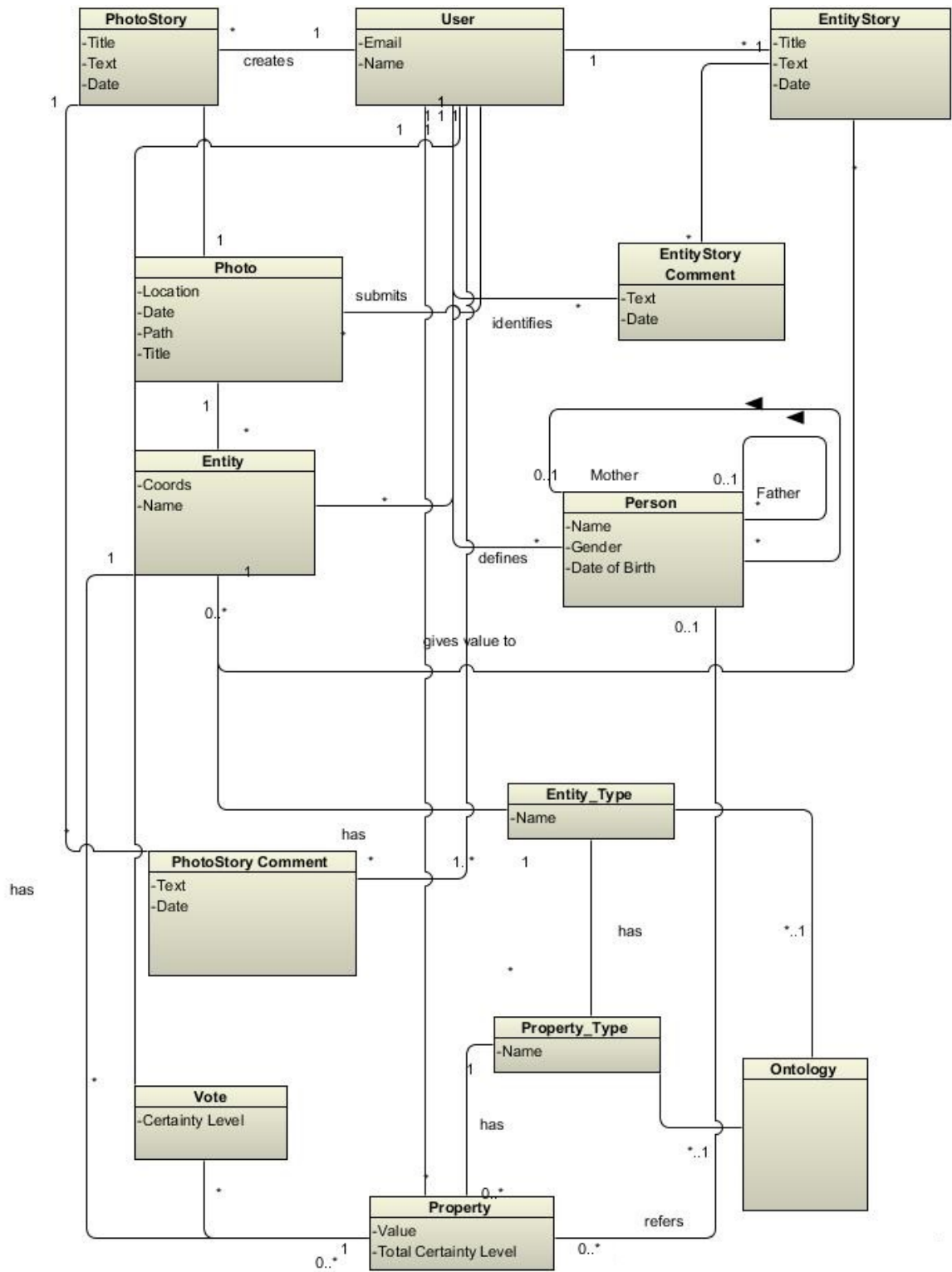


Figure 8.: Data Model.

3.4. Ontologies Analysis

3.4 ONTOLOGIES ANALYSIS

In this section the selected Ontologies will be analyzed to determine what information can be obtained from them and how that information can be used and integrated with Fotobora's ontology like structure.

3.4.1 *DBPedia*

The website DBPedia provides a fast sum of open source ontologies regarding many different subjects. It allows users to semantically query relationships and properties associated with Wikipedia resources, including links to other related datasets [29].


There are many fitting ontologies for the context of general identification; let us consider the examples in fig 9 to 11.

Property	Value
<code>rdf:type</code>	▪ <code>owl:Class</code>
<code>rdfs:isDefinedBy</code>	▪ http://dbpedia.org/ontology/
<code>rdfs:label</code>	▪ <code>animal</code>
<code>rdfs:subClassOf</code>	▪ <code>dbo:Eukaryote</code>
<code>owl:equivalentClass</code>	▪ <code>wikidata:Q729</code>
<code>wdrs:describedby</code>	▪ <code>dbo:data/definitions.ttl</code>
<code>http://www.w3.org/ns/prov#wasDerivedFrom</code>	▪ http://mappings.dbpedia.org/index.php/OntologyClass:Animal
<code>is http://open.vocab.org/terms/defines of</code>	▪ http://dbpedia.org/ontology/
<code>is http://open.vocab.org/terms/describes of</code>	▪ <code>dbo:data/definitions.ttl</code>
<code>is rdfs:domain of</code>	▪ <code>dbo:breeder</code> ▪ <code>dbo:dam</code> ▪ <code>dbo:damsire</code> ▪ <code>dbo:foalDate</code> ▪ <code>dbo:grandsire</code> ▪ <code>dbo:sire</code>
<code>is rdfs:range of</code>	▪ <code>dbo:animal</code> ▪ <code>dbo:dam</code> ▪ <code>dbo:damsire</code> ▪ <code>dbo:grandsire</code> ▪ <code>dbo:sire</code>
<code>is rdfs:subClassOf of</code>	▪ <code>dbo:Amphibian</code> ▪ <code>dbo:Arachnid</code> ▪ <code>dbo:Bird</code> ▪ <code>dbo:Crustacean</code> ▪ <code>dbo:Fish</code> ▪ <code>dbo:Insect</code> ▪ <code>dbo:Mammal</code> ▪ <code>dbo:Mollusca</code> ▪ <code>dbo:Reptile</code>

Figure 9.: Example of a branch of Animal in the DBpedia Ontology.

3.4. Ontologies Analysis

About: [person](#)
 An Entity of Type : [Class](#), from Named Graph : <http://dbpedia.org/resource/classes#>, within Data Space : [dbpedia.org](#)



Property	Value
rdfs:type	<ul style="list-style-type: none"> owl:Class
rdfs:isDefinedBy	<ul style="list-style-type: none"> http://dbpedia.org/ontology/
rdfs:label	<ul style="list-style-type: none"> person
rdfs:subClassOf	<ul style="list-style-type: none"> dbo:Agent
owl:equivalentClass	<ul style="list-style-type: none"> foaf:Person dul:NaturalPerson wikidata:Q215627 wikidata:Q5 http://schema.org/Person
wdrs:describedby	<ul style="list-style-type: none"> dbo:data/definitions.ttl
http://www.w3.org/ns/prov#wasDerivedFrom	<ul style="list-style-type: none"> http://mappings.dbpedia.org/index.php/OntologyClass:Person
is http://open.vocab.org/terms/defines of	<ul style="list-style-type: none"> http://dbpedia.org/ontology/
is http://open.vocab.org/terms/describes of	<ul style="list-style-type: none"> dbo:data/definitions.ttl
is rdfs:domain of	<ul style="list-style-type: none"> dbo:Person/height dbo:Person/weight dbo:abbeychurchBlessing dbo:abbeychurchBlessingCharge dbo:achievement dbo:activeYear dbo:activeYears dbo:activeYearsEndDateMgr dbo:activeYearsEndDateMgr dbo:activeYearsStartDateMgr dbo:activeYearsStartDateMgr dbo:activity dbo:affair dbo:agency dbo:allegiance

Figure 10.: Example of a branch of Person in the DBpedia Ontology.

About: [place](#)
 An Entity of Type : [Class](#), from Named Graph : <http://dbpedia.org/resource/classes#>, within Data Space : [dbpedia.org](#)



Immobile things or locations.

Property	Value
rdfs:type	<ul style="list-style-type: none"> owl:Class
rdfs:comment	<ul style="list-style-type: none"> Immobile things or locations.
rdfs:isDefinedBy	<ul style="list-style-type: none"> http://dbpedia.org/ontology/
rdfs:label	<ul style="list-style-type: none"> place
rdfs:subClassOf	<ul style="list-style-type: none"> owl:Thing
owl:equivalentClass	<ul style="list-style-type: none"> dbo:Location http://schema.org/Place
wdrs:describedby	<ul style="list-style-type: none"> dbo:data/definitions.ttl
http://www.w3.org/ns/prov#wasDerivedFrom	<ul style="list-style-type: none"> http://mappings.dbpedia.org/index.php/OntologyClass:Place
is http://open.vocab.org/terms/defines of	<ul style="list-style-type: none"> http://dbpedia.org/ontology/
is http://open.vocab.org/terms/describes of	<ul style="list-style-type: none"> dbo:data/definitions.ttl
is rdfs:domain of	<ul style="list-style-type: none"> dbo:altitude dbo:annualTemperature dbo:areaCode dbo:areaDate dbo:areaLand dbo:areaOfCatchmentQuote dbo:areaQuote dbo:areaRank dbo:areaTotal dbo:areaWater dbo:artificialSnowArea dbo:authorityMandate dbo:averageDepth dbo:averageDepthQuote

Figure 11.: Example of a branch of Place in the DBpedia Ontology.

3.4. Ontologies Analysis

In the provided system we we took three different branches of an ontology, Animal, Person and Place. These three entities should be commonly used to identify most tags in photographs, particularly when it comes to a context of general human taken photographs.

For this particular system the ontology is provided with several definitions, all of which are being constructed through an RDF annotation which is used to provide the necessary metadata to describe the ontology.

The components of relevance used are described as follows:

- **rdf:Property**: the class of properties.
- **rdf:type**: an instance of **rdf:Property** used to state that a resource is an instance of a class.
- **rdfs:isDefinedBy**: the definition of the subject resource.
- **rdfs:label**: a human-readable name for the subject.
- **rdfs:subClassOf** the subject is a subclass of a class.
- **rdfs:equivalentClass** the equivalent class.
- **rdfs:domain** a domain of the subject property.
- **rdfs:range** the definition of the subject resource.
- **is rdfs:subclassOf** the classes are subclasses of the subject.

From what we can analyze here the ontology provides with a vast range of components that can be useful in the Fotobora system, namely the relationship between each Class.

3.4.2 *Protégé Ontologies*

The Protégé library provides a vast list of available ontologies overall using the OWL (Web Ontology Language) markup that describe several specific subjects, such as a Wine, Amino-acid or even Camera Part's ontologies.

OWL adds semantics to the schema. It allows you to specify far more about the properties and classes. It is also expressed in triples. For example, it can indicate that "If A isRelatedTo B" then this implies "B isRelatedTo A". Or that if "C isAncestorOf D" and "D isAncestorOf E" then "C isAncestorOf B".

3.4. Ontologies Analysis

Another useful thing owl adds is the ability to say two things are the same, this is very helpful for joining up data expressed in different schemas. You can say that relationship "sired" in one schema is owl:sameAs "fathered" in some other schema. It can also be useful to say two things are the same in the same system.

As an example of an ontology that describes a simple family relationship we took one that Mother is exactly a Person that is not Father; isParentOf is a transitive property; isParentOf is inverse of hasParent. The example expressed in the owl syntax is as follows:

```
@prefix :      <http://example.com/family.owl#> .
@prefix rdf:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs:  <http://www.w3.org/2000/01/rdf-schema#> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .

:Person rdfs:subClassOf
  [ a owl:Restriction ;
    owl:onProperty :hasParent ; ] ;

:Mother owl:equivalentClass
  [ owl:intersectionOf
    ( [owl:complementOf :Father]
      :Person ) ] .

:isParentOf
  a owl:TransitiveProperty , owl:ObjectProperty ;
  owl:inverseOf :hasParent .
```

3.4. Ontologies Analysis

3.4.3 *Summary*

Despite the OWL Language providing further helpful information rather than with the RDF annotation provided by DBPedia the OWL Language is not represented like a database system; it poses no requirement that the only properties of an individual are those mentioned in a class it belongs to; there is also no assumption that everything is known and classes and properties can have multiple definitions.

RDF also defines some extra structure to triples. The most important thing RDF defines is a predicate called "rdf:type". This is used to say that things are of certain types.

RDFS (RDF Schema) defines some classes which represent the concept of subjects, objects and predicates. This means we can start making statements about classes of thing, and types of relationship. It also allows to describe in human readable text the meaning of a relationship or a class. This is a schema. It tells the legal uses of various classes and relationships. It is also used to indicate that a class or property is a sub-type of a more general type. For example "HumanParent" is a subclass of "Person". "Loves" is a sub-class of "Knows".

RDF can be exported in a number of file formats. The most common is RDF+XML but this has some weaknesses.

Its important to know that RDF is a way of working with triples, but not the file formats.

Mainly for this reason the RDF ontologies provided by DBPedia seem to offer an easier integration towards what can be represented in the Fotobora system.

FOTOBORA IMPLEMENTATION

This chapter aims to clarify the implementation of the proposed solution of Fotobora, by describing the technology, framework and methodologies behind its implementation. Furthermore it also exposes the system's key features as well as the potential future development based on its implementation.

4.1 APPLICATION FRAMEWORK

The proposed application, named as "Fotobora" consists on a web-based application that allows anyone to upload photographs and collaborate in the identification of any possible entity exposed by any user.

This application has been built under the "Ruby on Rails" web application framework.

Though other languages or tools can be used successfully to achieve these results, Ruby-on-Rails(RoR) on other hand, has an additional benefit: it allows a faster and smoother development. And the reason is that Rails is suitable for both state-of-the-art public websites that compete with Single Page JavaScript applications, and complex core system applications that usually look a bit 'uglier' (with a more generic, lower fidelity UI), but compensate this blemish with much more complicated business rules and logic.

Its benefit is that it is versatile and able to compete with both the sleek and the powerful. For most of common problems, Rails has a component at everyone's disposal almost right out-of-the-box with documentation that is consistently above average, leading to well written, concise and time-saving documents.

This allows to spend the development efforts on things that really matter, letting the construction of the whole platform to be performed without concerns with coding the basic functionalities.

4.1. Application Framework

"Ruby on Rails" [27] is a web application framework written in the "Ruby" [28] programming language. It provides several key features:

- **Database Access Library** also labeled as "DAL" is a object-relational mapping (ORM) that makes it easier to manipulate information that is stored in its relational database. It creates an "ActiveRecord" which maps database tables to object-oriented classes and table rows to objects.
- **MVC architecture** It is based on the model-view-controller paradigm, which is depicted in the figure 12. The central component of MVC, the model, captures the behavior of the application in terms of its problem domain, independent of the user interface.[25] The model directly manages the data, logic and rules of the application. A view can be any output representation of information, such as a chart or a diagram; multiple views of the same information are possible, such as a bar chart for management and a tabular view for accountants. The third part, the controller, accepts input and converts it to commands for the model or view. [26]

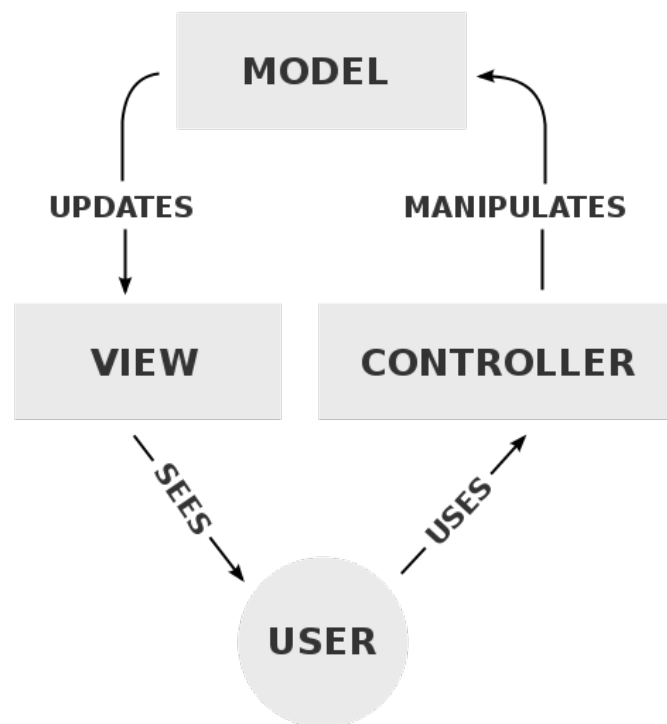


Figure 12.: MVC Process

- **Default Libraries** It includes by default several useful and core libraries that perform basic web related tasks.

4.2. Social Collaborative System

- **Guiding Principles** It is guided by two principles that improve productivity: “Do not Repeat Yourself” (DRY) aims the decrease repetition of information of all kinds, like for example, the schema of a database, or recurring code; “Convention over Configuration” (CoC), which entails that a developer only has to specify and configure unconventional aspects of his application, otherwise convention always is applied.
- **Extensibility** The framework nurtures an environment of extensibility, and a very large amount of additional libraries are available to support any type of application.

Several additional libraries were used, besides the default set included by Ruby on Rails. These enabled to implement the required features in Fotobora. They are as follows:

- “Devise”, for authentication management;
- “Nokogiri”, for XML parsing and manipulation;
- “Paperclip”, for file management;
- “Ruby-graphviz ”, provides an interface for the construction of genealogical trees;
- “Wkhtmltopdf ”, open source command line tools to render HTML into PDF and various image formats using the Qt WebKit rendering engine.
- “Pdkit ”, creates PDFs using plain HTML and CSS. Uses wkhtmltopdf on the back-end which renders HTML using Webkit.
- “sass-rails”, SCSS for stylesheets;
- “uglifyer”, compressor for JavaScript assets;

4.2 SOCIAL COLLABORATIVE SYSTEM

In order for a community to be able to collaborate with each other in the identification of entities in photographs there must be a system that allows some sort of social interaction.

For this reason, we implemented an authentication system that allows users to be able to log and sign in the system. This is intended for the purpose of being able to associate photographs with user created accounts and also to keep a record of their contributions in the system.

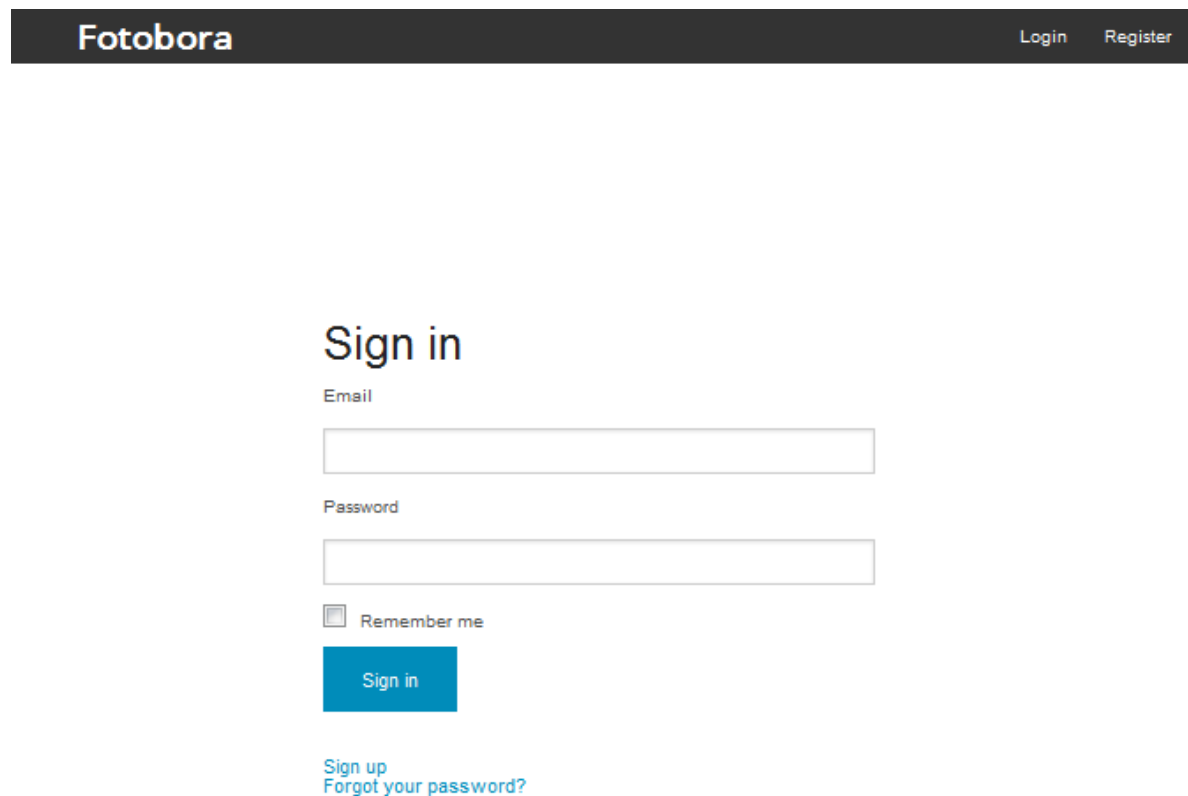
There was also the possibility of making it anonymous for the user submitted photographs and also for an anonymous contribution, but this could result in an uncontrolled system where it could be victim to unreasonable contributions from certain users, making it difficult to

4.2. Social Collaborative System

supervise if it did eventually happen.

Furthermore in the context of most photographs, these are usually private, or family oriented, and most users would prefer to allow a certain level of privacy instead of exposing it to every user publicly.

Figure 13 shows the authentication system, where the user is prompted if no session is currently active for him.



Fotobora Login Register

Sign in

Email

Password

Remember me

Sign in

[Sign up](#)
[Forgot your password?](#)

Figure 13.: Sign In Page

With a successful authentication the user is directed to the homepage (let's say, Fotobora main menu) where it is possible to visualize the photos published by the user as well as the photos published by the other users.

In Figure 14 we can see Fotobora main menu.


4.2. Social Collaborative System

Fotobora [armada@fotobora.com](#) [Edit registration](#) [Logout](#)

armada@fotobora.com
Registered on: 22 Jan 01:00


[Import Family Tree](#) [Upload Picture](#)

My Photos



[Wedding Braga](#)
2016-01-22 00:00:00 UTC
0 entities
0 stories

All Other Photos



[One Street](#)
2014-02-01 00:00:00 UTC
4 entities
1 stories

Figure 14.: Sign In Page

The system provides a search engine, that will search any entity or user based on the keyword that has been submitted, returning the photographs that contain it. Figure 15 illustrates the results of a query using the keyword "Wedding".

4.3. Identification of Entities in Photographs

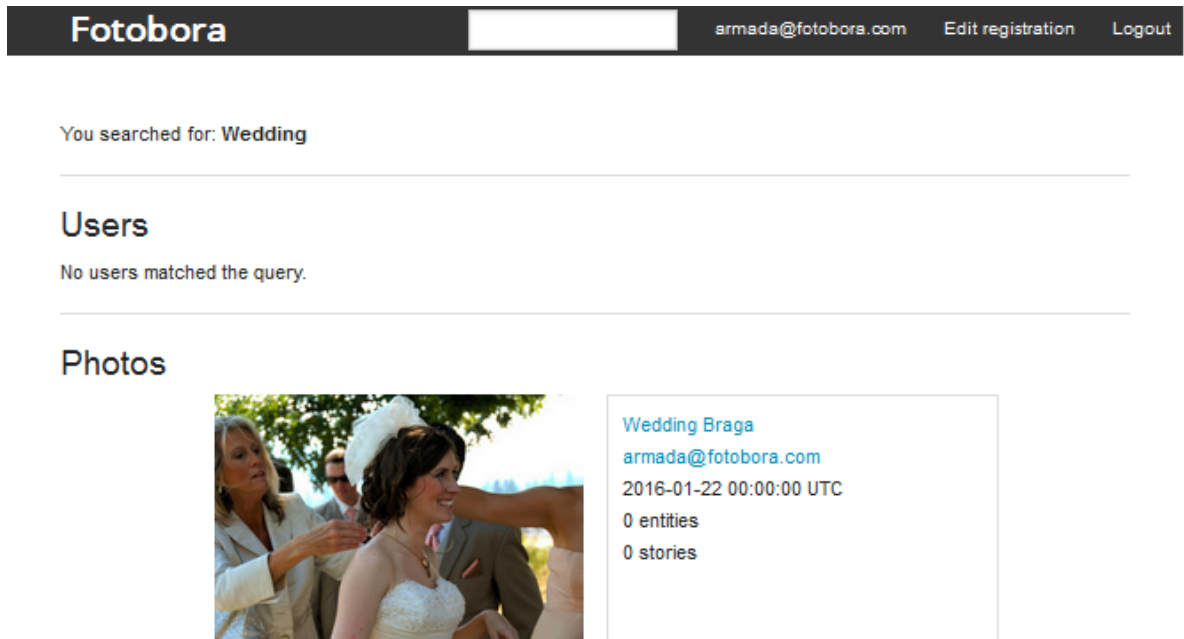


Figure 15.: Search Results

4.3 IDENTIFICATION OF ENTITIES IN PHOTOGRAPHS

To identify any entity in a photograph several different methods have to be applied in order to satisfy the needs of the users. The methods that were used are described in this section.

4.3.1 *Tagging of Entities*

A tag is a non-hierarchical keyword or term assigned to a piece of information, in the context of this system a Photograph. Tags are generally chosen informally and personally by the item's creator or by its viewer.

In a user point of view the process of tagging should be extremely simple, allowing them to be able to simply select the delimited area they wish to be identified.

This could be done in several ways, either by delimiting through a selection of a certain area of the photograph and then dragging, creating a rectangular shape as it moves around the first initial area; or through a circular area using the same algorithm; or simply by clicking on a certain area and automatically creating a standardized sized tag.

For an easier user experience we decided to approach with a rectangular tag implementation; while most entities do not have a rectangular shape, this could potentially reduce the ambiguity of what is being attempted to be identified rather than if it had a different shape or regular size.

4.3. Identification of Entities in Photographs

This is obviously dependent on the user's management of the tool itself in order to clarify what its meant to be identified. Below in figure 16 we can visualize its basic implementation.



New Entity



Select an area where the entity is in the picture above, then select a type and optionally fill out the properties.

Figure 16.: Tagging

With its first tagging the user is prompted to identify with some initial values, which would then serve as label on mouse over action based on the best results of the pool. For instance if the majority agreed that the entity that has been identified is a person and the person's name is Mark, then the label will state "(Person) Mark".

4.3. Identification of Entities in Photographs

4.3.2 *Identification*

These entities, are not restricted to people. Also other objects, such as monuments, animals, artifact, etc can be tagged and associated with contextual information, such as the object's owner, or the object's characteristics.

With this in mind the concept of Entity Type, Property and Property Value were implemented in a way that it would allow a system to identify anything based the the values of these three concepts.

The issue here is when this information is submitted by a large number of different users which results in information that is very difficulty parsed such as:

- Mislabeled or misspelled Entity types or their properties
- Duplicate entries for entity types or any of their properties
- Invalid types or properties
- Different Languages
- Extremely difficult parsing of information

This issue is resolved if the users have no involvement in the submission of this information through their end. With the injection of Entity Types and Property types in the system through an administrator can make all of this information available to the community without risking any of the issues labelled previously. This would also filter entity types and properties that have been previously added over on the same photograph preventing potential repetition on the same entity.

In order for the administrator to be able to import this kind of information a source must be used. Such a source is usually only available through ontologies since their definition covers very closely the concept of identification.

As of so users are able to select through a dropdown list the entity types and be able to visualize the following properties of such, in a dynamic matter that prevents further page loads everytime a new entity type is selected. This is addressed by the implementation of a preload of all entity types and their properties on the first page load. This also verifies the existing entity types that have already been assigned to the entity; filtering them from the list.

This ontology was initially static (i.e., predefined and closed); however it is desirable to allow the use of an open ontology that will evolve the time with the help of all the interventionists. This is further explained in section 4.5.

4.4. Ontology and Entity Types

4.4 ONTOLOGY AND ENTITY TYPES

According to the definition of Ontology, in computer science and information science, an ontology is a formal naming and definition of the types, properties, and interrelationships of the entities that really or fundamentally exist for a particular domain of discourse. It is thus a practical application of philosophical ontology, with a taxonomy.

An ontology compartmentalizes the variables needed for some set of computations and establishes the relationships between them.[24].

There are many concepts in an ontology that can be used to integrate with Fotobora in this chapter we will explain what we used from it to integrate with our system.

DBpedia provides with a very large net of ontologies providing many different classes and relationships in each one of them, however most of the information had to be trimmed in order to pass the only truly needed parts of it.

We analyzed the most common used ontologies such as Person, Animal and Place; however not everyone of them had an available RDF file, some of which were only available in either JSON or XML formats. A more manual approach was taken on some of them, using SPARQL queries over RDF files or over JSON files to inject the needed information over.

Despite the ontology having a graph like system, we implemented an hierarchic structure of entity types which would inherit properties based on their relationship with the other classes; for instance if we had the Entity Type of Animal and that had a property named as habitat this property would be inherited to the other nodes it gets derived from such as Horse or Dog which are two sub classes of Animal.

For Fotobora, the concept of Entity Type was reflected in these ontologies as the classes that had their property `rdf:type` as `owl:class`;, this we selected the classes we needed and selected the information that would fit in the system, this is detailed in Table 1:

4.4. Ontology and Entity Types

Table 1.: Properties used on Fotobora

Property in the Ontology	Use on Fotobora
rdf:type	Defines if the type of the rdf is of Class, which reflects as an Entity Type in Fotobora
rdf:isDefinedBy	Name We made no use of this property
rdf:label	This is the label that is used to name the Entity Type of the class
rdf:subclassOf	This provides the relationship between this Class and the father node Class
rdf:equivalentClass	In Fotobora there should be equivalent entity types, most are different even if they have the same name but a different relationship in the structure
wdrs:describedby	We made no use of this property
is rdfs:domain	These reflect the properties of the Entity Type
is rdfs:range	We made no use of this property
is rdfs:subClassOf	This provides all the subclasses associated with the class allowing to create the parental relationship between this Entity type and the provided other classes

The reason why the property range has not been used, is reflected on the fact that the property described cannot have any values associated with it, leaving no real reason for a pool of values. The type "Datatype" which is attributed to the properties described in the domain are usually measurable and debatable, leaving further room for discussion between the users; these properties have mostly been used to allow a better collaboration.

However an ontology in itself provides with a tremendous amount of data, the classes available in a system such as DBPedia are enormous and their properties entailed even if filtered still provide with a considerable list of information.

A considerable part of the information provided by the ontology is usually not very relevant to users; not only that it can also be overwhelming for the majority of inexperienced users it can also be lead to confusion and lack motivation to collaborate.

To address this we filtered the information based on the context of the application; Not only we limited the number of available entities, which was done by only importing branches of the ontology that made sense in what the system was all about. For instance, the ontology that was imported had one of the root classes named as Agent, one of the most common used entities is the entity Person which is by itself subclass of Agent. However Agent itself an entity that most users would opt to not use since it is too vague to be able to help in any sort of identification.

4.5. Poll and Voting

Additionally a lot of the used classes provide with an abundance of available properties, while most properties can be useful for any kind of identification they can also be not very useful for most of the context. With this we have filtered some of the available properties this was done by only importing a selected number of available properties through the `rdf:domain` variable in each of the classes.

The system is also flexible in a way that it allows any ontology to be implemented; it can provide with entities and types of entities that come from a family oriented ontology, or an ontology that simply provides mundane properties of what can be usually be represented through human taken photographs, such as old family pictures or a picture taken of a certain part of a historical city or site. Other types of ontologies can also be implemented such as one more focused to microorganism analysis, providing with tools that can allow scientists to identify bacteria, fungi and many others from microscopic photographs.

4.5 POLL AND VOTING

Most of the old photos archived by family generations have several questions on how to identify the people in them. By asking friends or some family relatives, it is possible to identify or at least have an idea of whom someone may be. However the doubt will most likely be there, and having a consensus could be difficult most of the time; much harder even would be to keep track of everyone's opinion and how confident they are about what they can identify in the picture.

As such, the the system is able to allow a community to vote (in favor or against) the previous tags and comments. In that way it will be possible to enrich the original photo with a lot of information gathered through crowd sourcing.

However the information added by one user can be incorrect. This raises the need for a Crowd-voting system where users collaborate or contest information with a certain degree of certainty. Altogether this provides a more consensual knowledge about that entity. This allows certain tags associated in the picture to have a certain consensus in the long run, instead of having the usual user not voting on certain tags because they aren't too sure about what or whom that entity might be.

This is valid to any entity type as well where users can contest if the entity that is being identified is in fact one or another; for instance a user can identify initially that the entity type of the entity itself is a monument, but another user can add new entity types, for example a Person, attempting to lure the other users that the entity that is being identified is not

4.6. Exporting the Photo Information as PDF Document

a monument but in fact a real person.

This is also associated with a voting system where users can agree or disagree with those entity types, the order of display is based on the score of each entity type, however in case of a tie of values, the entity type is named alphabetically.

For each property the system provides with a pool that lists the values that have been submitted by the users. Each value has a level of certainty associated with it and as a result a total score based on the combination of certainty that has been submitted. It starts with an initial certainty value which can go from 1 to 5, which represents if a user is not so sure (1) or if a user has absolute certainty (5) regarding that value. Any other user can add new values to that property and with those an initial level of certainty, as well as vote on other values submitted by other users, increasing their score depending on the level of certainty that has been submitted with the vote. Any user can also change their vote, for instance if they simply want to vote on a different value or even to reduce or increase the level of certainty of that specific vote.

In the end, the system will order the values of each property based on their score, allowing the value with the highest score to appear on top as well as highlighting the score of each one.

Furthermore the system also features the implementation of stories related to each entity or the photo itself, so that the users can express and share and even agree or disagree with the information that is being shared. This will allow an improvement of the explanation details of certain photos, such as events or historical scenes that would be rather interpreted through stories.

4.6 EXPORTING THE PHOTO INFORMATION AS PDF DOCUMENT

Users can at any time export their or other user's photograph's information as a Portable Document Files (PDF) which is exposed on the browser as well for initial visualization.

These include the most relevant information, such as it follows:

- Relevant basic information of the photograph such as Date Submitted, Name, Date Taken
- Total Number of Stories and the Total Number of Entities
- Brief information of its Photo Stories such as the Title, Summary, the user that submitted it, the date it was submitted and the number of comments related to that story

4.7. Import Annotations

- Brief information regarding its entities such as its type, who identified it, the date it was identified and the number of stories associated with each entity
- Detailed information of the photo stories with the stories that have been submitted
- Detailed information of its entities, such as the property values, their type, their current score and their initial score for each one of them

The algorithm to run this process is as follows:

- Retrieve all of the photograph's associated object values through ORM
- Calculates Number of Stories and Entities aswell as the best results from the voting
- Organizes the most relevant Entities and Orders the values of properties from entity types defined based on their score
- Associate all of those values to a View page in HTML; this page is not displayed to the user but instead kept in memory
- Validates the values of the View; if deemed valid it proceeds
- Parse the information from the view through a Ruby Gem named Wkhtmltopdf
- Creates the PDF using the plain HTML and CSS that has been provided to wkhtmltopdf in the back-end which renders HTML using Webkit
- Validates the end file result, if any errors were a result of its transformation the user is forwarded to an error view page instead
- Displays the PDF file result if no errors have been returned

4.7 IMPORT ANNOTATIONS

The system also support the possibility to import annotations of photographs, these annotations should describe the picture and will add additional information to the photograph with new tags.

Fotobora is dynamic when it comes to the description of its entities; the administrator of Fotobora's platform should be able to modify the types of entities and their properties that are valid in the description of its entities. This information has to be previously been written in an XML file following a provided XML schema that is available to download on the page itself.

4.7. Import Annotations

Each XML document will describe only a single photograph. With this, the element "Entities", which contains several real entities, identifies for each one of them their coordinates in the photograph and the type of entities in the form of an attribute.

The algorithm to run this process is as follows:

- Validates the XML file for its format and matches it with the XML Schema, if any validation errors occur they will be displayed on the web page
- Adds new values to the database based on the provided XML file that will reflect with new locations of tags on the photograph as well as the submitted values; the voting is assumed to be only with initial values

The schema that is used to validate the XML is depicted in [Figure 17](#).

4.7. Import Annotations

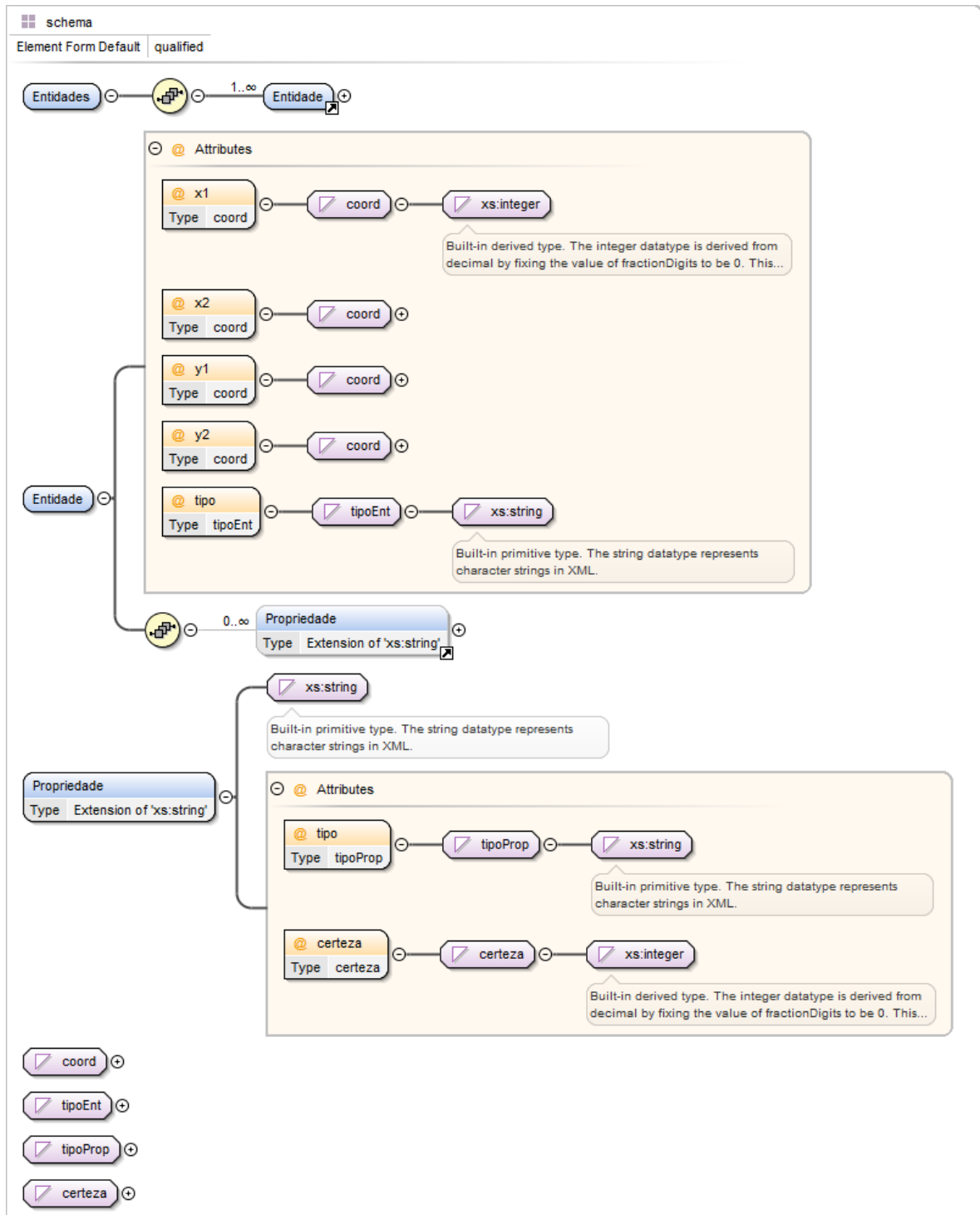


Figure 17.: Annotations Schema

4.8. Genealogical Trees

4.8 GENEALOGICAL TREES

Although the system allows any entity to be identified there is a common human necessity to identify people in photographs. Regardless if they are related family or not most users want to know what is the relationship between the entities in the photograph or even so if they are related with the user itself.

The system supports this feature by providing with a tool that allows users to upload a file containing a family tree in a format based on JSON. This family tree will be associated with each user and is replaceable on a new upload.

The grammar that was used to parse this kind of information is described below:

```
T = { num, str, '{', '}', '[', ']', ':', ',', '"arvgen"', '"name"', '"id"', '"idfather"', '"idmother"' }
N = { ArvGen, Entries, Entry, name, Id }
S = ArvGen
P = {
  p0: ArvGen -> '{' '"arvgen"' ':' '[' Entries ']' '}'
  p1: Entries -> Entry
  p2:      | Entry ',' Entries
  p3: Entry -> '{'
            '"name"' ':' name ','
            '"id"' ':' Id ','
            '"idfather"' ':' Id ','
            '"idmother"' ':' Id
            '}'
  p4: name -> str
  p5: Id -> num
}
```

Listing 4.1: Grammar Context Independant of the Genealogical Tree

One genealogical tree can easily be represented in an acyclic graph based structure . With this in mind, we created a grammar that obtains a group of entries, in which each one of them represents a person and it has a unique identifier. This entry will match a single node in the resulting graph. Each entry identifies the progenitors of each person, in which it will correspond with two edges at best.

```
{
  "arvgen" : [
    {
      "name" : "Joao Silva",
      "id" : 3,
      "idfather" : 1,
      "idmother" : 2
    }
  ]
}
```


4.8. Genealogical Trees

```
  },
  {
    "name" : "Ricardo Silva",
    "id" : 4,
    "idfather" : 1,
    "idmother" : 2
  },
  {
    "name" : "Luis Silva",
    "id" : 1,
    "idfather" : null,
    "idmother" : null
  },
  {
    "name" : "Mariana Silva",
    "id" : 2,
    "idfather" : null,
    "idmother" : 5
  },
  {
    "name" : "Joana Rodrigues",
    "id" : 5,
    "idfather" : null,
    "idmother" : null
  }
]
}
```

Listing 4.2: JSON example of the Genealogical Tree

The algorithm to run this process of import is described as follows:

- Verifies if it is in fact a file it is intended to be submitted
- Validates if the file is in a valid JSON format
- Once verified that the file is in fact a valid structure of JSON it is necessary to verify if the structure is depicted in the domain specific language, that is, if the information describes in fact the elements of a genealogical tree and if it has the necessary properties and these references between the elements are valid
- Verify if the key 'arvgen' exists and if this key has a value
- Verify if each element has the attributes name, id, idfather and idmother
- Verify if the elements name and id do not have null values
- Retrieve the values of the attribute id for every element

4.8. Genealogical Trees

- Test if the values of the references define the relationship between the elements of the genealogical tree
- Generate graphically the graph via a tool named Graphviz

To simplify the business logic and the usability of the application, the genealogical tree replaces the existing imported tree if applicable.

CASE STUDIES

After implementing the system, we tested it with different case studies in order to prove that the system was correctly implemented and that it satisfies the requirements stated in section 3.1.

With those goals in mind, it is described how the system was subjected to tests by exploring selected case studies. First, in section 5.1, it is discussed the use of Fotobora to demonstrate how the system can allow the identification of people through collaboration.

Then, in section 5.2, the application a Fotobora demonstrates another situation where it is shown how the system can also allow the identification of different entities in a completely different context, in this case in a medical point of view.

It is recommend to read Appendix A before going through this chapter to get a feeling about the use of Fotobora and a detailed explanation about the features it provides.

5.1 CASE STUDY 1

As the first case study, let us consider a Family scenario, in this case Vitor Armada's ancestry. Vitor Armada has several photographs at his disposal, these have been kept locked in time over the years and just recently whilst searching for old family pictures he came across one where he could not recognize any of the members in the photograph, he did had an idea but he wasn't absolutely sure.

To be able to obtain aid in the identification of those entities in the photograph Vitor Armada decides to start using Fotobora, since it can allow a collaboration between different users, it can provide with the information he needs in order to clarify their identity.

Assuming that Vitor Armada already has created an account in the system, to begin, he starts by uploading his photograph in the system (Figure 18 to 19); this will create a new photograph page in the system in which it will enable the users in the system to collaborate.

5.1. Case Study 1



Figure 18.: Vitor's Family Picture



by armada@fotobora.com

Figure 19.: Vitor's Family Picture Photograph Page

5.1. Case Study 1

Some of the members in the community of Fotobora are in fact closer relatives of Vitor Armada and may know some information regarding the just recently published photograph from Vitor. One of them is Amadeu which is a relative that is living in another country and has took an interest in Fotobora to allow a better understanding of the family roots; he took notice in Vitor Armada's photograph by searching in the system using the family name keyword and he knows he is the cousin relative from his account email(Figure 20).

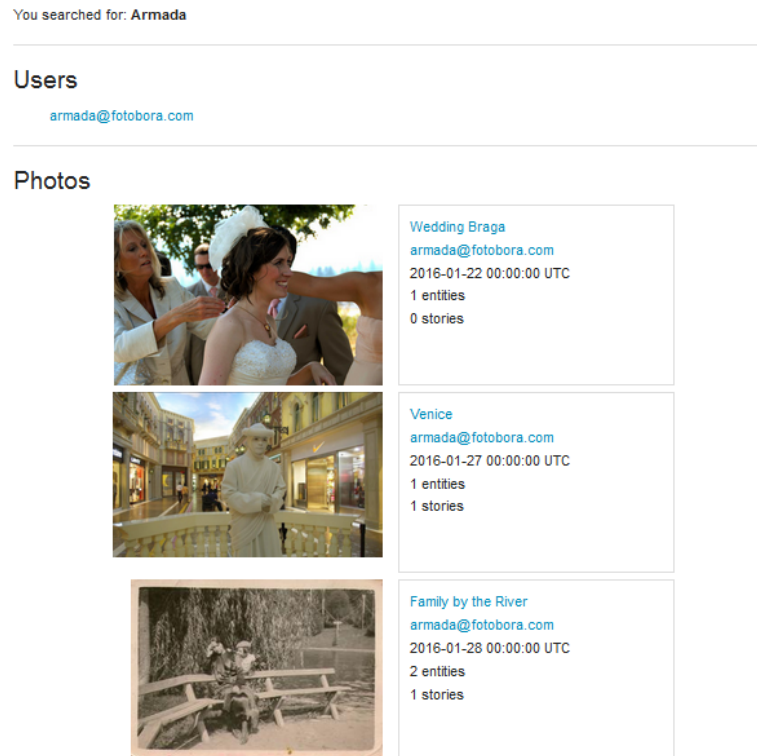


Figure 20.: Example of Amadeu using Fotobora's search engine

Amadeu then starts to contribute by navigating into Vitor's photograph page and immediately recognizes the lady sat in the bench. He then begins by navigating to the Identify Entity page and he creates a tagging box around the lady (Figure 21). He does not recognize the child next to the lady so he attempts to delimit the area to be represent her without any ambiguity.

5.1. Case Study 1



Select an area where the entity is in the picture above, then select a type and optionally fill out the properties.

Type

Name Certainty None Absolute

[Back to Photo](#) [Identify Entity](#)

Figure 21.: Vitor's Family Picture Photograph Page (Part 1)

Amadeu then submits the new entity type; redirecting him to the main photograph page and allowing every other user to be able to view his submission (Figure 22).

At the same time in the system, the other members of the family that are registered, can also contribute to the identification of the entity. Another member Cristina took notice that another member of the family has given up a first identification stating that the entity of the lady is in fact a person however she may not believe that the first identification is correct and that it might be another person.

She then agrees that the entity type is in fact of a person and ends up upvoting the value of the type itself, she navigates to the poll and notices that someone has already submitted a value of "Claudina Rocha Armada" to the Name property with a score of 5, which means that the person that identified was very sure about it.

However even though she believes it might be someone else she ends up creating a new

5.1. Case Study 1



[Back to Photo](#) [Create Story](#) [Add New Entity Type](#)

Identified by: armada@fotobora.com

Type: Person (0)



Properties

Name: Claudina Rocha Armada




Stories

No stories available. Be the first to add one!

Figure 22.: Vitor's Family Picture Photograph Page (Part 2)


value of "Maria Loureiro" which is the great grand mother of Vitor's mother side, however she isnt absolutely sure but very sure, so she ends up adding a score of 4 to the poll (Figure 23).

5.1. Case Study 1




Back to Photo Create Story Add New Entity Type

Identified by: armada@fotobora.com

Type: Person (1) 

Properties

Name: Claudina Rocha Armada 

Poll

Claudina Rocha Armada (5)

Certainty None Absolute

[Vote or Change Vote](#)

New Name

Value

Certainty None Absolute

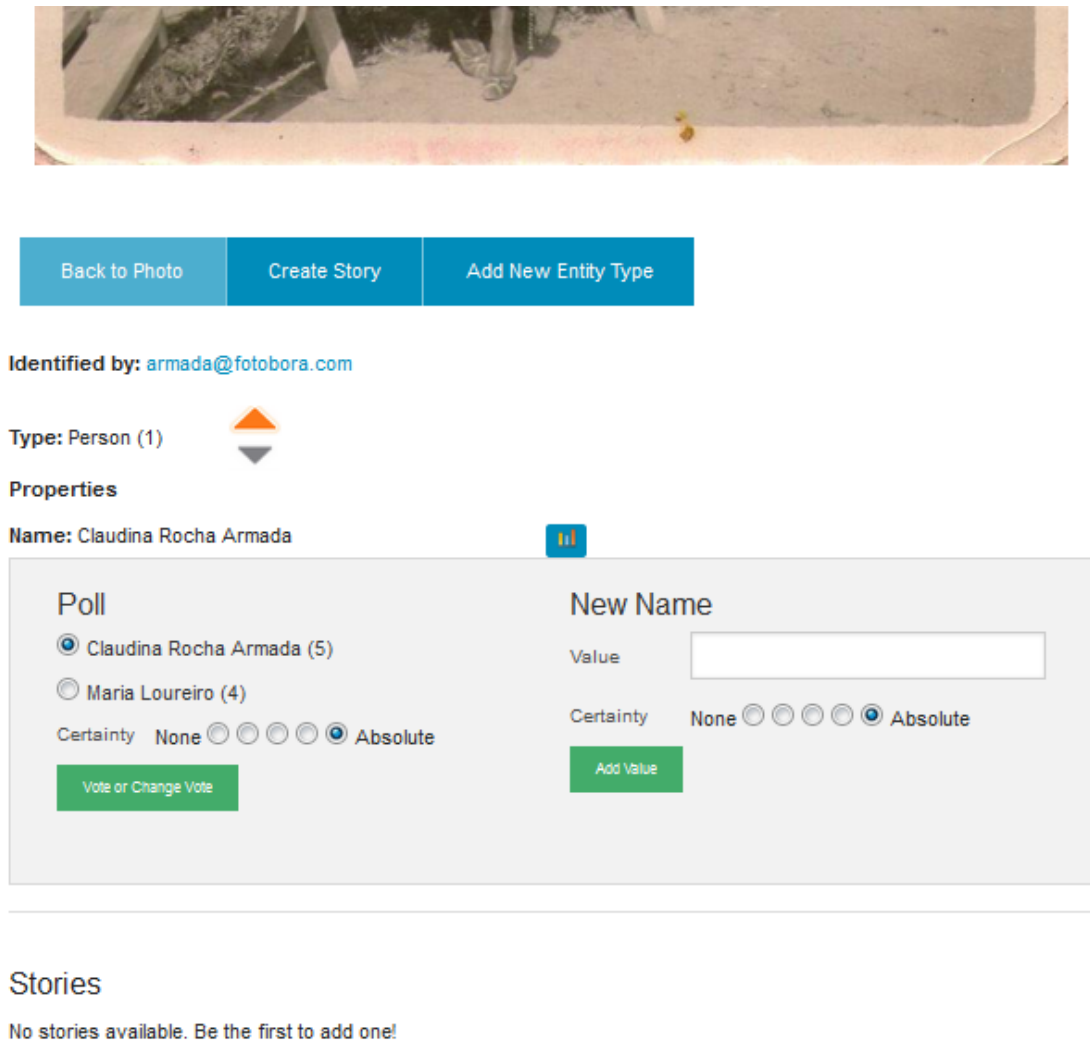
[Add Value](#)

Figure 23.: Vitor's Family Picture Photograph Page

Another member of the family, Pedro recognizes both of the child and the lady in the photograph, he knows from previous family pictures that the lady is in fact Vitor's great-grandmother and the child is Vitor's grandmother from his father's side instead.


He begins by identifying the child in the photograph, and adding the value of the photograph as "Pureza Rocha Armada" with an absolute level of certainty (Figure 25).

5.1. Case Study 1




Back to Photo Create Story Add New Entity Type

Identified by: armada@fotobora.com

Type: Person (1) 

Properties

Name: Claudina Rocha Armada 

Poll

Claudina Rocha Armada (5)

Maria Loureiro (4)

Certainty None Absolute

[Vote or Change Vote](#)

New Name

Value

Certainty None Absolute

[Add Value](#)

Stories

No stories available. Be the first to add one!

Figure 24.: Vitor's Family Picture Photograph Page

He then navigates to the other entity that has been identified, and analyses what has been contributed so far. He notices that someone has already identified the person as "Claudina Rocha Armada" in which he agrees with the value by voting with another absolute level of certainty, allowing an increment of 5 into the poll score of the Name property (Figure 26).

5.1. Case Study 1



Select an area where the entity is in the picture above, then select a type and optionally fill out the properties.

Type

Name Certainty None Absolute

[Back to Photo](#) [Identify Entity](#)

Figure 25.: Vitor's Family Picture Photograph Page



[Back to Photo](#) [Create Story](#) [Add New Entity Type](#)

Identified by: armada@fotobora.com

Type: Person (1)



52

Properties

Name: Claudina Rocha Armada



Poll

New Name

5.1. Case Study 1

After some time more members have contributed into the photograph, some of which contribute by also adding stories to the photograph itself and also to each one of the entity types, in which in this case they contribute by saying the genealogical relationship between Vitor and the people in the photograph.



Title

Melgaco

Text

This was taken by a professional photographer back in the days, in which your great grandmother and your grandmother posed by the river in Melgaco zone. It must have been in the 1920 where your grandmother was still a baby.

Back

Create Story

Figure 27.: Vitor's Family Picture Photograph Page

5.1. Case Study 1



[Back to Entity](#)

[Edit](#)

Grandmother Pureza

Submitted on 28 Jan 07:55 by armada@fotobora.com

This is your grandmother Pureza taken by the Melgaco river, she must have been about 10 years old at the time and this was taken before the family moved abroad to Argentina.

Comments

Add a comment

[Submit Comment](#)


Figure 28.: Vitor's Family Picture Photograph Page

As more and more members contribute to the identification of the entities in the photograph the community starts to reach some sort of consensus. With this Vitor navigates to the "Save to PDF" button which creates a PDF result of what has been added to the photograph (Figures 29 to 31).

5.1. Case Study 1

Page: 1 of 3 Automatic Zoom

Photo Details



Submitted by armada@klobuora.com on 28 Jan 07:00
Address: armada@klobuora.com
Date taken: 28 Jan 00:00

Figure 29.: Vitor's Family Photograph Report (Part 1)

5.1. Case Study 1

Page: 2 of 3 Automatic Zoom

Photo Stories

ID Title	Summary	Submitted by	Date Submitted	Number of Comments
3	Malgaco professional photographer back...	armada@fotobora.com	28 Jan 07:53	0

Entities

ID Type	Identified by	Date Identified	Number of Stories
3 Entity	Claudina Rocha Armada	armada@fotobora.com 28 Jan 07:29	0
4 Entity	Puraza Rocha Armada	armada@fotobora.com 28 Jan 07:14	-

Photo Stories (Detailed)

(#3) Malgaco

by armada@fotobora.com on 28 Jan 07:53

This was taken by a professional photographer back in the days, in which your great grandmother and your grandmother posed by the river in Malgaco zona. It must have been in the 1920 where your grandmother was still a baby.

Entities (Detailed)

(#13) (Person) Claudina Rocha Armada

by armada@fotobora.com on 28 Jan 07:29

Figure 30.: Vitor's Family Photograph Report (Part 2)

5.2. Case Study 2

{#3) Melgaco

by armada@fotobora.com on 28 Jan 07:53

This was taken by a professional photographer back in the days, in which your great grandmother and your grandmother posed by the river in Melgaco zone. It must have been in the 1920 where your grandmother was still a baby.

Entities (Detailed)

{#13) (Person) Claudina Rocha Armada

by armada@fotobora.com on 28 Jan 07:29

Type: EntityType

Properties:

Type	Value	Current Score	Total Score
Nome	Claudina Rocha Armada	0	5
Nome	Maria Loureiro	1	1

{#14) (Person) Pureza Rocha Armada

by armada@fotobora.com on 28 Jan 07:14

Type: EntityType

Properties:

Type	Value	Current Score	Total Score
Nome	Pureza Rocha Armada	5	5

Stories:

Grandmother Pureza

by armada@fotobora.com on 28 Jan 07:55

This is your grandmother Pureza taken by the Melgaco river, she must have been about 10 years old at the time and this was taken before the family moved abroad to Argentina.

Figure 31.: Vitor's Family Photograph Report (Part 3)

Vitor is then able to conclude that by several collaborations of other members that the two entities in the photograph are in fact his great-grandmother and her grandmother, that it was taken in Melgaco and potentially in 1920 just before the family have been to Argentina.

5.2 CASE STUDY 2

Fotobora also allows a completely different context of photographs and people to interact with it. In this case, let us consider a hypothetical hematologist named Richard Bradford. Richard usually works in a lab position in which he analysis films of blood through a microscope. However he came across a sample on which he wasn't completely on determining one component of the blood sample. He decided to use Fotobora in order to be able to ask other

5.2. Case Study 2

fellow hematologists or scientists regarding the blood film sample that he was working with.

The Fotobora system has been built using a Medicine oriented ontology in which the entity types available in the system suit the needs of the scientists or other medical doctors.

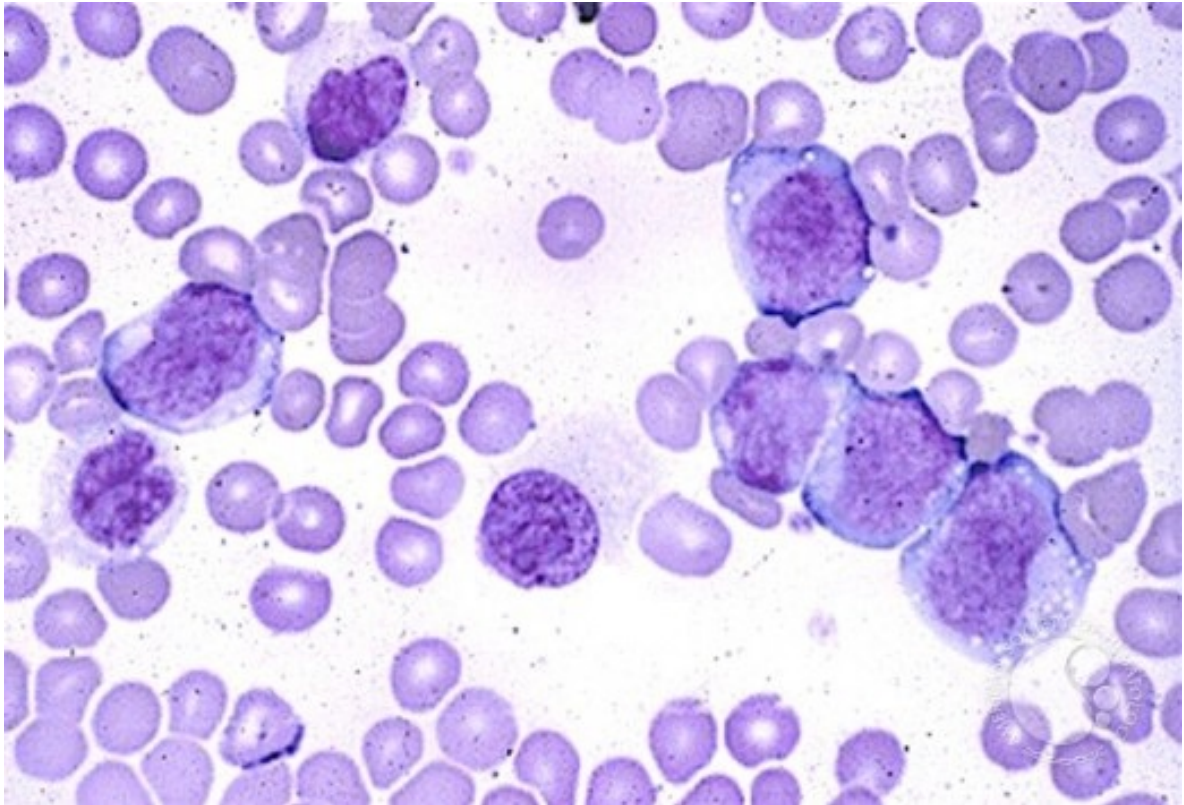
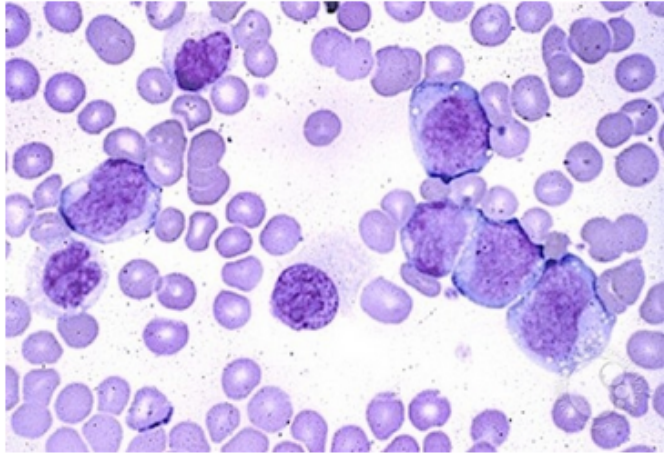


Figure 32.: Microscopic picture of a blood sample

Richard then begins by uploading the photograph onto the system (Figure33). This photograph is then available publicly for analysis for the remaining medical community (Figure 34).

5.2. Case Study 2

Fotobora armada@fotobora.com Edit registration Logout



[Edit](#) [Identify Entity](#) [Create Story](#) [Save to PDF](#) [Import Annotations](#)

by [armada@fotobora.com](#)

Title: Oxford Lab Blood Sample from Brighton
Photographed on: 29 Jan 00:00
Uploaded on: 29 Jan 03:47

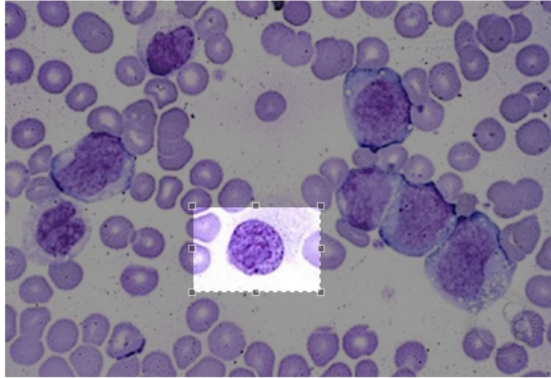
Entities:
No entities available. Be the first to add one!

Figure 33.: Blood sample upload

Richard then begins to identify a potential cell in the photograph; he assumes that the cell is a stem cell through his knowledge of hematology and also begins to contribute filling its properties based on what he can observe (Figures 34 to 36).

5.2. Case Study 2

New Entity



Select an area where the entity is in the picture above, then select a type and optionally fill out the properties.

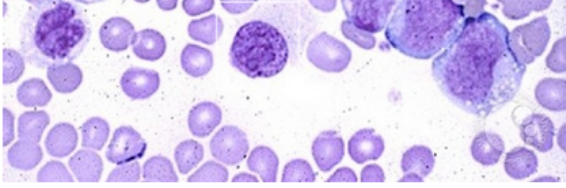
Type

Figure 34.: Tagging of a stem cell

Type	<input type="text" value="Stem Cell"/>		
Type	<input type="text" value="Hematopoietic"/>	Certainty	None <input type="radio"/> <input type="radio"/> <input checked="" type="radio"/> <input type="radio"/> Absolute
Stage	<input type="text" value="Adult"/>	Certainty	None <input type="radio"/> <input type="radio"/> <input type="radio"/> <input checked="" type="radio"/> Absolute
Living Being Holder	<input type="text" value="Human"/>	Certainty	None <input type="radio"/> <input type="radio"/> <input type="radio"/> <input checked="" type="radio"/> Absolute
Cluster of Differentiation	<input type="text" value="Human HSC"/>	Certainty	None <input type="radio"/> <input type="radio"/> <input type="radio"/> <input checked="" type="radio"/> Absolute
SLAM Code	<input type="text" value="CD150"/>	Certainty	None <input type="radio"/> <input type="radio"/> <input type="radio"/> <input checked="" type="radio"/> Absolute
LT-HSC	<input type="text"/>	Certainty	None <input checked="" type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> Absolute
ST-HSC	<input type="text"/>	Certainty	None <input checked="" type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> Absolute
Early MPP	<input type="text"/>	Certainty	None <input checked="" type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> Absolute
Late MPP	<input type="text"/>	Certainty	None <input checked="" type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> Absolute


Figure 35.: Property fill of the stem cell

5.2. Case Study 2



Back to Photo Create Story Add New Entity Type

Identified by: vitor@fotobora.com

Type: Stem Cell (1) 

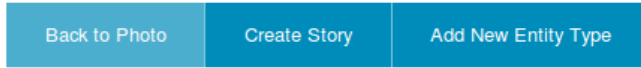
Properties

Type: Hematopoietic	<input type="checkbox"/>
Stage: Adult	<input type="checkbox"/>
Living Being Holder: Human	<input type="checkbox"/>
Cluster of Differentiation: Human HSC	<input type="checkbox"/>
SLAM Code: CD150+	<input type="checkbox"/>
LT-HSC: CD34-	<input type="checkbox"/>
ST-HSC: CD34+	<input type="checkbox"/>
Early MPP: CD34+	<input type="checkbox"/>
Late MPP: CD34+	<input type="checkbox"/>

Figure 36.: Initial display of property poll of the stem cell

Other hematologists then begin to contribute by sharing their knowledge over what they can identify in this stem cell. Allowing the community to reach a higher consensus of what it really is. However other medical doctors have a suspicion that the steam cell might actually be of a cancerous type. The community then ends up adding a new entity type to the polling, in this case "Cancer Stem Cell". (Figures 37 to 39). These two types entities are very similar, in fact the entity type Cancer Stem Cell inherits the properties of Stem Cell since it its subclass. With this, the system shares the poll between the properties of both entity types, preventing repetition or loss of information thats has been submitted on previous polls.

5.2. Case Study 2



Identified by: [vitor@fotobora.com](mailto: ritor@fotobora.com)

Type: Stem Cell (2)



Properties

Type: Hematopoietic



Poll

Hematopoietic (10) ✓

Granulocyte-macrophage (3)

Megakaryocyte-erythroid (1)

Certainty None Absolute

[Vote or Change Vote](#)

New Type

Value

Certainty None Absolute

[Add Value](#)

Stage: Adult




Living Being Holder: Human




Figure 37.: Polling of the stem cell's properties

5.2. Case Study 2

Type: Cancer Stem Cell (1) 

Properties

Type: Hematopoietic 

Poll


Hematopoietic (10) ✓
 Granulocyte-macrophage (3)
 Megakaryocyte-erythroid (1)


Certainty: None Absolute


New Type


Value:


Certainty: None Absolute


Stage: Adult 


Living Being Holder: Human 


Cluster of Differentiation: Human HSC 

SLAM Code: CD150+ 

LT-HSC: CD34- 

ST-HSC: CD34+ 

Early MPP: CD34+ 

Late MPP: CD34+ 













Disease: - 

Figure 38.: Polling of the cancer stem cell's properties

5.2. Case Study 2

Type: Cancer Stem Cell (1) 

Properties

Type: Hematopoietic 
Stage: Adult 
Living Being Holder: Human 
Cluster of Differentiation: Human HSC 
SLAM Code: CD150+ 
LT-HSC: CD34- 
ST-HSC: CD34+ 
Early MPP: CD34+ 
Late MPP: CD34+ 
Disease: Myelofibrosis 

Poll

Myelofibrosis (14) ✓
 Lymphocytic leukemia (3)

Certainty None Absolute


New Disease

Value

Certainty None Absolute

Figure 39.: New property of the cancer stem cell

5.2. Case Study 2



Back to Entity

Myelofibrosis cancerous cell

Submitted on 29 Jan 02:24 by [vitor@fotobora.com](mailto: ritor@fotobora.com)

The abnormally low levels of red blood cells represented here as well as the strangely mutated stem cell indicate that this might be the sign of Myelofibrosis, I have dealt with patients in the past and the images shown here are very similar to what we had to visualize in the lab.

Comments

[sergio@fotobora.com](mailto: sergio@fotobora.com) on 29 Jan 02:26:
How can you tell the red blood cells are low? I have dealt with issues like this in the past and it could just be too early to conclude the cell has been mutated.

Add a comment

Submit Comment

Figure 40.: Story sharing and comment dispute

The two case studies introduced along this chapter were crucial to test the use of Fotobora in completely different contexts of usage. With them it was possible to verify the system performance in different scenarios.

CONCLUSION

This masters' project is entitled "Interactive tagging of Photographs in Cooperative systems". The underlying research hypothesis is: *Is it possible, in a proper and effective way, to build a system that supports the identification of entities in photographs through collaboration?*

To prove that hypothesis, several tasks, described below, were accomplished, giving rise to the project contributions:

- Study on how the new technologies of the semantic web can contribute to the implementation of a system that allows a better identification of entities in photographs through collaboration;
- Study how crowd-sourcing and on-line social networks can aid in the development of collaborative systems: what social networks can provide collaboration through tagging, either it being more generic or specific, and what information do they provide.
- Compare existing solutions able to use crowd-sourcing and crowd-voting to obtain information regarding photographs while identifying their weak points.
- Study how ontologies could support the information required to assist in the identification of entities in a structured matter and how they could be integrated.
- Study what other features could be additionally implemented to aid in further collaboration between the users.
- Apply the conducted research to implement a system capable of satisfying the objective of this master work, and detail the implementation of that system.

SEMANTIC WEB AND ANALYSIS

This master work begun with a thorough research phase. In this phase, we conducted a research over what the Semantic Web could contribute in standardizing tagging in photographs. Many websites provide a set of standards that allows any user to include information in a

standardized manner and over the years it has improved to the point where most data can already be shared, processed and understood by machinery and people . The use of semantic technologies makes it possible to describe context of the information that is being exchanged, allowing the possibility to develop an idea where the collaboration could be taken on something more visual, in this case with photographs.

CROWD SOURCING AND ONLINE SOCIAL NETWORKS ANALYSIS

The research continued by investigating additional existing Online Social Networks (OSN), and how users interacted with them using any kind of crowd-sourcing. These OSN in their majority did not provide any kind of useful crowd-sourcing among their users, in fact the whole concept of crowd-sourcing and OSN were not being used that much especially if it involved photographs.

Facebook appeared to be a very popular choice for overall photograph sharing and identification, however none of this was ever being related with crowd sourcing, and most other crowd sourcing projects that did had some sort of OSN did not actually provide a similar functionality to interactive tagging.

With this an idea to be able to mix up both of these concepts came to fruition.

EXISTING SOLUTIONS ANALYSIS

Existing solutions that shared the objective of this master work were also studied. The purpose of this analysis was to categorize the existing solutions by their features, their limitations and so that they could be addressed, in order to propose a solution that would overcome those weak points and arise as objectively better. The major issues that are common between the solutions are:

- Not using any generic sort of standardized data; most solutions only had a very specific goal and something very concrete on what could be identified;
- Being limited to what every user would initially identify, it was immune to any contesting in a structured matter
- There is no concept of level of certainty towards anything that is being identified in the photographs; most would have to assume that anything identified is of absolute certainty, or just simply not and that was never taken in consideration. Also, because of this, these systems are less reliable and trustworthy, especially when the contribution is very limited.

- Not following an ontology like system and only allowing a single type of entity to be identified, there was no concept of different potential types of entities

CONCEPTION OF THE PROPOSED SOLUTION

The proposed solution allows an improvement towards all of the alternative solutions that were examined.

Many of the noted issues that were addressed by the conceptualization of a system that allows users to share and upload photographs and mixing this with the idea of identification through collaboration.

The idea of identification was depicted through the tagging of something; something of which was specified as an entity and this entity has a type, properties associated with it and values for each one of those properties.

The truthfulness aspect in an identification was also taken in consideration, allowing the conception of level of certainty, which was later applied to several elements in the process of collaboration.

Concepts such as crowd voting that were previously analyzed were also taken into consideration, this would project the idea of allowing any user to be able to vote regarding any user submitted information in the photograph.

In a formal matter the proposed system's requirements were specified in order to allow an overall view of how the proposed system enables an advanced level of interactivity as well as the entailment of additional features that would stimulate it further.

The proposed system's architecture was specified, with a throughout description of all its components that it would require. Its data model was also defined and how it would provide with a dynamic and generic conceptualization of any entity.

ONTOLOGY ANALYSIS AND INTEGRATION

The concept of Ontology has always been a very philosophical and very broad term. We analyzed what useful information ontologies could provide in order to standardize the information that could be submitted by the users of the application. This study was aimed at addressing the major issues that the existing solutions have, and how the ontology could be a key to overcome them.

In this analysis, several online available ontologies were taken into consideration for potential integration; they were available in several different formats and provided with a broad range of distinct context of information.

From these ontologies, two major ones were thoroughly examined and described: the first was the "DBPedia" ontology in which it provides with a range of different formats and a large

standardized ontology that is subdivided into several branches; the other is the "Protégé" library which provides with many distinct separated ontologies as well as with new concepts of relationship.

Both of the available libraries have advantages and disadvantages, which were explicitly stated. For the purpose of the solution's implementation we decided to pick one for convenience of integration, despite both being viable.

IMPLEMENTATION OF THE PROPOSED SOLUTION

With all of the research and investigation completed, the proposed system was implemented, and it has been baptized as "Fotobora".

The proposed solution came to life with the implementation of a multi-featured system that is capable of everything that was proposed and, due to the analysis and testing of the application itself, new extra features were implemented in order to satisfy the user's interactivity with the application.

To illustrate how the implemented solution functions, the most important features were presented via a series of instructions and figures. With that purpose as well, two case studies were defined and presented. These represent two realistic uses of the application, and featured most of the features that Fotobora has to offer.

The first case study depicts a story about a young member of a family, Vitor Armada. It displays how in a closed family environment, Fotobora allowed to help him identify potential family members in an old photograph through the aid of collaboration from other users in the system, in this case friends and family of Vitor; it also showed how the system organized their entities and their properties based on a family focused ontology. This allowed Vitor to create a PDF file report based on what his family has been debating over, giving him a better insight on who those persons might be, in this case, with a major consensus he managed to identify that the people in the photograph were his grand-grand mother and his grandmother.

The second case study showed how a hematologist interacted with Fotobora in order to identify potential stem cells that he was analyzing in his laboratory; it also depicted how the system was flexible to the point of allowing a medical ontology to tailor how the entities and properties of the system would be available for the medical doctors and scientists that were collaborating. Furthermore in this particular case, the case also depicted a situation where the users were indeed identifying something that was correct (a stem cell) but later in the poll the majority agreed that the stem cell was indeed of a cancerous type, allowing a new type of entity to be added; this new entity type is a subclass of stem cell, as of so it inherits

its properties; with this both types of entities are correct and the new type of entity has new properties that are relevant to the context of a cancerous type, in this case the disease. The other properties are shared with the other entity types, and for this reason both of the polls are too, allowing a further debate over the new entity type.

FUTURE WORK

This master work proved that it is possible and viable to create a system that allows an interactive tagging of photographs in collaborative environments. It also explored the possibilities of its integration with ontologies, proving that it can be flexible to nearly every contextual case which is done for two selected case studies; also how it can potentially integrate with genealogical trees if in fact the system is of a family context. Possible future work could entail:

- In this master work, it was taken a manual approach towards many of the used ontologies that were available through out the communities. The ontologies were closed and were strictly defined based on the original author of each one of them, and it was necessary a certain level of caution when it came to keep the integrity of the database, based on the addition of new branches of another ontology.

Additional work could also be done with the intent of creating a self sustained ontology in which the users contribute in the creation of an overall ontology that could potentially suit any context of any photograph. The ontology would grow and reconstruct itself based on the input of the users, this could also be subject to a crowd voting system in which the ontology itself could be fruit of collaboration.

- In Fotobora, genealogical trees are used as a complement to the system's identification goal. However these genealogical trees are submitted by the users themselves and are limited to Fotobora itself. The semantic web provides with a set of standards that could standardize this information.

It could be feasible to integrate these genealogical trees with existing systems that already structure family relationships. Websites such as MyHeritage that already have an advanced system and an appealing visual system to represent family genealogical trees could be used to import information from both ways.

BIBLIOGRAPHY

- [1] Anderson, Paul (2007), *All that Glisters is Not Gold- Web 2.0 and the Librarian Vol 39* .
- [2] Social Science Computer Review 27.4 (2009), *Mapping for the Masses: Accessing Web 2.0 through Crowdsourcing* Pages 524-538.
- [3] Library Technology Reports 43, *The Ongoing Web Revolution*, no.5 (2007): Pages 10-14.
- [4] Warschauer, M. and Grimes, D.(2007) Audience, authorship, and artifact: The emergent semiotics of Web 2.0. ,*Annual Review of Applied Linguistics*, 27, Pages 1-23.
- [5] Oder, Norman (2008). "LC Seeks Community Tagging." *Library Journal* 133, no.4 : Page 20.
- [6] Howe, Jeff (2008), *Crowdsourcing: Why the Power of the Crowd is Driving the Future of Business* , The International Achievement Institute.
- [7] Porter, Joshua (2008). *Designing for the Social Web*. Berkley, CA: New Riders. pp. 1–32. ISBN 978-0-321-53492-7.
- [8] Butler, John (May 2008). "The Family Diagram and Genogram: Comparisons and Contrasts". *American Journal of Family Therapy*. 12 36 (3): Pages 169–180.
- [9] Enrique Estellés-Arolas and Fernando González Ladrón-de-Guevara, researchers at the Technical University of Valencia.
- [10] Arnott, Rebecca (2013-09-07). "uropeana1914-1918.eu". Europeana1914-1918.eu.
- [11] [The Great War Archive | First World War Poetry Digital Archive](http://TheGreatWarArchive.org)". Oucs.ox.ac.uk.
- [12] Arrington, Michael (2006-09-01). "Google Image Labeler Uses Human Labor". [techcrunch](http://techcrunch.com).
- [13] [Hockey Riot 2011 | Vancouver Police Department](http://HockeyRiot2011.com)". Vancouver.ca.
- [14] McLaughlin, Elliott C. (August 18, 2014). "Image overload: Help us sort it all out, NASA requests". *CNN*. Archived from the original on August 22, 2014. • [Link to NASA portal](#).
- [15] "Folksonomy and science communication". *Information Services and Use* (27): 97–103. 2007. – Folksonomies as a tool for professional scientific databases

Bibliography

- [16] Vander Wal, Thomas. "Explaining and Showing Broad and Narrow Folksonomies".
- [17] "Delicious.com Site Info". Alexa Internet. Retrieved 2015-01-04.
- [18] Tim Berners-Lee, James Hendler, Ora Lassila (2001) .
- [19] Pollock and Hodgson, 2004, Adaptive Information
- [20] <http://www-ksl.stanford.edu/kst/what-is-an-ontology.html>
- [21] <http://www.w3.org/standards/semanticweb/ontology>
- [22] Xiang Zhang, Gong Cheng, Yuzhong Qu, Ontology Summarization Based on RDF Sentence Graph, School of Computer Science and Engineering Southeast University Nanjing 211189, P.R. China
- [23] Gruber, Thomas R. (June 1993). "A translation approach to portable ontology specifications" (PDF). *Knowledge Acquisition* 5 (2): 199–220. doi:10.1006/knac.1993.1008.
- [24] Arvidsson, F.; Flycht-Eriksson, A. "Ontologies I" (PDF). Retrieved 26 November 2008.
- [25] Burbeck, Steve (1992) Applications Programming in Smalltalk-80(TM):How to use Model-View-Controller (MVC)
- [26] Simple Example of MVC (Model-View-Controller) Design Pattern for Abstraction
- [27] Tate, Bruce; Hibbs, Curt (August 22, 2006). "Ruby on Rails: Up and Running" (First ed.). O'Reilly Media: 182. ISBN 0-596-10132-5.
- [28] Thomas, Dave; Fowler, Chad; Hunt, Andy (April 28, 2009), Programming Ruby 1.9: The Pragmatic Programmers' Guide (Third ed.), Pragmatic Bookshelf, p. 1000, ISBN 1-934356-08-5.
- [29] Komplette verlinkt - Linked Data" (in German). 3sat. 2009-06-19. Retrieved 2009-11-10.



FOTOBORA FEATURES

Of all the system's features, the following were deemed most important and will be illustrated:

- Photograph Display
- Tagging of entities;
- Voting on the entity type and their properties;
- Pooling of new values to an entity's property;
- Contribute with stories and comments;
- Exporting the information identified in the Photograph;
- Associate different ontologies;
- Import Annotations;
- Support for Genealogical Trees.

A.1 PHOTOGRAPH DISPLAY

After logging in the system (Sign in, or Sign up) a user can import a genealogical tree, choose a photograph (in his own collection or picking it up from the other's collection) to work on it, or upload a picture (remember figure 14 for details on the user's homepage). If he decides to create a new Photo, the main Photo page is presented to him. This page is shown in figure 42. At this point there are no entities identified and no stories added. The only information available is the title of the Photograph, the user that submitted it and the date that it has been submitted as depicted in Figure 42.

Below the photo that has been submitted by the user, it is possible to visualize a menu bar with different buttons (see Figure 42) which provides several functionalities:

A.1. Photograph Display

- Edit - Enables the user to edit the basic information of the photograph, such as the Name, Date and the Picture itself. Note that by editing the photo it destroys the previous tags it had as well as the votes associated with it;
- Identify Entity - Redirects the user to the Identify Entity Page;
- Create Story - Redirects the user to the Create Story Page;
- Safe to PDF - Converts all of the obtained information into a PDF result page;
- Import Annotations - Redirects the user to the Import Annotations Page;

Fotobora sergio@fotobora.com Edit registration Logout

New Photo

Name

Date

Picture

 1.jpg

Figure 41.: Photo creation page

A.1. Photograph Display

Fotobora armada@fotobora.com [Edit registration](#) [Logout](#)



[Edit](#) [Identify Entity](#) [Create Story](#) [Save to PDF](#) [Import Annotations](#)

by armada@fotobora.com

Address: Wedding Braga
Photographed on: 22 Jan 00:00
Uploaded on: 22 Jan 01:01

Entities:
No entities available. Be the first to add one!

Stories
No stories available. Be the first to add one!

Figure 42.: Photograph display page

Figure 43 shows the photo edition, where the creator of the photograph can edit the meta-information about the photograph. The user is also able to change the photograph itself, but by doing so the system will erase the previously submitted values of the entities tagged in the photograph.

Fotobora
armada@fotobora.com
Edit registration
Logout

Edit Photo


Address

Date

Picture

No file selected.

Current Picture:



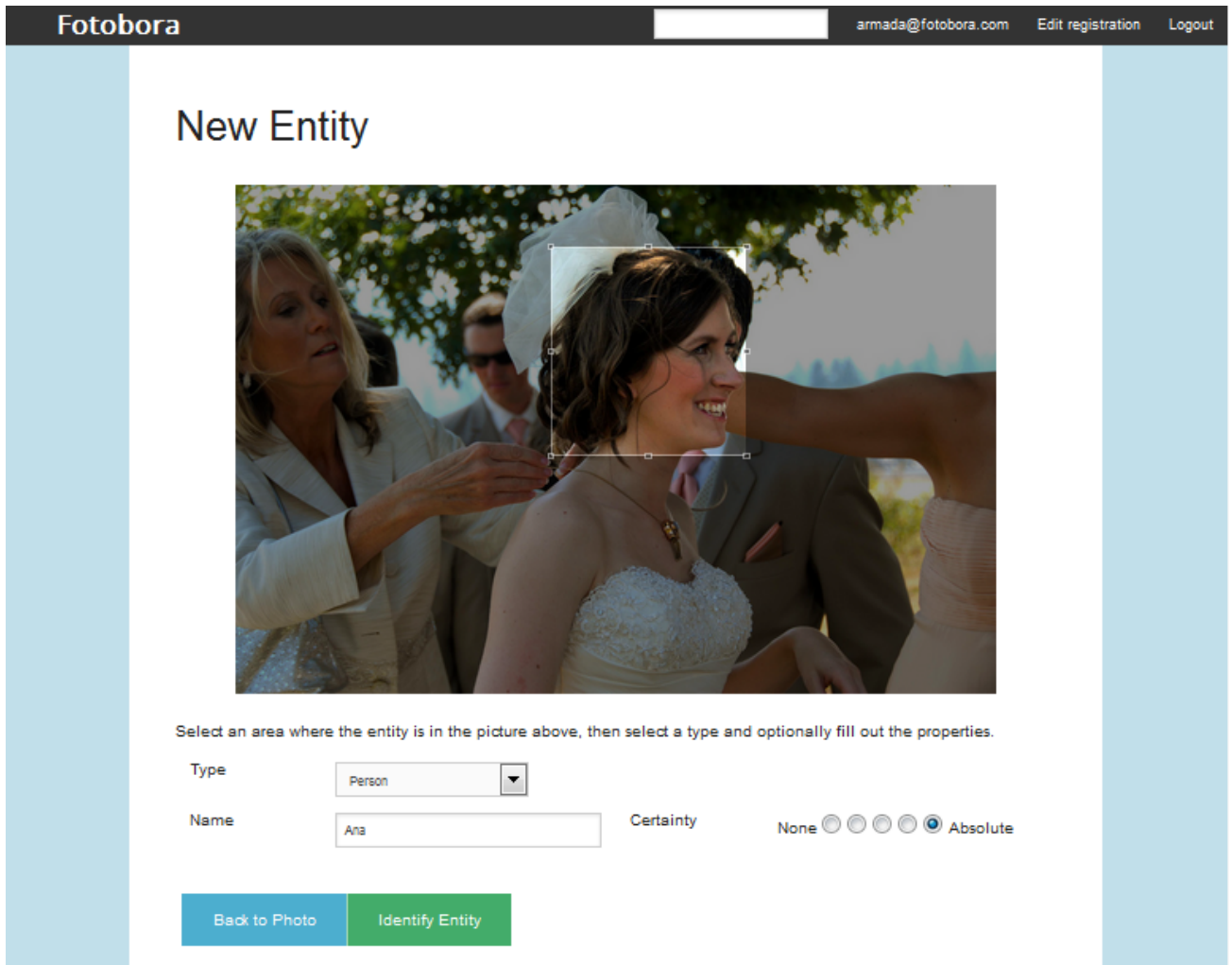
Back to Photo
Update Photo

Figure 43.: Edit Photo

A.2 TAGGING OF ENTITIES

From the navigation menu available in the middle of the photograph display page (see Figure 42) the user is able to press on "Identify Entity" and redirect himself to the Identify Entity page. In this page the user is allowed to tag anything in the photograph. This is done by dragging around the photograph the rectangular shape that is initially positioned on the upper left side. The user can scale the size of the desired area to be tagged by dragging the square dots on the borders of the rectangle. This can be visualized on Figure 44.

A.2. Tagging of entities



Fotobora

armada@fotobora.com Edit registration Logout

New Entity

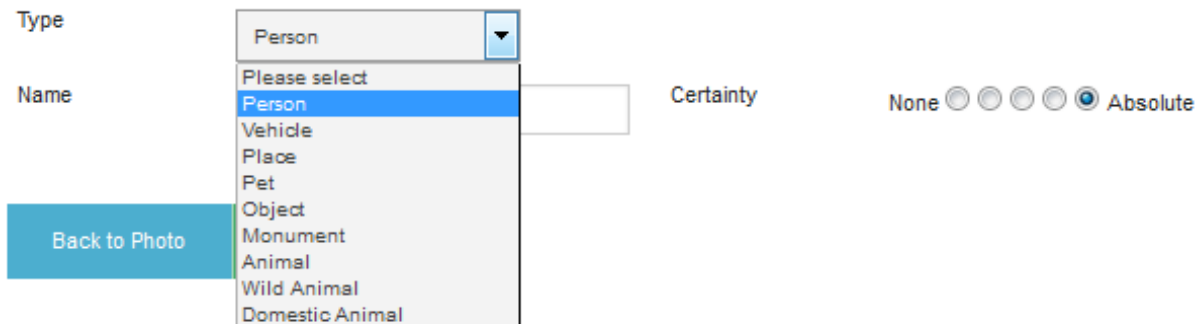
Select an area where the entity is in the picture above, then select a type and optionally fill out the properties.

Type:

Name: Certainty: None Absolute

Figure 44.: Identify Entity Page

Once the area has been selected the user has the possibility to select the type of entity. Figure 45 shows short list of common types of entities.



Type:

Name:

Certainty: None Absolute

- Please select
- Person
- Vehicle
- Place
- Pet
- Object
- Monument
- Animal
- Wild Animal
- Domestic Animal

Figure 45.: Identify Entity : selecting the entity Type

A.2. Tagging of entities

The entity types have different properties associated. Depending on the type that is selected, the system will adapt and display above the type box, the text boxes corresponding to the associated properties. Figure 46 shows two properties (Inauguration Date and Name) associated with the type "Monument" and Figure 47 illustrates the properties associated with the type "Person" (in that case, just Name). Each one of these properties has a certainty level associated (with 5 possible grades from None to Absolute) that represents the degree of confidence that the user has in the value assigned to the respective property.

Type	<input type="text" value="Monument"/>							
Inauguration Date	<input type="text"/>	Certainty	None	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/> Absolute
Name	<input type="text"/>	Certainty	None	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/> Absolute

Figure 46.: Identify Entity Page

Once the user submits the form by pressing the button "Identify Entity", the user is redirected to the Entity page where the system will show a message if the entity has been created successfully, as this can be seen in Figure 47.

In case of success, a hyper-link is created associating annotation with the entity on the photograph. The effect of this action can be seen clicking on the selected area in which the entity has been tagged.

Additionally it will also create a mouse-over link on the selected area, showing the entity type as well as the best value of the entity type. On its creation the mouse-over will show information that has been submitted initially by the user that created it.

A.3. Voting on the entity type and their properties

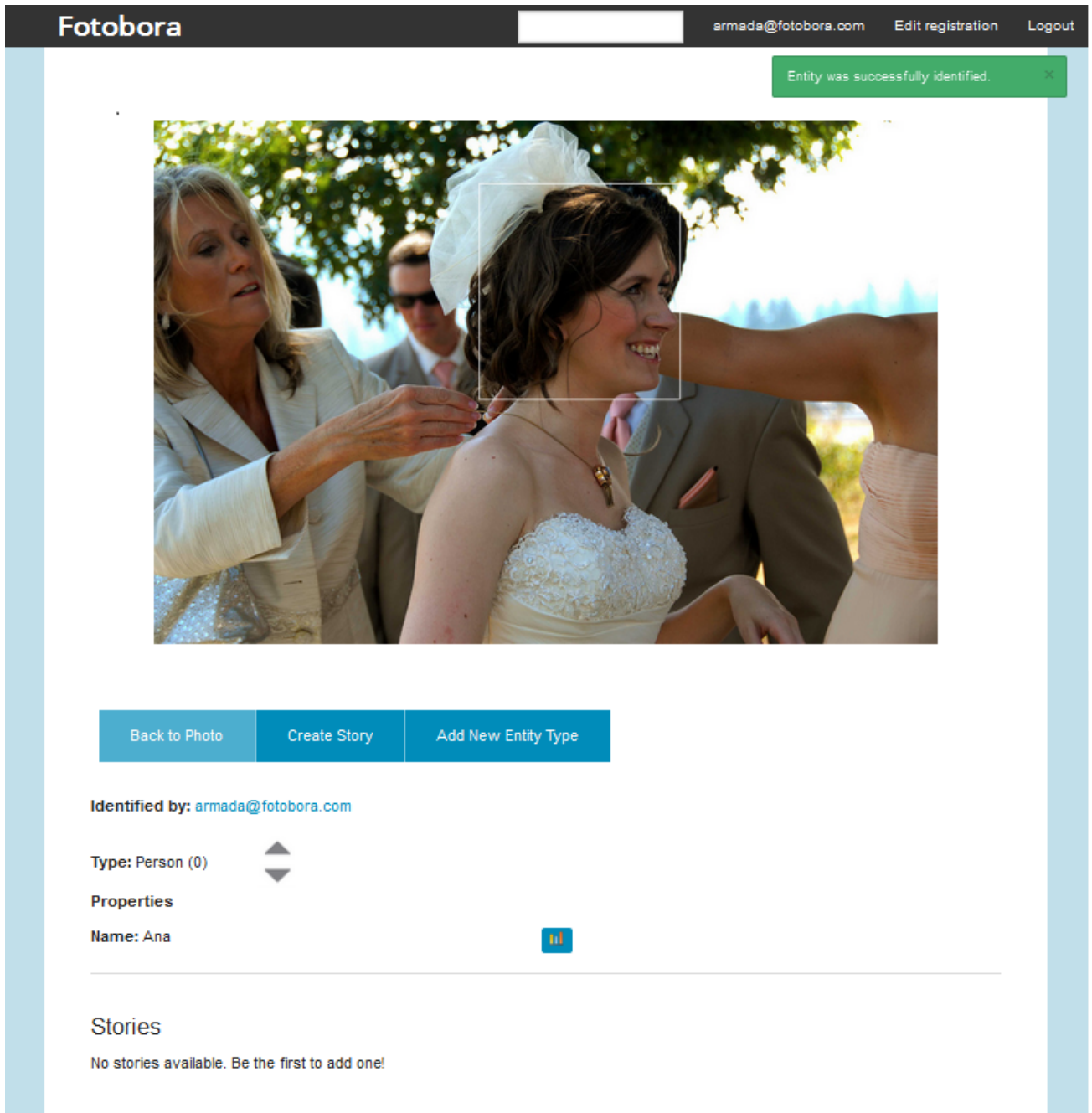


Figure 47.: Identify Entity Page

A.3 VOTING ON THE ENTITY TYPE AND THEIR PROPERTIES

One of the biggest features offered by Fotobora is the collaboration aspect that is implemented through crowd-voting. The system assumes that every entity identified in a photograph can be associated with a poll of votes. Figures 48 to 56 respectively show examples of a user

A.3. Voting on the entity type and their properties

identifying an entity in a photograph whilst being subjected to the voting system that is available to the other users.



Select an area where the entity is in the picture above, then select a type and optionally fill out the properties.

Type ▼

Name Certainty None Absolute

[Back to Photo](#) [Identify Entity](#)

Figure 48.: New Entity Type Page

A.3. Voting on the entity type and their properties

In Figure 48 we can visualize the tagging of an entity (namely, the head and shoulders of a man's statue, located in the middle of a street) in a photograph. In this scenario the user that created the entity is assuming that the entity tagged is in-fact a person and that person goes by the name of Paul Enrico Alba a known statue impersonator. It is also assumed that the user is very sure but not absolutely sure about his tag.

Other Fotobora users can support this original tagging by up-voting the choice. However other users can disagree with the entity type selected and exhibited, by either down-voting or by adding new types of entities (Figure 49). This is done by pressing the Add New Entity Type button and creating a new Entity Type; the current entity types proposed by different users are listed associated to that photograph, prevent potential repetition.

Once the input has been validated the system will reply with a pop up message stating that the update has been successful or unsuccessful (Figure 50).

A.3. Voting on the entity type and their properties



Back to Photo Create Story Add New Entity Type

Type: ▼

Inauguration Date:

Name:

Certainty: None Absolute

Certainty: None Absolute

Figure 49.: New Entity Type example

As said above, if there are more than one entity type, associated to an entity in a photograph, the system will display several polls of types of entities (Figure 51). Each poll has a particular score associated with the entity type. This score can be either positive or negative (Figure 52) and it starts with a default value of zero.

A.3. Voting on the entity type and their properties

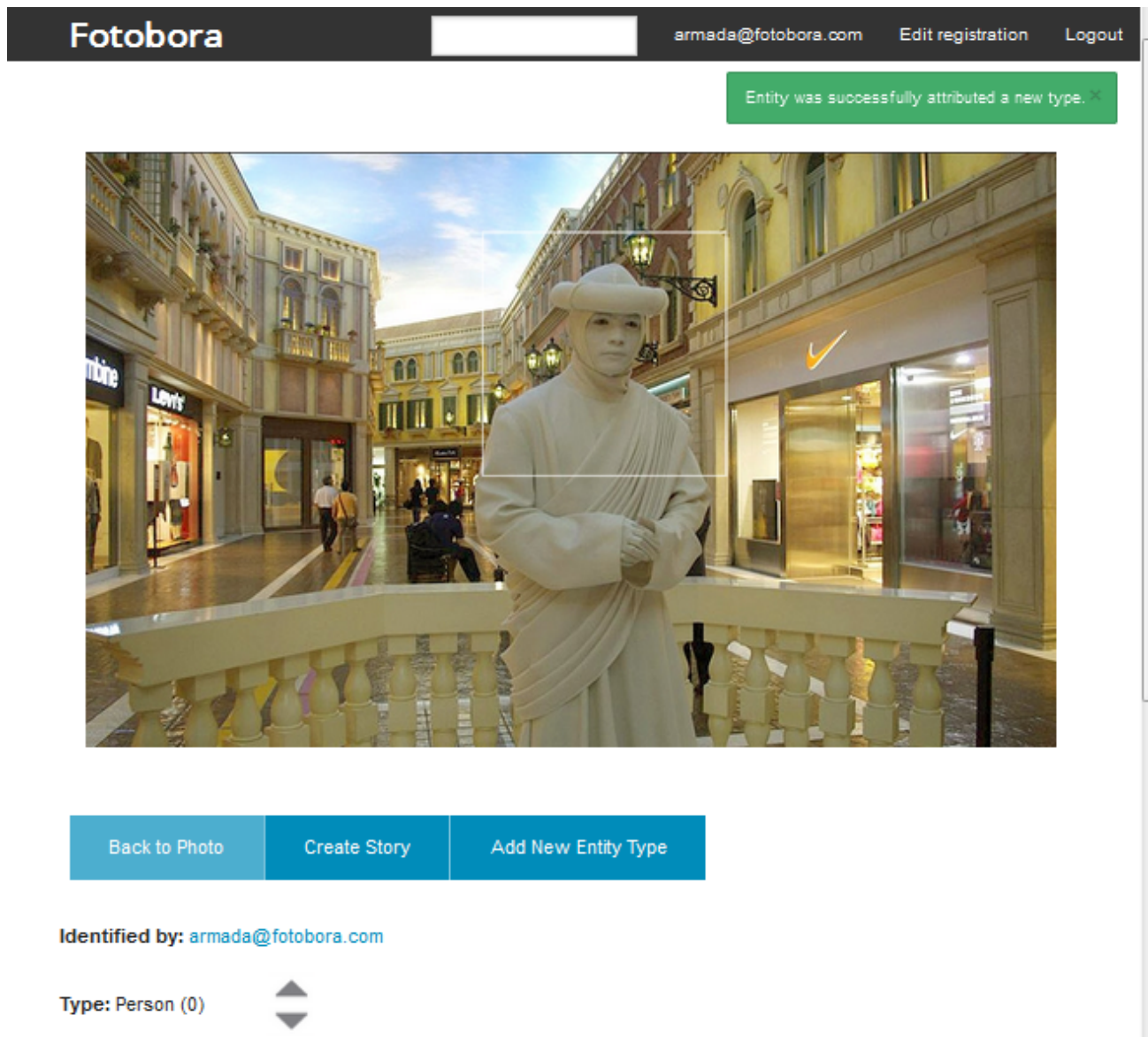


Figure 50.: New Entity type submission

Any user may also Up-vote or Down-vote any of the possible entities influencing its current score and affecting the order of the displayed entity types. At any point a user is able to cancel his own vote; this is done by clicking on the Up-vote or Down-vote button that is highlighted.

A.4. Polling of new values to an entity's property



Identified by: armada@fotobora.com

Type: Person (1)



Properties

Name: Paul Enrico Alba



Type: Monument (0)



Properties

Inauguration Date: -



Name: Venetian Hotel Statue



Stories

No stories available. Be the first to add one!

Figure 51.: Score example on poll

A.4 POLLING OF NEW VALUES TO AN ENTITY'S PROPERTY

As stated in chapter 4.5

Every single entity type that is identified in a photograph is subjected to its own poll of values. Every type has different properties and every single property has a poll associated. This can be accessed by clicking on the small bar graph icon right next to the property. The system will display outside of the poll the value with the highest score. This was explained with detail in section 4.5.

A.4. Polling of new values to an entity's property

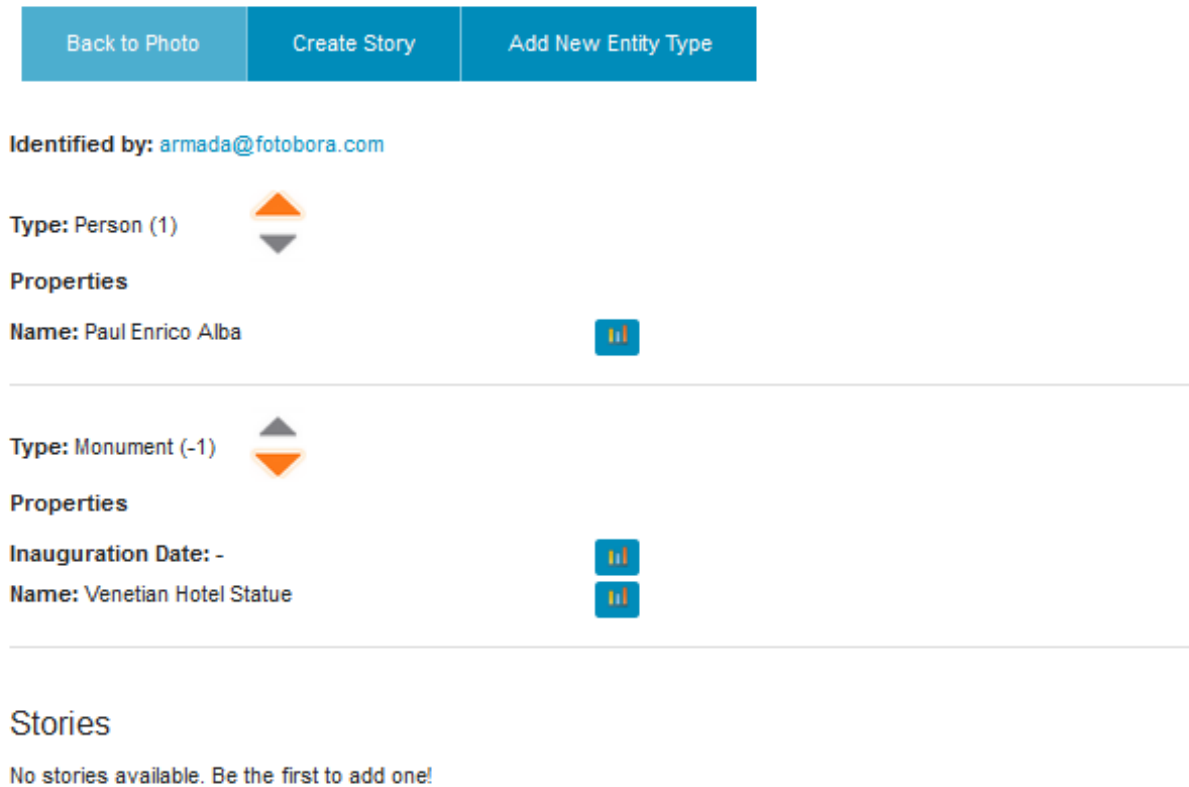


Figure 52.: Upvote or Downvote on Poll

Figures 53 to 56 show an example of the flow of actions to collaborate in the poll of values. In this case another user is attempting at adding a new value.


Once the value has been successfully submitted, the system will return a pop up message and display the results in the poll with an initial score that comes from the first level of certainty associated with that new value (Figure 55).

If another user votes on the new inserted value and that surpasses the previous highest score the system sort again the scores and redisplay the property values according to the new order (Figure 56).


At any point a user may change his mind regarding his vote and may vote on a different value or assign a new level of certainty. The system will calculate the new assigned values and reorder the poll depending on what has been changed.

A.5. Contribute with stories and comments

Identified by: armada@fotobora.com

Type: Person (1) 

Properties

Name: Paul Enrico Alba 

Poll

Paul Enrico Alba (4)

Certainty None Absolute


[Vote or Change Vote](#)

New Name


Value


Certainty None Absolute

[Add Value](#)

Type: Monument (-1) 

Properties

Inauguration Date: - 

Name: Venetian Hotel Statue 

Stories

No stories available. Be the first to add one!

Figure 53.: Voting Page

A.5 CONTRIBUTING WITH STORIES AND COMMENTS

From the main photograph page, if a user navigates to the "Create Story" page he will be prompted to create a story, and a form will be available to help him add it to the picture. (Figure 57)


Once the story has been submitted the user will be redirected to the main page of the photograph where the story will be displayed (Figure ??). A hyper-link will be added to a summary of the story in which it will navigate to the story page, allowing other users to publish comments regarding that story. A user may also edit the existing story by navigating to "Edit" on the corresponding story page (Figure 59).

Furthermore every entity can also be associated with a story. The entity story can be ac-


A.5. Contribute with stories and comments

[Back to Photo](#) [Create Story](#) [Add New Entity Type](#)

Identified by: armada@fotobora.com

Type: Person (1) 

Properties

Name: Paul Enrico Alba 

Poll

Paul Enrico Alba (4)

Certainty None Absolute


[Vote or Change Vote](#)

New Name


Value


Certainty None Absolute

[Add Value](#)

Type: Monument (-1) 

Properties

Inauguration Date: - 

Name: Venetian Hotel Statue 

Stories

No stories available. Be the first to add one!


Figure 54.: Voting Page

cessed via the "Create Story" page, in which Fotobora allows any user to add a new story to the entity or share any comment on it.


A.6. Exporting the information identified in the Photograph

Back to PhotoCreate StoryAdd New Entity Type

Identified by: armada@fotobora.com

Type: Person (1) 

Properties

Name: Paul Enrico Alba 

Poll

Paul Enrico Alba (4)


Jay Pierre Cuevas Elma (3)

Certainty None Absolute

New Name

Value

Certainty None Absolute

Type: Monument (-1) 

Properties

Figure 55.: New Value submission

A.6 EXPORTING THE INFORMATION IDENTIFIED IN THE PHOTOGRAPH


At any moment, a user can decide to, export the information that has been accumulated over time in the photograph into a readable PDF format file.

To illustrate the result of selecting the "Save to PDF" option, the PDF report generated, containing all the annotations associated to the photograph, is represented in Figures 60 to 62.


A.6. Exporting the information identified in the Photograph

Back to Photo Create Story Add New Entity Type

Identified by: armada@fotobora.com

Type: Person (1) 

Properties

Name: Jay Pierre Cuevas Elma 

Poll

Jay Pierre Cuevas Elma (8) ✓

Paul Enrico Alba (4)

Certainty None Absolute

[Vote or Change Vote](#)

New Name

Value

Certainty None Absolute

[Add Value](#)

Figure 56.: Example of Poll Reordering

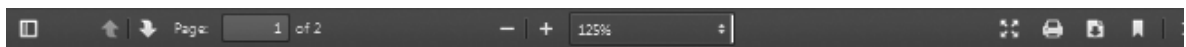


Photo Details



Submitted by armada@fotobora.com on 27 Jan 14:33

Address: armada@fotobora.com

89

Date taken: 27 Jan 00:00

Number of Stories: 1

Figure 60.: Example of the Convert to PDF Functionality

A.6. Exporting the information identified in the Photograph



Title

Venetian

Text

The Venetian has another unusual attraction: the "living statue." Every day, a young man get dressed and made up to look like a statue, and then he stands motionless for long stretches of time along the Grand Canal pretending to be one -- as hordes of mostly Chinese tourists gawk and poke and take photos.

Back

Create Story

Figure 57.: Story Page

by armada@fotobora.com

Address: Venice

Photographed on: 27 Jan 00:00

Uploaded on: 27 Jan 14:33

Entities:

(Person) [Paul Enrico Alba](#)

Stories

[Venetian](#) by armada@fotobora.com 28 Jan 01:23 0 comments *"The Venetian has another unusual attraction: the *..."*

Figure 58.: Story Page

A.6. Exporting the information identified in the Photograph



[Back to Entity](#)

[Edit](#)

Impersonation in Macau

Submitted on 28 Jan 01:18 by armada@fotobora.com

There is actually a small group of performers who embody the human statue. One of these "statues" is Paul Enrico Alba. Trained as a dancer in the Philippines, he was intrigued by the idea of "living statues" and studied other similar performances on YouTube, incorporating this style into his dancing moves. Five years ago, he was recruited by the Venetian and moved to Macau.

Comments

armada@fotobora.com on 28 Jan 01:20:

I saw him when I was there, he does an amazing performance!

Add a comment

[Submit Comment](#)

Figure 59.: Story Page

Entities (Detailed)

(#12) (Person) Paul Enrico Alba

by armada@fotobora.com on 27 Jan 14:34

Type: Entitytype

Properties:

Type	Value	Current Score	Initial Score
Name	Paul Enrico Alba	5	4
Name	Jay Pierre Cuevas Elrna	3	3
Name	Venetian Hotel Statue	3	3

Stories:

Impersonation In Macau

by armada@fotobora.com on 28 Jan 01:18

There is actually a small group of performers who embody the human statue. One of these "statues" is Paul Enrico Alba. Trained as a dancer in the Philippines, he was intrigued by the idea of "living statues" and studied other similar performances on YouTube, incorporating this style into his dancing moves. Five years ago, he was recruited by the Venetian and moved to Macau.

Comments:

[28 Jan 01:20] armada@fotobora.com: I saw him when I was there, he does an amazing performance.

A.6. Exporting the information identified in the Photograph

Entities (Detailed)

(#12) (Person) Paul Enrico Alba

by arnada@fotobora.com on 27 Jan 14:34

Type: Entitytype

Properties:

Type	Value	Current Score	Initial Score
Name	Paul Enrico Alba	5	4
Name	Jay Pierre Cuevas Elma	3	3
Name	Venetian Hotel Statue	3	3

Stories:

Impersonation In Macau

by arnada@fotobora.com on 28 Jan 01:18

There is actually a small group of performers who embody the human statue. One of these "statues" is Paul Enrico Alba. Trained as a dancer in the Philippines, he was intrigued by the idea of "living statues" and studied other similar performances on YouTube, incorporating this style into his dancing moves. Five years ago, he was recruited by the Venetian and moved to Macau.

Comments:

[28 Jan 01:20] arnada@fotobora.com: I saw him when I was there, he does an amazing performance.

Figure 62.: Example of the Convert to PDF Functionality

A.7. Associate different ontologies

A.7 ASSOCIATE DIFFERENT ONTOLOGIES

The system supports an ontology like structure, to support the categorization of each photograph entity. The ontology provides with set of entity types and the respective properties.

As said in chapter 3.4.1 the ontology predefined was extracted from DBPedia. Only a user with administrative privileges can maintain or change the content of the original ontology.

There are several methods and approaches that can be used to extract the information from the ontology in order to be injected in Fotobora's database. One of the possibilities is to assemble it manually using the framework provided by the ruby on rails *seed.rb* file. Listing A.1 illustrates the structure of a simplified ontology in that file.

```
Entitytype.create( { id: 1, name: 'Person', propertytype_id: 1 } )
Entitytype.create( { id: 2, name: 'Vehicle', propertytype_id: 3 } )
Entitytype.create( { id: 3, name: 'Place', propertytype_id: 6 } )
Entitytype.create( { id: 4, name: 'Pet', propertytype_id: 9 } )
Entitytype.create( { id: 5, name: 'Object', propertytype_id: 11 } )
Entitytype.create( { id: 6, name: 'Monument', propertytype_id: 12 } )
Entitytype.create( { id: 8, name: 'Animal', propertytype_id: 15,
  entitytype_id: 7 } )
Entitytype.create( { id: 9, name: 'Wild Animal', propertytype_id: 16,
  entitytype_id: 8 } )
Entitytype.create( { id: 10, name: 'Domestic Animal', propertytype_id: 18,
  entitytype_id: 8 } )

Propertytype.create( { entitytype_id: 1, id: 1, name: 'Name' } )
Propertytype.create( { entitytype_id: 2, id: 2, name: 'Brand' } )
Propertytype.create( { entitytype_id: 2, id: 3, name: 'Model' } )
Propertytype.create( { entitytype_id: 2, id: 4, name: 'License Plate' } )
Propertytype.create( { entitytype_id: 2, id: 5, name: 'Owner' } )
Propertytype.create( { entitytype_id: 3, id: 6, name: 'Name' } )
Propertytype.create( { entitytype_id: 3, id: 7, name: 'Address' } )
Propertytype.create( { entitytype_id: 3, id: 8, name: 'Owner' } )
Propertytype.create( { entitytype_id: 4, id: 9, name: 'Name' } )
Propertytype.create( { entitytype_id: 4, id: 10, name: 'Owner' } )
Propertytype.create( { entitytype_id: 5, id: 11, name: 'Description' } )
Propertytype.create( { entitytype_id: 6, id: 12, name: 'Name' } )
Propertytype.create( { entitytype_id: 6, id: 13, name: 'Inauguration Date' } )

Propertytype.create( { entitytype_id: 8, id: 15, name: 'Species' } )
Propertytype.create( { entitytype_id: 9, id: 16, name: 'Habitat' } )
Propertytype.create( { entitytype_id: 10, id: 17, name: 'Owner' } )
Propertytype.create( { entitytype_id: 10, id: 18, name: 'Name' } )
```

Listing A.1: Fotobora's Ontology in a Ruby Structured Seed File

Another possibility is to add new information automatically using SPARQL queries.

A.8. Import Annotations

A.8 IMPORT ANNOTATIONS

The user is able to import annotations by clicking the "Import Annotations" button within the context of each photograph. This will redirect the user to a Import Annotations page in which the user can visualize the existing XML schema that allows him to import an XML file that contains all the tagging information into the photograph (Figure 63) This will not replace any of the existing tags, it will only add additional ones.

The system will also display all the validation errors found on the page itself, allowing the user to restructures the file until it passes through every validation module (Figure 64).

Import Anotations



Here you can upload an XML file containing annotations that describe the picture above.

The XML file is validated against [this XML schema](#). Any validations errors that might occur will be displayed here.

The schema was last updated on 2016-01-07 15:02:43 -0500.

Please choose a file:

No file selected.

[Back to Photo](#)

[Import anotations](#)

Figure 63.: Import Annotations Page

A.9. Support Genealogical Trees

Import Annotations



Here you can upload an XML file containing annotations that describe the picture above.

The XML file is validated against [this XML schema](#). Any validation errors that might occur will be displayed here.

The schema was last updated on 2014-07-03 17:18:11 +0100.

Error(s) occurred:

- Invalid property type on line 4: Brand isn't a valid property for Pet entities, the valid properties are [Name, Owner]

Please choose a file:

Anotações Inválidas - Incoerência de Tipos.xml

Figure 64.: Import Annotations - Example of a Page Error

A.9 SUPPORT GENEALOGICAL TREES

To import a family tree the user must be in its main page context and click the button "import family tree". The user is prompted with some instructions on how to structure the file he wishes to submit (Figure 65).

A.9. Support Genealogical Trees

Fotobora armada@fotobora.com Edit registration Logout

Import Family Tree

Here you can upload a file containing a family tree in a format based on JSON. This family tree will replace your current family tree, if you already have one.

The file should be formatted according the following example:

```
{  "arvgen" : [    {      "name" : "João Silva",      "id" : 2,      "idfather" : 1,      "idmother" : null    },    {      "name" : "Luís Silva",      "id" : 1,      "idfather" : null,      "idmother" : null    }  ] }
```

Use the form to submit the file. Any errors will be displayed here.

Please choose a file:

No file selected.

Figure 65.: Family Tree Page

Any possible errors will be displayed on the box below the JSON example (Figure 66). Figure 67 displays the graphical result produced by Fotobora after adding a family tree to a photograph.

A.9. Support Genealogical Trees

Import Family Tree

Here you can upload a file containing a family tree in a format based on JSON. This family tree will replace your current family tree, if you already have one.

The file should be formatted according the following example:

```
{  "arvgen" : [    {      "name" : "João Silva",      "id" : 2,      "idfather" : 1,      "idmother" : null    },    {      "name" : "Luis Silva",      "id" : 1,      "idfather" : null,      "idmother" : null    }  ]}
```

Use the form to submit the file. Any errors will be displayed here.

Error(s) occurred:

- Element #5 has an unmatched 'idmae' value (999)

Please choose a file:

ArvGen Inválida - Referências Inválidas.txt

Figure 66.: Family Tree Error Example

Photos



Family Tree

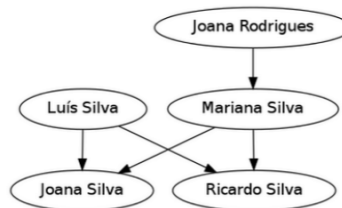


Figure 67.: Family Tree Example

This appendix was intended to explain all the features provided by Fotobora at the moment, using the system's screenshots to illustrate each one.