TÉCNICO
LISBOA

Optimization in Historical Digital Library Workflows

João Manuel Fernandes Cardoso

Thesis to obtain the Master of Science Degree in

Communication Networks Engineering

Examination Comittee
Chairperson: Prof. Luís Manuel Antunes Veiga
Supervisor: Prof. José Luis Brinquete Borbinha
Member of the comittee: Prof. André Ferreira Ferrão Couto e Vasconcelos

October 2013

Acknowledgments

I've always said that there are two kinds of leadership. You either lead by example, or by fear. For the past 25 years of my life I've been led by example by a person who has inspired me like no other, my father. As a father he taught me how to be a good man, as an engineer he has shown me what it is to be a professional. Whenever I felt like quitting I regarded his life as an example. If I get to be half the man he is I'll die a happy man. This is not to say that my mother hasn't played an equally important role. Undoubtedly she has. A kind hearted person who has faced terrible hardships in life. If anything, her life has taught me that true happiness lies not in ourselves but in the service of those we cherish. My sister has been equally important in my life. She constantly reminds me to enjoy life for what it is, and not for what it should be. The remainder of my extended family was a big support throughout these years. But I must mention my cousins Zé, Rosa, Henrique, Xico, Cátia, Teresa and Tomás and Xana for putting up with me every now and again. And of course a special thank you to my cousin Zé. A key element in the finalization of this thesis and a moral support throughout the whole course. Thank you man.

I have the pleasure to be called a friend by an excellent group of people. They're the family I got to select. Life wouldn't be the same without Rica, Rui, Tiago, Gil, Pedro, Inês, Filipe, Patrícia, Bárbara and André. Thank you guys.

Obviously I can not mention all of my colleagues here, so I thank them all equally, but notable mentions must be made to both Jackie and Hugo. The first for sharing some of her motivation when I was lacking belief, and the second for teaching me how to believe in myself. To the both of you, thank you.

During the development of this thesis I was mightily helped by both my supervisor Prof. José Borbinha, and my *de facto* co supervisor Gilberto Pedrosa. It was a pleasure working with you both, thank you.

Last but not least I must thank Sara, for she was a moral support to the end.

Abstract

Humankind have always felt the need to record its endeavours. Libraries are a natural consequence of that everlasting desire. They are but locations where information is stored, preserved and made accessible for those who seek it. With the advent of the information age, digitization of physical records became common practice, and thus the first "digital libraries" were born.

This report focuses on optimizing the workflows from an historical digital library Hemeroteca Municipal de Lisboa. It presents a study on the business processes at Hemeroteca's digitizing and image service, and the workflows they generate. Finally it describes the implementation of optimizations to those business processes and analyses the results from that implementation.

Keywords

Digital Library, Digitization, Metadata, Digital Publishing, Hemeroteca, Business Process, Workflow

Resumo

A humanidade sempre sentiu a necessidade de registar os seus feitos. As biblitecas são uma consequência natural desse desejo. São simplesmente locais onde a informação é armazenada, preservada e disponibilizada para todos os que a procuram. Com o advento da era da informação, a digitalização de registos fisicos tornou-se prática comum, e como tal foram criadas as primeiras "bibliotecas digitais".

Este relatório foca-se na optimização de fluxos de trabalho de uma biblioteca digital histórica, a Hemeroteca Municipal de Lisboa. Apresenta um estudo nos processos de negócio do seu serviço de digitalização e imagem, bem como dos fluxos de trabalho por estes gerados. Por fim descreve a implementação de optimizações nesses mesmos processos de negócio e analisa os resultados dessa implementação.

Palavras Chave

Biblioteca Digital, Digitalização, Metadados, Publicação Digital, Hemeroteca, Processo de Negócio, Fluxo de Trabalho

Index

List of Figures

List of Tables

List of acronyms

ANSI	American National Standards Institute
BLX	Bibliotecas Municipais de Lisboa
BND	Biblioteca Nacional Digital
BNP	Biblioteca Nacional de Portugal
CAN/MARC	Canadian MARC
CC	Creative Commons
CC REL	Creative Commons Rights Expression Language
CML	Câmara Municipal de Lisboa
CQL	Contextual Query Language
DASI	Divisão de Administração de Sistemas e Infrastructuras
DCMI	Dublin Core Metadata Initiative
DLF	Digital Library Federation
DOI	Digital Object Identifier
DMSI	Departamento de Modernização e Sistemas de Informação
Exif	Exchangeable Image File Format
GIF	Graphics Interchange Format
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol

IEC	International Electrotechnical Commission
IFLA	International Federation of Library Associations and Institutions
IT	Information Technology
ISO	International Organization for Standardization
JFIF	JPEG File Interchange Format
JPEG	Joint Photographic Experts Group
LOC	Library of Congress
MARC	Machine-Readable Cataloguing
MARCXML	MARC exchanged in XML
METS	Metadata Encoding and Transmission Standard
MIME	Multipurpose Internet Mail Extensions
MOA2	Making of America II
MODS	Metadata Object Description Schema
MPEG	Moving Picture Experts Group
MarcXchange	MARCXML generalization
NCSA	National Center for Supercomputing Applications
NISO	National Information Standards Organization
OAI-ORE	Open Archives Initiative Object Reuse and Exchange
OAI-PMH	Open Archives Initiative Protocol for Metadata Harvesting
OCLC	Online Computer Library Center
OPAC	Online Public Access Catalogue
P1	Process Digitizing and Publish
P1.1	Sub-Process Digitizing

P1.2	Sub-Process File Name Normalization
P1.3	Sub-Process Metadata Editing
P1.4	Sub-Process Publishing
P1.4.1	Sub-Process Collaboration File Creation
P1.4.2	Sub-Process Index Creation
P2	Process Metadata Sharing
PDF	Portable Document Format
PNG	Portable Network Graphics
PREMIS	Preservation Metadata: Implementation Strategies
PURL	Persistent Uniform Resource Locator
RDF	Resource Description Framework
RDF/XML	RDF expressed in XML syntax
ReM	Resource Map
RFC	Request for Comments
RLG	Research Libraries Group
RNOD	Registo Nacional de Objectos Nacionais
SOAP	Simple Object Access Protocol
SRU	Search/Retrieve via URL
SRW	Search/Retrieve Web Service
TEL	The European Library
TIFF	Tagged Image File Format
TXT	Text File
UNIMARC	Universal MARC

URI	Uniform Resource Identifier
URL	Uniform Resource Locator
URN	Uniform Resource Name
USMARC	United States MARC
W3C	World Wide Web Consortium
XHTML	Extensible HyperText Markup Language
XLS	Excel Binary File Format
XML	Extensible Mark-up Language

1

Introduction

Humankind have always felt the need to record its endeavours. Cave paintings were the first form of recording known to man, with the earliest records dating back to over 40,000 years. The first libraries were used to record the earliest form of writing: clay tablets in cuneiform script dating back to 2600 BC. Undoubtedly the most notorious of all the great libraries of the ancient world was the Royal Library of Alexandria in Egypt, around 300 BC.

The concept of library, a location where collections of records were preserved, stored and made accessible, didn't experience radical change until the arrival of the information age. Only then advances in computer science allowed for libraries to start working with digital representations of its physical records. Thus, the concept of "digital library" was born, having only appeared in the early 1990s with the advent of the Internet.

This report focuses on optimizing the workflows from an historical digital library, the Hemeroteca Municipal de Lisboa (Hemeroteca)[1]. This will imply studying the busi-

[1] Hemeroteca Municipal de Lisboa website: `http://hemerotecadigital.cm-lisboa.pt/`

ness processes at Hemeroteca's digitizing and image service, and the workflows they generate. Only fully understanding and optimizing the business processes can those optimizations be reflected in the workflows they generate.

Before even beginning to tackle such a subject, a characterization of the concepts that relate to digital libraries is required. Ranging from the concept of digital library itself, to how data is stored, enriched with metadata and finally shared with the general public.

The study of the business processes at the digitizing and image service at Hemeroteca was made by resorting to two distinct sources. Firstly, a general technical analysis of the domain was carried out, through the investigation of the main topics and related state of the art. Secondly, a pragmatic raising of requirements was carried out, based on the analysis of the actual case at the Hemeroteca and on interviews to the actors that are currently in charge and operating the business processes at the digitizing and image service. It was only by fully comprehending the actors needs and faults that any solutions aiming to optimize their business processes could be proposed and executed.

These solutions ranged from software applications which aimed to either optimize tasks which are manually performed or completely eliminate the human factor by automating them. However it is important to mention that not all solutions were self evident. On occasion constraints originated from the context of the problem, had an impact on the development and design of the solutions that are presented within this report. Additionally training was also provided for the existing staff members in order for them to improve their performance while performing tasks.

Overall this report will show that the objectives that were proposed during the project phase of this thesis were accomplished. Not only that, but their success led to the improvement of both the efficiency and effectiveness of a digital library, thus proving that masters thesis can and should have a real world impact.

1.1 Motivation and objectives

The generic objective of this work is to propose solutions to optimize workflows which result from business processes for digitization and publishing in an existing cultural heritage digital library, mostly comprised of newspapers and magazines whose intellectual property rights have expired, i.e., have fallen into public domain.

The concrete focus and motivation is as stated above the Hemeroteca Municipal de Lisboa, specifically their digitizing and image service. The optimizations to be proposed, implemented and validated must contribute so that the tasks that make up their business processes can be performed more efficiently (using less resources and time) and more effectively (improving the quality of the final product and providing in the end a better experience to the final users when interacting with the digital library).

1.2 Document Structure

This report is divided into six different chapters. Chapter 1 presents the introduction, motivation and objectives for this thesis report. Chapter 2 describes the required state of the art in digital libraries, which is required for this project. Througout chapter 3 an analysis on the problem that was tackled in this project is made, and the results of the assessment made on the Hemeroteca are detailed. Chapter 4 on the other hand describes the solutions and their designs to the issues related in chapter 3. Those solutions are then validated in chapter 5. Finally, chapter 6 finishes this report by drawing conclusions on the execution of this project, and proposes future changes to the solutions that might improve on what has been achieved.

2

State of the Art in Digital Libraries

This chapter provides a theoretical context to the concepts that are abridged throughout this report. It is organized into three sections. Section 2.1 focuses on the subject of digital libraries, and tries to provide a general overview on the subject, by describing generic business processes that existing digital libraries implement. It also characterizes the concept of authority control. Section 2.2 clarifies the concept of digitization, and provides a short summary of several file formats which are used within the realm of digital libraries. Section 2.3 takes on the subject of metadata, by categorizing the different metadata types, as well as providing examples for each of them. Section 2.4 is centered around the issue of digital publishing. Therefore it covers themes such as metadata harvesting protocols and digital identification.

2.1 Digital Libraries

This section focuses on the subject of digital libraries, and tries to provide a general overview on the subject. In order to do that, the several existing digital library types will be specified having, the care to contextualize each type with an example.

The concept of "digital library" is fairly recent, although a permanent definition is yet to be set. A broad definition could be that digital libraries consist on *"A managed collection of information, with associated services, where the information is stored in digital formats and accessible over a network"* [1].

Digital libraries cannot be clearly classified into distinct categories. It is however undeniable that most digital libraries share common organization types and overall objectives.

Depending on the objective of the digital library, it may have to implement up to seven generic business processes. Those generic business processes are:

- *Selection and Acquisition*: Content selection and digitization process;
- *Metadata Organization*: The assignment of metadata to the digitized objects;
- *Indexation and Storage*: Content and metadata indexation and storage;
- *Repository*: Indexed digital object records, which are associated with metadata. Can be combined with the Indexation and Storage component;
- *Search and Retrieval*: Availability to browse, search, retrieve and view the content stored in the repository;
- *Digital Library Website*: The digital library's front end. Can be combined with the Search and Retrieval component;
- *Network Connectivity*: Connect the digital library with the internet, thus making its content available to both internal and external users.

This project's work targeted primarily on three business processes. Which were *Metadata Organization*, *Indexation and Storage*, and the *Digital Library Website*. The reasons that led to the focus being set on these three components, are described in chapter 4.

Digital libraries try to emulate the traditional library format. The difference between digital and traditional libraries lies on the use of computerized systems that manage the digital library's fully digital repository in a centralized manner. The digital records are obtained through systematic digitization of original objects (primary sources). The United States Library of Congress (LOC) [1] is an example of a digital library, which not only possesses the original objects, but also provides access to its digital versions. The Biblioteca Nacional de Portugal (BNP) can also be considered an example of such a library, for it grants access to digitized versions of its original objects through the Biblioteca Nacional Digital (BND) [2].

Digital libraries which resort to using metadata harvesting protocols, such as the OAI-PMH (see section 2.4.4) can be compared to virtual libraries, because they are not content holders, but content providers. Metadata is harvested, so that each record allows for the direct retrieval of an actual item. The items themselves are scattered within many different servers over the network. Therefore, and since only metadata from that content is held, this causes the harvested metadata repositories to be small and compact. Both the Europeana.eu internet portal [3] and The European Library (TEL) [4] provide access to millions of digitized works, through the use of metadata harvesting protocols.

2.1.1 Authority Control

A key process in the realm of digital libraries is authority control, which is technical process for the organization of a digital library's catalogue and bibliographic information. It is based on the providence of information uniqueness, standardization and linkage.

As is described by Doris Hargrett Clark [2, p. 1], "Authority control of a library catalogue is maintained through an authority file that contains the terms used as access points in the catalogue. The access points that determine the structure of the catalogue may be real entry headings on bibliographic records or cross references. In library catalogues the entry headings under control generally consist of personal and corporate names, uniform titles, series, and subjects".

Instating authority control in a digital library brings about 6 main benefits. They are:

- Quicker and more accurate searches;

[1] Library of Congress website: http://www.loc.gov Retrieved 2012-12-27

[2] Biblioteca Nacional Digital website: http://www.bnportugal.pt Retrieved 2012-12-27

[3] Europeana.eu website: http://www.europeana.eu Retrieved 2012-12-27

[4] The European Library website: http://www.theeuropeanlibrary.org Retrieved 2012-12-27

- Record consistency;
- Cross referenceable structure;
- Cataloguing efficiency;
- Easier catalogue maintenance;
- Error minimization.

Authority control requires researches only a short query to focus on a specific subject. Thus improving the search accuracy and speed. By having a centralized control records are easily kept consistent, and can be cross referenciable. Cataloguing efficiency is promoted by allowing cataloguers to use authority records when trying to categorize new items. This way cataloguers can save time by analysing which records have already been registered in the catalogue. Logically if cataloguers have access to the records, it also means that they will be able to spot errors and inconsistencies easily. Software tools that do this task automatically can also be used. All this works to simplify a catalogue's maintenance and avoid errors.

The subject of authority control is relevant to this project, because an authority file was developed as part of the solution (see chapter 4) to the problems presented in chapter 3.

2.2 Digitization

The term "digitization" refers to the process of converting analogue information, (e.g., images, documents, sound, signals, etc.), into digital information. This section briefly introduces the concept of digital image, and defines several file formats that are used to store digital images, and are relevant to the project related in this report.

2.2.1 Digital Images

Digital libraries, as previously stated, can be interpreted as collections of information stored in a digital format. This is where the concept of digital image comes in play. This section briefly introduces the topics of discretization, quantization and compression. Which are key to understanding how a digital image is created.

The discretization process consists on sampling the infinite range of values present on an analogue signal in periodic time intervals. The quantization process on the other hand comprises on the division of the samples, derived from the discretization process, into non-overlapping subranges. To each subrange, a discrete value is assigned. The number of subranges varies according to the number of bits used to represent the sample. Finally in order to reduce the number of bits which are necessary to represent a digital image, there is the need to compress it. Compression aims to reduce both the redundant and irrelevant information stored within an image.

2.2.2 File Formats

After digitization, the resulting digital images must be stored. There are several file formats which are used to serve this purpose. Bearing in mind that within the digital library realm, text is the main content represented in digital images, the most suited file formats are the TIFF and PDF. Both of which are described in this subsection. However the JPEG file format is also relevant to this particular project, as will be shown in section 4.

2.2.2.A Tagged Image File Format (TIFF)

The Tagged Image File Format (TIFF) is, as the name alludes, a tag-based file format used for storing and interchanging bitmap images [3]. TIFF was first developed by Aldus in 1986, as an attempt to develop a common scanned image file format. Aldus has since been acquired by Adobe Systems, which is the current holder of the TIFF specification copyright.

TIFF is used to handle bitmap images and data within a single file. To accomplish this, it uses header tags which help define the proportions of the images. In a way TIFF acts as a container for different bitstream encodings for bitmap images. For example, TIFF may be used to hold Joint Photographic Experts Group (JPEG) images, by using a lossy compression scheme.

On the other range of the spectrum, TIFF also supports the use of lossless compression schemes, or no compression schemes at all. Thus making it ideal to store images which may be edited, for it prevents quality loss.

Additionally TIFF is also suited for images with a very high spatial resolution, as well as being flexible in terms of image colour depth (e.g. it allows for both 8 bit and 24 bit colour).

2.2.2.B Portable Document Format (PDF)

The Portable Document Format (PDF) is a formatting language, which was developed by Adobe Systems in 1993. Currently Adobe Systems still holds patents to PDF, but it is maintained as an International Organization for Standardization (ISO) standard [4].

A PDF document structure is composed of four code components [5]:

- Header;
- Body;
- Cross-Reference table;
- Trailer.

The Header has information to identify the PDF version.

The Body contains object information. There are eight different types of objects that can be contained within a PDF document. They are: boolean values, numbers, strings, names, arrays, dictionaries streams, and the null object. Objects can be classified as direct, i.e., embedded in another object, or indirect. The later meaning that they must be numbered with both an object and a generation number.

The Cross-Reference table has pointers to the byte offset of each object within the Body. This feature that makes efficient random access to object within PDF documents possible.

Finally the Trailer contains pointers to the Cross-Reference table, and other key objects which are contained within the Trailer itself.

2.2.2.C JPEG and JPEG2000

The JPEG standard was named after the Joint Photographic Experts Group, which was responsible for its development [6]. This standards compresses digital images with a lossy compression method.

Due to the fact that the JPEG comittee did not specify a file format for the JPEG standard, several formats have been developed. The most commonly used are the

JPEG File Interchange Format (JFIF) [7], and the Exchangeable Image File Format (Exif) [8].

The JFIF allows for JPEG encoded bitsteams to be exchanged between various applications and platforms. It has three main features: The allowance for multiple components, (i.e., the luminance and chrominance signals), to have different resolutions; The provisioning of a resolution or aspect ratio coding method, which is absent in the JPEG standard; And finally the definition of a colour model for images. As for the Exif standard, it specifies formats for images, sound, and tags in digital still cameras and in other systems handling image and sound files recorded by digital still cameras [8, p. 5].

The JPEG 2000 image compression standard [9] was designed by the JPEG committee in 2000 with the intention of replacing the original JPEG standard. JPEG 2000 achieves a moderate increase in compression performance, when compared to the JPEG standard. However it allows greater flexibility by allowing the choice between the two different compression types, (i.e., lossless or lossy). The ability to store different parts of a same image with different quality, by randomly accessing the code stream, also proves to be a major advantage over the traditional JPEG standard.

2.3 Metadata

There isn't a clear definition of the term metadata within the realm of digital libraries. The dictionary definition of metadata states that it is "data about data" [10]. But clearly this definition does not apply to structural metadata, for it characterizes the structures in which data is organized rather than the data itself.

A broader definition can be attributed to the National Information Standards Organization (NISO), which states that *"metadata is structured information that describes, explains, locates, or otherwise makes it easier to retrieve, use, or manage an information source."* [11, p. 1].

The use of metadata as a way to systematically catalogue objects, has long been common policy in libraries throughout the world. The advent of the digital era only enforced the already growing demand for metadata standardization. NISO divided metadata into three separate types: [11, p. 1]

- Descriptive metadata;
- Structural metadata;

- Administrative metadata.

Whenever metadata elements are grouped into sets to tackle a specific purpose, such as describing particular types of information resources or specific domains, these sets are called metadata schemes.

Metadata schemes enforce a semantic upon its metadata element sets, i.e., each metadata element sees its name and meaning specified. Additionally content formulation, representation and allowable values may also be regulated.

Most metadata schemes do not specify syntax rules, (i.e., how the metadata elements content is to be encoded), but specific syntaxes may be prescribed (e.g., METS uses XML, see section 2.3.2.A). The Extensible Mark-up Language (XML) [12] is a syntax which for its flexibility in the handling of metadata records, and it allows for greater interoperability with different schemas.

There are many medatada schemes. They vary both on their user environment and focus discipline. The following subsections present the most relevant metadata schemes regarding the area of study of this report.

2.3.1 Descriptive Metadata

Descriptive metadata refers to information used for searching and locating objects. (e.g., the object's author, title, etc.)

2.3.1.A Dublin Core

Dublin Core was first introduced at the 1995 OCLC/NCSA Metadata Workshop [13] , which was held in Dublin, Ohio and organized by the Online Computer Library Center (OCLC) and the National Center for Supercomputing Applications (NCSA). Currently Dublin Core is maintaned by the Dublin Core Metadata Initiative (DCMI) [5], an independent entity which separated from the OCLC in 2008.

Dublin Core was conceived having in mind authors unfamiliarized with cataloguing. Therefore it has 15 different elements,(title, creator, subject, descriptions, publisher, contributor, date, type, format, identifier, source, language, relation, coverage and rights), that share a common semantics and act as core set of description elements which are used by authors to generate descriptions for their resources with relative

[5] DCMI website: `http://dublincore.org` Retrieved 2013-01-02

ease. The existence of these elements promotes interoperability, in the sense that it offers a set of common core descriptors that unify related attributes, thus facilitating cross disciplinary searches.

There is an ongoing discussion between having a minimal number of elements and a simpler syntax, and being able to cover an ever growing number of disciplines through finer semantics and a higher number of elements, (specialized elements). This has led to the clear distinction between the previously described classical Dublin Core,(now called Simple Dublin Core), and its extension the Qualified Dublin Core. Apart from these element refinements (e.g., specialized elements) Qualified Dublin Core also features a new set of encoding schemes which are used for element value interpretation.

Albeit its initial focus on author generated resource description, Dublin Core has surpassed that role by achieving acceptance within formal resource description communities. Stuart Weibel attests to Dublin Core's flexibility by stating that *"The Dublin Core, in the hands of cataloguing experts, is expected to provide an economical alternative to more elaborate description models such as full Machine-Readable Cataloguing (MARC) cataloguing"* [14, p. 10].

According to the DCMI, Dublin Core metadata can be expressed through four distinct syntaxes. The syntaxes are:

- *DC-Text:* Using the Dublin Core text format;
- *DC-HTML:* Using Hypertext Markup Language (HTML) and Extensible HyperText Markup Language (XHTML) meta and link elements;
- *DC-RDF:* Using the RDF;
- *DC-DS-XML:* Using XML to express the Dublin Core description sets.

All of these syntaxes are used to describe Dublin Core metadata description sets. Description sets are documents that define and describe the metadata used to meet specific application needs, while providing semantic interoperability with other applications, through the use of common vocabularies and models.

The "DC-Text" format uses plain text to represent a Dublin Core metadata description set. The "DC-HTML" meta data profile, is used in order to provide a specific set of conventions by which a Dublin Core metadata description set can be represented within an HTML/XHTML web page. The "DC-RDF" provides recommendations for expressing Dublin Core metadata using Resource Description Framework (RDF) [6]. It may

[6] Resource Description Framework website: `http://www.w3.org/RDF/` Retrieved 2013-01-02

be used in association with XML. This association is dubbed RDF/XML, which is a normative syntax defined by the World Wide Web Consortium (W3C) [7] to express RDF graphs as XML documents. Finally the "DC-DS-XML" format uses XML to represent a Dublin Core metadata description set using a syntax appropriate for XML documents.

2.3.1.B Machine-Readable Cataloguing (MARC)

The Machine-Readable Cataloguing (MARC) is a standard digital format for bibliographical item description. It was first developed by the United States LOC during the 1960s in order to provide an easy method of creating an distributing digital library records. By 1973 it had already been set as an international standard.

Initially MARC only used the record structure described in ISO 2709 [15]. In ISO 2709 each record consists in three different sections, which are separated by a special record separator character. The three sections are:

- Record label;
- Directory;
- Datafields.

The Record label consists on the first 24 characters of the record. It holds both the record length and its data base address. It also holds data regarding the number of characters used for both indicators and subfield indicators.

The Directory holds the start positions for all the fields within the record and also its respective tags. Each directory entry consists on 4 parts, and totals a maximum of 9 characters in length. The 4 parts are the field tag, the field length, its starting position and an optional implementation defined part. Fields can themselves be sorted into three distinct categories. The *record identifier field* which identifies the record and its assigned by the organization which created the record; the reserved fields which support data that might be needed for record processing; and finally the bibliographical fields.

The Datafields consist in a single string which holds all the different fields and subfields which compose the record.

[7] RDF/XML Syntax Specification: `http://www.w3.org/TR/rdf-syntax-grammar/` Retrieved 2013-01-02

A – MARC Versions There are a great multitude of MARC versions. Universal MARC (UNIMARC) , which is still used in Portugal, and MARC 21 are two examples of MARC versions.

UNIMARC was first introduced by the International Federation of Library Associations and Institutions (IFLA) in 1977. It's currently in its third edition, which was published in 2008 [16].

Its objective was to create a MARC version which would allow for international record exchange, which had been a problem in MARC due to incompatibility between the various existing MARC national formats.

MARC 21 was created in 1999, as a redefinition or the MARC standard for the 21st century. It converges several MARC versions, (e.g. United States MARC (USMARC), the Canadian MARC (CAN/MARC) and the European UNIMARC),into a single harmonized version with the intent of providing a greater format integration. It has five communication formats used for representing and exchanging information between machines [17]. The formats are: MARC 21 format for bibliographic data, MARC 21 format for authority data, MARC 21 format for holdings data, MARC 21 format for classification data, and MARC 21 format for community information. Of all these formats the most relevant in light of the scope of this report is its format for bibliographical data.

B – MARC and XML Satisfying the necessity to provide means to exchange MARC records in XML lead to the introduction of the MARCXML schema [18] by the United States LOC in 2001. This new schema was in its essence framework to work with MARC 21 records in an XML environment.

MARCXML simply reflects the ISO 2709 structure, adapting it in order to be used within an XML schema. A relevant feature is the providence of lossless round-trip conversion of the ISO 2709 MARC 21 records.

In 2006 a new standard was introduced, the MarcXchange standard [19]. Its goal was to generalize the MARCXML schema so that all existing formats based on ISO 2709 syntax could be represented. In order to fulfil this goal, MarcXchange reuses the original ISO 2709 elements whilst introducing links to its terminology. The *record label* element from ISO 2709 is referred to as *leader* whilst the *record identifier field* and *reference field* from ISO 2709 are incorporated in MarcXchange as *controlfield*. Additionally the need to specify a record content has lead to the introduction of the *format* and *type* attributes, which respectively specify both the MARC format, and the kind of record.

2.3.1.C Metadata Object Description Schema (MODS)

The Metadata Object Description Schema (MODS) is, as the name states, a descriptive metadata scheme developed by the United States LOC Network Development and the MARC Standards Office in 2002 [20]. Its main objective is to provide a bibliographical element set for library applications, but it may also be used for generic purposes.

MODS can be used to create original resource description records. Moreover, since it is a derivative of MARC 21, MODS is also able to carry information from existing MARC 21 records. Nonetheless not all MARC fields are defined in MODS , and both the field and subfield tags are not used. There is also a lack of full compatibility in terms of data elements, thus the round-trip conversion between MODS and MARC is lossy.

One of MODS distinguishing features is its use of XML. This opens a wide range of possible uses such as being used as an extension to METS (see section 2.3.2.A), as a metadata set for harvesting (see section 2.4.4), or simply to create new original resource description records in an XML syntax.

In MODS an XML document may contain a wide list of elements [8], and their related attributes. There is a division between "Top Level" and "Root" elements. "Root" elements are mandatory in every XML document, whilst "Top Level" elements are optional, however at least one must be present in every XML document.

Furthermore MODS has several key advantages. Its element set is richer than Dublin Core, yet it is simpler than full MARC. Additionally its element set has a higher compatibility with existing descriptions in large library databases, than standards such as Dublin Core [11, p. 6]. The use of language based tags instead of the numeric tags used in MARC make MODS a more user-friendly schema.

2.3.2 Structural Metadata

Structural metadata refers to the metadata which specifies how records should be organized to represent complex object structures, (e.g. how digitized articles are ordered make up the original digitized magazine).

Additional technical metadata is necessary to record information regarding the dig-

[8] Outline of Elements and Attributes in MODS Version 3.4 (2011-09-22): `http://www.loc.gov/standards/mods/mods-outline.html` Retrieved 2013-01-02

itization process. This technical metadata is relevant for researchers, for it helps to ensure that the digital version is an accurate reproduction of the original object. Libraries on the hand need technical metadata in order to periodically refresh and migrate data, thus providing the necessary maintenance for its digital records.

2.3.2.A Metadata Encoding and Transmission Standard (METS)

The Metadata Encoding and Transmission Standard (METS) has its origins within the Making of America II (MOA2). A digitization project which took on the issue of structural metadata, and created a encoding format for administrative and structural metadata for textual and image-based digitization works. The Digital Library Federation (DLF) used MOA2 as a stepping stone for the creation of METS [21].

METS is a schema which allows for the encoding of the three NISO defined metadata types, (i.e., administrative, descriptive and structural metadata), for objects within a digital library. It uses XML to create documents which represent the hierarchical structure of the original object. Additional technical metadata regarding the names of the objects and comprising files locations is also associated to the document that METS creates.

All METS XML documents have seven main sections, which are laid out in the following order:

1. METS Header;
2. Descriptive Metadata;
3. Administrative Metadata;
4. File Section;
5. Structural Map;
6. Structural Links;
7. Behaviour.

The METS Header section contains metadata which describe the METS document itself, (e.g., creator, editor, etc.).

The Descriptive Metadata section may point to descriptive metadata, which is external to the METS document, (e.g., a MARC record kept in a library catalogue). It may also hold internally embedded descriptive metadata, or both previous cases can be true. Regardless, multiple instances of both external or internally kept descriptive metadata can be included in this section.

The Administrative Metadata section provides information as to how files were created and stored, their intellectual property rights, metadata regarding the original object which was subjected to digitization to create a digital library object and finally information describing the provenance of said comprising files of a digital library object.

The File Section is used to list all files which contain versions of the digital library object.

The Structural Map section profiles an hierarchical structure for the digital library object. It also links the elements that comprise the structure, to files which hold both content and metadata regarding each of those elements.

The Structural Link section allows for METS users to register the existence of links between nodes within the hierarchical structure profiled in the structural map section.

The Behaviour section is used to associate executable behaviours to the METS object content.

2.3.2.B MPEG-21

The MPEG-21 standard, is an open framework for multimedia applications, which aims to ensure interoperability of digital multimedia objects. The standard is developed by the ISO/IEC Moving Picture Experts Group (MPEG), and it is ratified in the ISO/IEC 21000 - Multimedia framework (MPEG-21) standards [22].

This standard its built around two fundamental concepts: The Digital Item, which is the standard's unit for content distribution and transaction (e.g., video collections, music albums). And the user interaction with Digital Items.

Therefore the objective of MPEG-21 is to allow for users to interact with each other, using the MPEG-21 framework. This way they're able to preform a series of actions on Digital Items in an efficient, transparent and interoperable way.

2.3.3 Administrative Metadata

Administrative metadata provides information to manage objects, such as its origins (e.g., creator, date of creation, etc.) and its access regulations. Administrative metadata can itself be divided into two separate metadata types: *Rights management* *metadata* which refers to intellectual property rights such as access rights and restrictions and preservation rights and restrictions; And *preservation metadata* which refers to resource management information such as the files technical features.

This section introduces two standards: Preservation Metadata: Implementation Strategies (PREMIS) and Creative Commons (CC). These, respectively illustrate the concepts of *preservation metadata* and *rights management metadata*.

2.3.3.A Preservation Metadata: Implementation Strategies (PREMIS)

The Preservation Metadata: Implementation Strategies (PREMIS) is a working group established in 2003 by the OCLC and the Research Libraries Group (RLG). It comprises of international experts in the use of metadata to support digital preservation activities. Its original objective was to develop a core set of implementable preservation metadata, broadly applicable across a wide range of digital preservation contexts and supported by guidelines and recommendations for creation, management, and use [23, p. 1].

PREMIS' data model is based on the existence of five entities, according to its Data Dictionary for Preservation Metadata [23, p. 6]. The entities are:

- Intellectual Entity;
- Object;
- Rights;
- Agent;
- Event.

Each one of these entities has semantic units mapped to themselves. These semantic units can be viewed as properties for entities, and have values associated to them (e.g., *size* is a object entity property, which has a value associated to it).

The Intellectual Entity comprises on a set of content which is regarded as a single intellectual unit for management and description purposes, (e.g., books, albums, etc.). A single Intellectual Entity may contain other Intellectual Entities, and may have several digital representations, (e.g., books have pages and pages can have figures).

Objects are in turn regarded as common digital objects, i.e., discrete units of information in a digital form. Objects have three subtypes: File which is a named and ordered sequence of bytes which contains information for preservation purposes; bitstream which is a contiguous or non-contiguous set of data within a file; and representation which is a set of files with the necessary structural metadata, that allows them to form a complete Intellectual Entity.

Rights are assertions of one or more intellectual property rights or permissions regarding an Object and/or an Agent. In PREMIS a preservation repository is allowed to determine whether a certain action can or cannot be preformed, based on documentation regarding the assertions that preforming the action would question.

Agent entities are people, organizations or software, which are associated with Events during an Object or its associated Rights lifetime

The Event entity gathers metadata on actions (e.g., creating a new version of a digital object). These actions affect both Objects or Agents, and are known or associated to a preservation repository. The preservation repository's decision on which Events to record, can based on their relevance or it can also be defined during implementation.

PREMIS metadata is stores through the use of XML documents. The "Root" elements of a PREMIS XML schema [9] mimic the entities described in the data model.

2.3.3.B Creative Commons (CC)

Creative Commons (CC) [10] is a non-profit organization founded in 2001.

As a standard for digital rights management, CC released various copyright-licences known as Creative Commons licences. These allow for the distribution of copyrighted works and are distributed free of charge.

CC has defined three "layers" for each of its licences [11]. The layers are:

- A legal code layer;
- A commons deed layer;
- A "machine readable" layer.

[9] PREMIS Schema 2.2 (2012-05-15): `http://www.loc.gov/standards/premis/v2/premis.xsd` Retrieved 2013-01-02

[10] Creative Commons website: `http://creativecommons.org/` Retrieved 2013-01-02

[11] Creative Commons licences: `http://creativecommons.org/licenses/?lang=en` Retrieved 2013-01-02

The legal code layer is expressed as a traditional legal tool. The commons deed is a summary of the most important terms and conditions expressed within the licence. The commons deed layer is aimed at laymen. Finally the "machine readable" version is expressed through the use of Creative Commons Rights Expression Language (CC REL) [24], a specification describing how license information may be described using RDF and how license information may be attached to works.

2.4 Digital Publishing

This section focuses on digital publishing. For this purpose the subjects of digital identification, the techniques to provide access to information objects, search and retrieve protocols and metadata harvesting protocols will be analysed in sections 2.4.1, 2.4.2 2.4.3 and 2.4.4, respectively.

2.4.1 Digital Identification

Metadata schemes tend to provide ways to identify the objects which the metadata is referring. In general terms there are three main types of identification methods. An objects location can be given by its file name (e.g., Uniform Resource Identifier (URI)), it can have a persistent Uniform Resource Locator (URL) which does not change over time (e.g., Persistent Uniform Resource Locator (PURL)), or it may have a Digital Object Identifier (DOI).

Structured data can also be published with the objective of establishing links between information sources, as can be seen in the Linked Data method.

2.4.1.A Uniform Resource Identifier (URI)

A Uniform Resource Identifier (URI) is a compact sequence of characters which identify a resource. This is done through the use of a syntax which is specified in Request for Comments (RFC) 3986 [25, p. 15]. URIs can be classified as names (Uniform Resource Name (URN)), locators (URL) or as both. Each URI type is defined through a specific syntax, and its associated protocols.

URNs are used for identifying resources, and use a specific syntax which was defined in RFC 2141 [26, p. 1]. On the other hand URLs are meant to locate or find resources. The URL specific syntax can be found in RFC 1738 [27].

2.4.1.B Persistent Uniform Resource Locator (PURL)

A Persistent Uniform Resource Locator (PURL) is a URL, (i.e., a location-based URI as seen in the previous section), that redirects to the URL of the requested resource. The concept was first introduced in 1995 by the OCLC, which currently manages a PURL resolver [12].

PURLs provide permanent identifiers for resources, this allows the common user always use the same address to find the resource, in spite of changes to its real address. This level of indirection solves the issue of transitory URIs in location-based URI schemes (e.g. Hypertext Transfer Protocol (HTTP)).

2.4.1.C Digital Object Identifier (DOI)

URIs when combined with PURL, provide mechanisms to publish resources and making them widely available. These mechanisms however powerful are not enough to manage intellectual property. The Digital Object Identifier (DOI), which was launched in October 1997 following a prototyping phase [28], takes on the sensitive issue of publishing such content.

According to Norman Paskin *"The DOI is a unique identifier of any piece of intellectual content (in any form), together with a system for using that identifier to locate digital services (on the internet) associated with that content."* [29, p. 14]. This means that the DOI remains unchanged regardless of its object ownership and location, which is a similar concept to the already mentioned PURL. However DOI takes a completely different approach to syntax than the one previously seen in PURL. DOI takes advantage of the syntax defined for URNs and creates a namespace for DOIs.

The DOI system is developed and promoted under the International DOI Foundation. The DOI initiative is a system which includes a resolution mechanism, a store of metadata (regarding the identified object), an administrative agency that manages the business side of the identification and management process, and an authority which controls the DOI namespace and defines policies [29, p. 15].

[12] Online Computer Library Center PURL resolver: `http://purl.org/docs/index.html` Retrieved 2012-11-20

2.4.1.D Linked Data

Linked Data is a method of publishing structured data in way that allows for its intrinsic value to be increased by way of data interlinking. This concept builds upon standard web technologies, that use hypertext (e.g., HTTP or URIs), by using them to share data that can be automatically read by machines. Thus allowing for the establishment of links between different information sources, that in turn enable data querying.

The term Linked Data was first used by Tim Berners-Lee, which is the acting director of the W3C[13]. Berners-Lee states four guide rules for the implementation of a Linked Data system in his 2009 article about the design issues brought about by Linked Data [30]. The guide rules are:

1. URI usage, for resource identification;
2. HTTP URI usage, for look-up purposes;
3. Providence of useful resource information when a look-up is preformed, by using standard formats such as RDF/XML;
4. Inclusion of links to other related URIs, in order to improve the discovery of related information on the web.

2.4.2 Access to the Information Objects

After being created through digitization, having been enriched with metadata addition and subsequently been provided with a digital identifier, a digital object must be made accessible. Currently three distinct techniques can be conceptually considered as an answer to this issue. They are:

- The reproduction of the original object through the use of a file format;
- The use of HTML pages;
- The use of software to emulate interaction with real objects.

The first technique consists on distributing the digitized object using a file format wich represents the original object with a high degree of fidelity (e.g., using the TIFF file format to represent maps, or representing monographs resorting to PDFs).

[13] World Wide Web Consortium website: `http://www.w3.org/` Retrieved 2013-01-04

The second technique which can be conceptually considered, is the representation of the original objects, and if needed their structures, through the use of an HTML page structure[14]. This technique is best exemplified using the book metaphor. In this scenario an index HTML page displays links to other HTML pages, each representing a book page.

The third and final technique consists on using software to emulate the experience of handling real objects. An example of such software are "page turning" applications[15], through which users can read documents simulating the experience of turning pages.

2.4.3 Search and Retrieve Protocols

Conceptually there are two distinct procedures to preform information search and retrieval. It can be preformed by Human users, through Online Public Access Catalogues (OPACs), or by automated systems through the use of the client-server Z39.50 protocol and its web service successors, the Search/Retrieve via URL (SRU) and Search/Retrieve Web Service (SRW).

The following sections offer a description on OPACs, and a characterization of the Z39.50, SRU and SRW protocols.

2.4.3.A Online Public Access Cataloguing (OPAC)

An Online Public Access Catalogue (OPAC)[16] is a database whose entries reflect the objects present within a digital library's catalogue, and which is made available online. A parallel can be made with search engines because OPACs use sophisticated search technologies, such as faceted search, relevance ranking and user reviews and tagging. However an OPAC, unlike a search engine, preforms its queries on the database to which is linked.

[14] Example of the use of an HTML page structure to distribute content by the BNP: `http://purl.pt/1/1/` Retrieved 2013-01-05

[15] Example of the use of a "page turning" application by the British Library: `http://www.bl.uk/onlinegallery/virtualbooks/index.html` Retrieved 2013-01-05

[16] Example of the Bibliotecas Municipais de Lisboa (BLX) OPAC: `http://catalogolx.cm-lisboa.pt/ipac20/ipac.jsp?profile=cmlht&menu=search` Retrieved 2013-01-05

2.4.3.B Z39.50

Z39.50 comprises on a information retrieval application service and its corresponding client-server protocol. It is maintained by the the United States LOC and it's an ANSI/NISO standard [31].

The Z39.50 information retrieval service depicts the interaction between a client application and a server application, which is connected at least one database. It is therefore possible for client applications to preform actions based on the information present at the databases connected to the server application (e.g., searches, browsing, retrieval, etc.).

Since database indexation models vary from system to system, Z39.50 has a generic database description model. This abstracts the database structure, allowing for the z39.50 syntax to become a common model for all database structures. This leaves the server application with the task of mapping each request to its system specific indexing model.

2.4.3.C Search/Retrieve via URL (SRU)

The Search/Retrieve via URL (SRU) is a standardized search protocol for internet queries. It allows for clients to submit searches and retrieve requests for matching records from servers. It's currently promulgated by the United States LOC [32].

The SRU requests are a type of URI called URL (see section 2.4.1.A), in which queries are represented with the Contextual Query Language (CQL). CQL's official specification clarifies the objective behind its use by stating: *"[...] that queries be human readable and writeable, and that the language be intuitive while maintaining the expressiveness of more complex languages."* [33].

As for the SRU replies, they are an XML document with a specifically designed schema [34].

2.4.3.D Search/Retrieve Web Service (SRW)

The Search/Retrieve Web Service (SRW) [35] is a modification of the SRU. The main difference between the two versions, is the way messages are carried from the client to the server. In SRU they are carried by a URL, whilst in SRW, XML over HTTP SOAP [36] is used.

2.4.4 Metadata Sharing Protocols

The following protocol descriptions characterize different ways to share metadata between entities. Whether it is through metadata harvesting protocols such as the Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH). Through resource descriptions and aggregations exchange as is done by the Open Archives Initiative Object Reuse and Exchange (OAI-ORE).

Regardless of means, all these protocols strive to achieve the common goal of promoting metadata sharing.

2.4.4.A Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH)

The OAI-PMH is, as is stated by its name, a protocol which was developed by the Open Archives Initiative. Its objective is to harvest metadata descriptions of records within an archive. Services can then be built to use metadata from several archives. The OAI protocol stands on a client-server architecture, having clients ("harvesters") requesting information from servers ("repositories").

The OAI protocol is has six possible requests (or verbs), which are carried within HTTP POST or GET methods. All OAI protocol requests have a two part structure [37, p. 6]. The first part is a "base-URL", which contains the internet host and port of the HTTP server which is acting as a repository. Additionally a path to act as the OAI protocol handler can be specified by the HTTP server. The second part are "keyword arguments", which are a list of key-value pairs. Each OAI protocol needs to have at least one key-value pair that specifies its request name.

As for the responses to OAI protocol requests, they are all encoded in XML according to an XML schema. This is due to verifiability purposes, for each response should conform to the XML schema defined in the protocol. Furthermore, the OAI protocol requests are always included within its responses. This measure provides for an easier machine batch processing of the responses.

There are four different requests in the OAI protocol. They are:

- GetRecord;
- Identify;
- ListIdentifier;
- ListMetadataFormats;

- ListRecords;
- ListSets.

The GetRecord request is used to retrieve individual records, (by providing the record key and metadata format), from items within a repository. The Identify request is used to retrieve information regarding a repository. It returns a readable name for the repository, it's base URL, the OAI protocol version that the repository supports and its administrator's e-mail address. Additionally it also provides a mechanism for individual entities to extend the request functionality. The ListIdentifier request is used to retrieve record identifiers that are available for harvesting from a repository. The ListMetadataFormats request is used to retrieve the metadata formats available in a repository. The ListRecords request is used to selectively harvest records from a repository. Finally the ListSets request is used to retrieve a repository set structure.

In terms of the data provider conformance and registration, the OAI expects data providers to fall under one of three mutually encapsulating layers, which are:

1. OAI-conformant
2. OAI-registered
3. OAI-namespace-registered

OAI-conformant is the innermost layer, and ensures that data providers support the protocol definition. The second layer, OAI-registered, ensures that data providers register in an OAI-maintained database, which is made available through the OAI web site. The outermost layer is called OAI-namespace-registered, it ensures that data providers name their records according to an OAI identifier naming convention. The adoption of this convention assures that record identifiers will resolvable via the OAI central resolution service, and thus made available at the OAI web site.

2.4.4.B Open Archives Initiative Object Reuse and Exchange (OAI-ORE)

The OAI-ORE [17] defines standards that help to describe and exchange aggregations of web resources. These aggregations are referred as compound digital objects

[17] Open Archives Initiative Object Reuse and Exchange website: `http://www.openarchives.org/ore/` Retrieved 2012-12-28

for each of the resources which compose them may have a distinct Multipurpose Internet Mail Extensions (MIME), (e.g., text, data, images, video, etc.).

The overall objective of the OAI-ORE is to allow applications to use the aggregated information within compound digital objects, by publishing a machine-readable document that describes that aggregation.

In order to achieve this objective, the ORE Model must define four separate entities [18]. They are:

- Aggregated Resource;
- Compound Digital Objects;
- Resource Map (ReM)
- Proxy.

The Aggregated Resource is any resource which is part of a Compound Digital Object. It is identified by an URI (see section 2.4.1.A).

The Compound Digital Object is composed by a series of Aggregated Resources, and it is also identified by an URI. However Compound Digital Objects, as conceptual constructs, do not require a "machine readable" representation.

The Resource Map (ReM) is a concept which was introduced to the ORE Model due to the need to follow Linked Data guidelines. These state that the inclusion of links to other related URIs must be made, in order to improve the discovery of related information on the web. Therefore the ReM, which is identified by an URI, comprises of a single Compound Digital Object and has a "machine readable" representation which details both their structure and semantics. ReMs can be made available on the web through several different syntaxes, such as Atom feeds, RDF/XML, RDFa and HTTP.

Finally a Proxy resource, which indicates an Aggregated Resource in the context of a specific Compound Digital Object. The Proxy is identified by an URI, which provides a mechanism to indicate specific Aggregated Resources in the context of the Compound Digital Object associated with the Proxy.

[18] ORE User Guide - Primer (2008-10-17): `http://www.openarchives.org/ore/1.0/primer` Retrieved 2012-12-28

3 Problem Analysis

This chapter focuses on both contextualizing and analysing the problem that is the aim of this dissertation. Therefore a closer look is taken at the procedures and structure of the Hemeroteca Municipal de Lisboa (Hemeroteca) for the optimization of its business processes workflow is this project's focus case.

This chapter is divided into three sections. Section 3.1 presents a contextualization of the focus case of this dissertation. Section 3.2 goes through the two business processes in play at the Hemeroteca. And finally section 3.3 displays the results of the analysis made on the business processes and identifies the problems with a specific selection of sub-processes.

3.1 Hemeroteca

The Hemeroteca is under the supervision of Câmara Municipal de Lisboa (CML). Its objective is to create and maintain a cultural heritage digital library, mostly comprised of newspapers and magazines whose intellectual property rights have expired, i.e., have fallen into public domain. In its own domain, Hemeroteca is Lisbon's second largest digital library, holding over half a million records, with a production between 140 to 150 thousand new records per year.

The Hemeroteca is expected to change its facilities in 2014, however it has been based on the palace of the counts of Tomar, which is located at Bairro Alto in Lisbon, since the year 1973. Currently the facility is over-capacitated, thus justifying the upcoming move.

Hemeroteca's digitization and image service has as a main goal the digitalizing and publishing of works whose rights have already fallen into public domain. There are four people currently attached to this service. Anabela Ferreira, João Oliveira, Joaquina Cunha and Paula Cardoso. All four staff members have years of experience working in the field of digital libraries, however only three have had specific training to do so. Both Joaquina Cunha and Paula Paula have a technical course in articles and libraries, and João Oliveira holds a graduate course in information and documentation sciences, specialized in libraries.

The service's workload and responsibility is split between all the four members of staff, however due to the nature of his training João Oliveira occupies a *de facto* management position within the service. With Alvaro de Matos overseeing the service as its coordinator.

In terms of resources, the digitization and image service has four flatbed scanners, with plans on acquiring a book scanner after the move to the new facilities in 2014. It has five computers which have Intel Core 2 Duo processors, with RAM capacities ranging from 900MB to 3GB, all of which are equipped with Microsoft's Windows XP Professional operating system.

As for the software tools in use by the staff of the digitization and image service, eight software tools should be considered:

- **Microsoft FastStone Image Viewer**[1] An image browser, organizer, converter and editor designed for Microsoft Windows by FastStone Soft and provided free of charge for non-commercial use;

- **PAPAIA** [38, p. 3] A software tool which was originally developed to be used at the BND. PAPAIA processes batches of images, and can preform actions such as: renaming images, editing the TIFF headers and registering structural metadata;

[1] Microsoft FastStone Image Viewer: `http://www.faststone.org/` Retrieved 2013-09-30

- **Adobe Acrobat** Hemeroteca holds one Adobe Acrobat[2] software licence. Adobe Acrobat is a software application developed by Adobe Systems to view, create, manipulate, print and manage files in PDF (see section 2.2.2.B);
- **JPEGToPDF**[3] A freeware software application used to convert JPEG image files to PDF, that does not require Adobe Acrobat or Acrobat Reader ;
- **BecyPDFMetaEdit**[4] A freeware software application which loads PDFs and allows editing of its descriptive metadata, i.e., author, title, subject and keywords of the document;
- **ContentE**[5] [39] A software application developed by Gilberto Pedrosa. It produces master copies for preservation, copies for access, structural descriptions in METS, (see section 2.3.2.A) and also indexes. The master copies are organized within a folder structure, which has a folder for each MIME [40] (e.g., TIFF, JPEG, Portable Network Graphics (PNG), Graphics Interchange Format (GIF), PDF or Text File (TXT));
- **Microsoft FrontPage** An HTML editor and web site administration tool which was developed by Microsoft and was distributed with Microsoft Office from 1997 to 2003. However it has since been discontinued;
- **WinSCP (Windows Secure CoPy)**[6] A free and open-source SFTP, SCP and FTP client for Microsoft Windows. It offers secure file transfer between a local and a remote computer as well as a basic file manager and file synchronization functionality.

As for Information Technology (IT) support, it is provided by the Departamento de Modernização e Sistemas de Informação (DMSI), which is inserted within the Divisão de Administração de Sistemas e Infrastructuras (DASI) of the CML. This delegation of responsibility means that none of the four members of staff currently working at

[2] Adobe Acrobat website: `http://www.adobe.com/pt/products/acrobat.html` Retrieved 2013-09-30

[3] JPEGToPDF website: `http://www.jpegtopdf.com/` Retrieved 2013-09-30

[4] BecyPDFMetaEdit website: `http://www.becyhome.de/becypdfmetaedit/description_eng.htm` Retrived 2013-09-30

[5] ContentE website at the BND: `http://purl.pt/index/geral/PT/infoProfContentE.html` Retrived 2013-09-30

[6] WinSCP website: `http://winscp.net/eng/index.php` Retrieved 2013-09-30

Hemeroteca's digitization and image service has any experience in managing information systems. This is an important constraint to the design presented in chapter 4.

3.2 Business Processes

Hemeroteca has two business processes. The P1 has as overall objective the publishing of a digitized work, and it is comprised by four tasks, as can be seen in figure 3.1. A brief description of each task is given in table 3.1. However each the four tasks can be perceived as a collapsed sub-process, therefore a detailed description will be provided in sections 3.2.1 through 3.2.4.

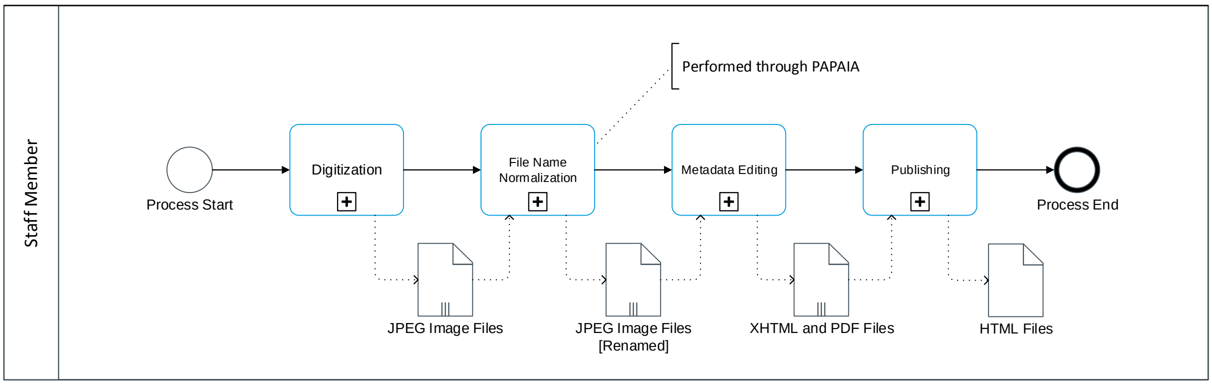


Figure 3.1: The Process Digitizing and Publish (P1)

The P2 has as main goal updating a catalogue hierarchy of which Hemeroteca's catalogue sits on the bottom. As it was decided by the stakeholder not to tackle this business process, the tasks that make up this process are only briefly described in section 3.2.5.

3.2.1 Sub-Process Digitizing (P1.1)

The P1.1 consists on the creation of a series of JPEG digital images from the original publication. There are two distinct workflows that originate from performing this sub-process, as can be seen in figure 3.2.

The first task comprises on the selection of publications to digitize.

Activity Name	Summarized Description
Digitization	Creation of a digital images by digitizing an original publication
File name normalization	Renaming the image files according to a specific naming convention
Metadata editing	Creating PDF files, and editing both the descriptive and structural medatada of the digital publication
Publishing	Publishing the digital publication at Hemereoteca's website

Table 3.1: First business process activities

The second task is the digitization itself. This produces digital images in the TIFF format, (see section 2.2.2.A), which have a resolution of 300 dpi. This task is generally preformed internally via a non-destructive scanning process, through the use of a flatbed scanner. However it may be preformed externally, whenever two separate conditions are met: an excessive number of pending objects to digitize, and the necessary funds must be made available by the CML to subsidize the outsourcing process. External object digitization implies two tasks for the staff element in charge of the subprocess. First it must send the publication to the external entity for it to be digitized. And second, the ordering of the TIFF files delivered by the external digitization entity must be verified, and if needed corrected.

Regardless of whether the object digitization is preformed internally or externally, the following task is the storing of all the resulting TIFF files. Files are stored both in external discs and DVDs, which are archived within Hemeroteca's facility.

The final task is the creation of JPEG files, (see section 2.2.2.C). This is necessary, for the JPEG file format is designed for online publication as opposed to the TIFF file format which, as a bitmap based format, is suitable for storage purposes. The conversion of the TIFF files to JPEG and its subsequent image post processing, is made through the FastStone Image Viewer, which was described in section 3.1.

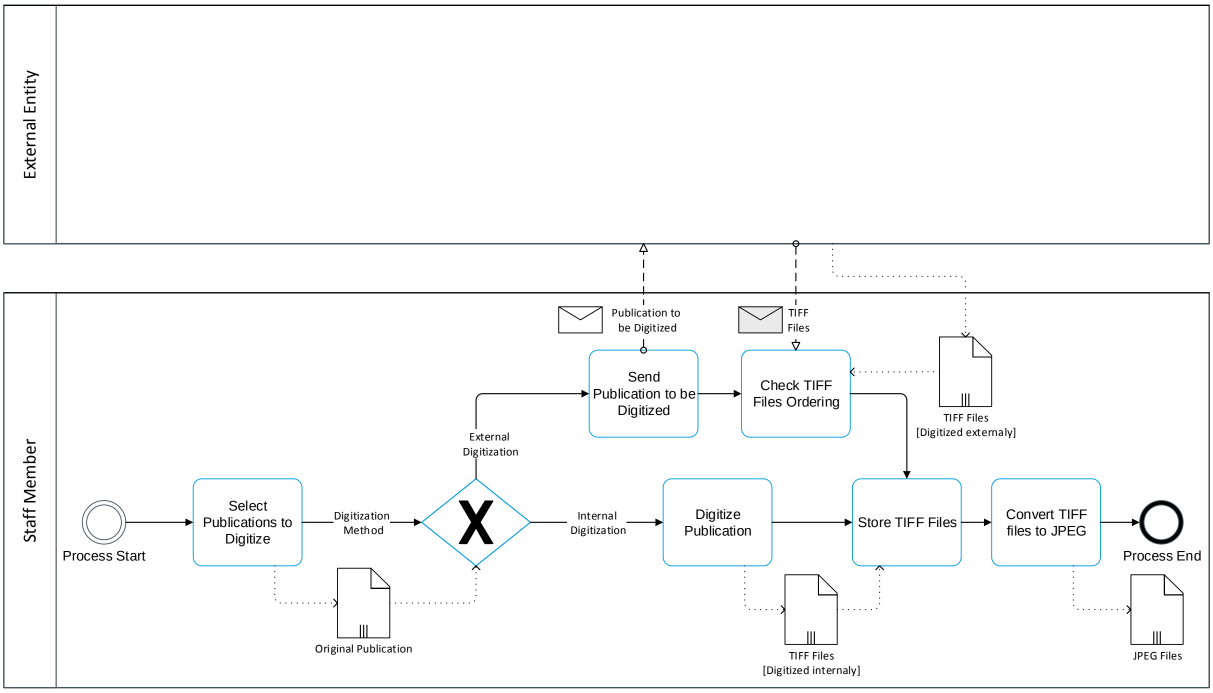

External Entity
Staff Member
Publication to be Digitized
TIFF Files
Send Publication to be Digitized
Check TIFF Files Ordering
TIFF Files [Digitized externaly]
External Digitization
Process Start
Select Publications to Digitize
Digitization Method
Internal Digitization
Digitize Publication
Store TIFF Files
Convert TIFF files to JPEG
Process End
Original Publication
TIFF Files [Digitized internaly]
JPEG Files

Figure 3.2: The Sub-Process Digitizing (P1.1)

3.2.2 Sub-Process File Name Normalization (P1.2)

The P1.2 (see figure 3.1) consists on the naming of the images according to a determined naming convention, which is in use at the BND [41]. This sub-process serves two purposes: The identification of the image through a unique identifier, and the display of the image's technical features in its name. This procedure is accomplished through the use of a software tool named PAPAIA (see section 3.1). Using PAPAIA the staff at the digitization and image service processes the JPEG image files that originated in the digitization sub-process. The end product is a renamed set of JPEG image files.

3.2.3 Sub-Process Metadata Editing (P1.3)

The P1.3 (see figure 3.3) starts by picking up on the products from the file name normalization sub-process, and proceeds to create both a PDF file that aggregates all the JPEG image files and structural metadata describing how the JPEG image files should be organized to reproduce the original publication's structure.

Hemeroteca's digital objects are attainable to users as PDFs, (see section 2.2.2.B),

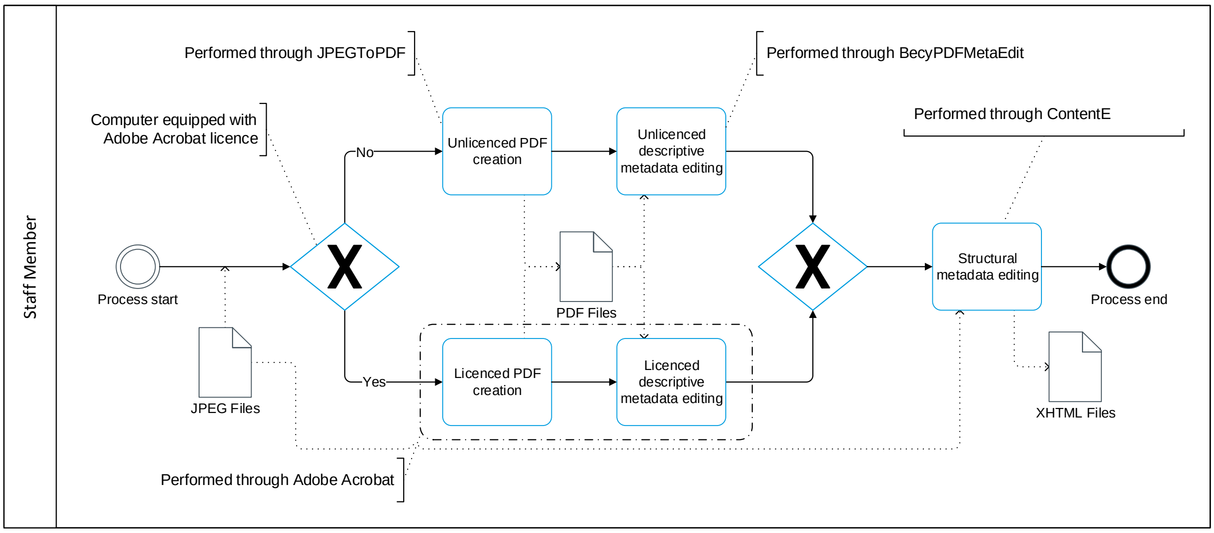

Staff Member
Performed through JPEGToPDF
Performed through BecyPDFMetaEdit
Computer equipped with Adobe Acrobat licence
Performed through ContentE
No
Unlicenced PDF creation
Unlicenced descriptive metadata editing
Process start
PDF Files
Structural metadata editing
Process end
Yes
Licenced PDF creation
Licenced descriptive metadata editing
JPEG Files
XHTML Files
Performed through Adobe Acrobat

Figure 3.3: The Sub-Process Metadata Editing (P1.3)

and JPEG image files (the latter inserted within HTML pages). This implies that the creation of PDF files must be a necessary task within the P1.3. However only a single Adobe Acrobat licence is held at the digitization and image service, this implies that the remainder of the staff must use a freeware software application (JPEGToPDF, see section 3.1) to create PDF files.

The task that follows is the enrichment of the PDF files with descriptive metadata. Again if the staff member that is preforming the P1.3 does not hold an Adobe Acrobat licence, he will be forced to use a freeware application (BecyPDFMetaEdit, see section 3.1) to perform the task.

The structural metadata editing task is performed with the assistance of a software tool called ContentE (see section 3.1), which also produces access copies as XHTML files.

3.2.4 Sub-Process Publishing (P1.4)

The P1.4 (see figure 3.4) consists on the creation of HTML files which will then be published at a external server which is managed by the DMSI-DASI. The whole sub-process relies heavily on manual tasks being performed by staff members of the digitizing and image service.

The first task of the sub-process depends on whether the work to which the new publication belongs exists at Hemeroteca's work index. The work index is a web page

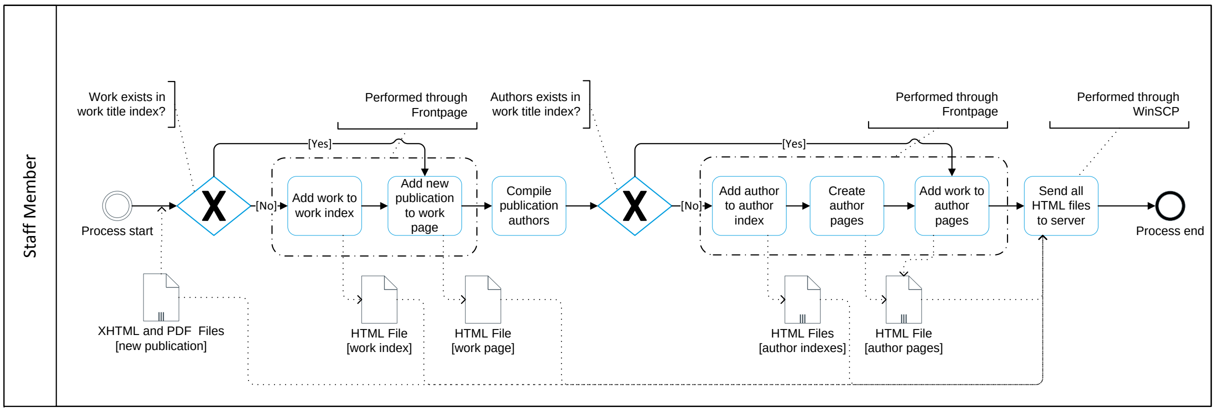
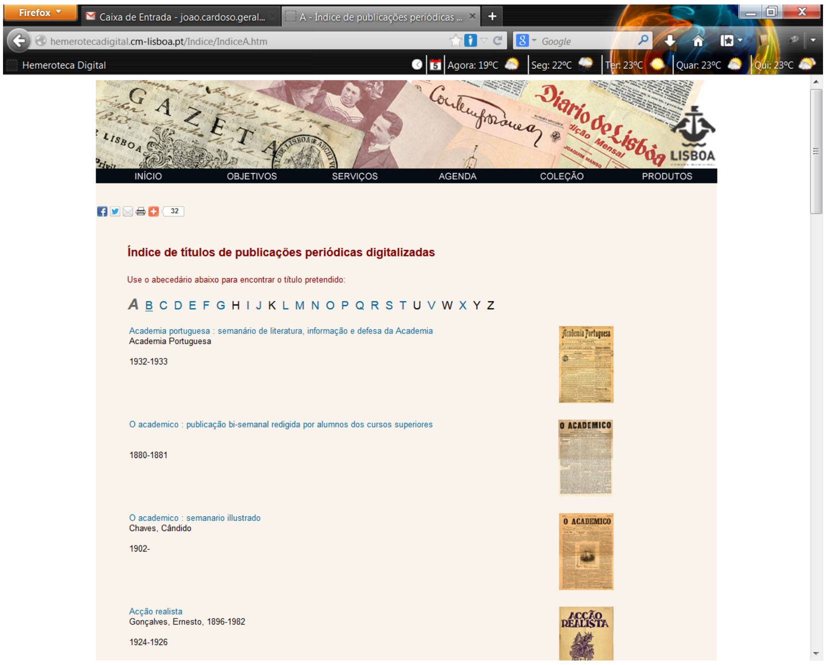

Staff Member
Work exists in work title index?
Performed through Frontpage
Authors exists in work title index?
Performed through Frontpage
Performed through WinSCP
[Yes]
[Yes]
Process start
[No]
Add work to work index
Add new publication to work page
Compile publication authors
[No]
Add author to author index
Create author pages
Add work to author pages
Send all HTML files to server
Process end
XHTML and PDF Files [new publication]
HTML File [work index]
HTML File [work page]
HTML Files [author indexes]
HTML File [author pages]

Figure 3.4: The Sub-Process Publishing (P1.4)

containing the titles of all the works that have been digitized to date (see figure 3.5). If the work does not exist, then it must be added to the work index, and that involves creating a new work page (see figure 3.6). If the work exists in the work index, then the new publication is simply added to the existing work page. Both these tasks are performed using FrontPage (see section 3.1), which means manually editing HTML files.



Índice de títulos de publicações periódicas digitalizadas

Use o abecedário abaixo para encontrar o título pretendido:

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Academia portuguesa : semanário de literatura, informação e defesa da Academia
Academia Portuguesa

1932-1933

O academico : publicação bi-semanal redigida por alumnos dos cursos superiores

1880-1881

O academico : semanario illustrado
Chaves, Cândido

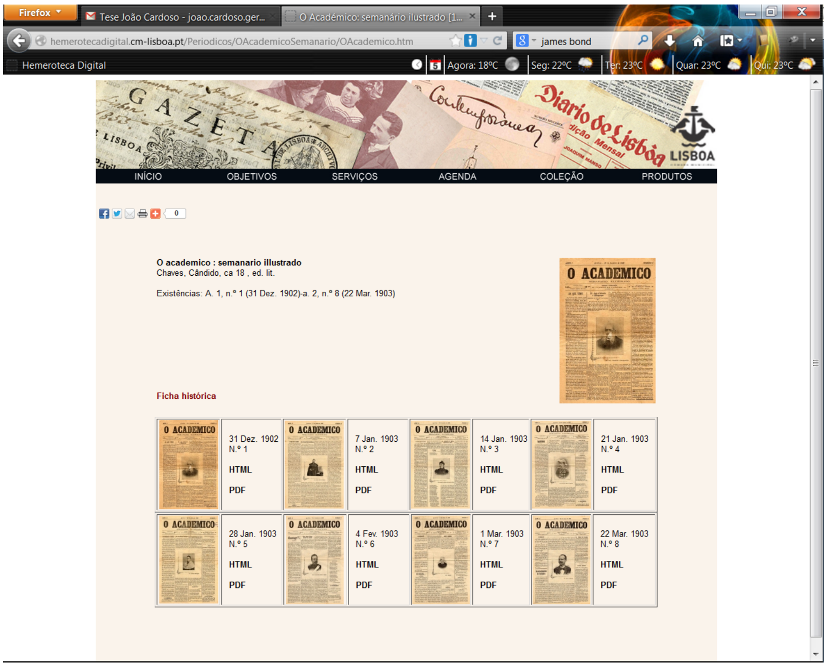
1902-

Acção realista
Gonçalves, Ernesto, 1896-1982

1924-1926

Figure 3.5: An example of a section of the work index

Once the necessary work related HTML files have been created, the next task is



Firefox
Tese João Cardoso - joao.cardoso.ger...
O Académico: semanário ilustrado [1...
hemerotecadigital.cm-lisboa.pt/Periodicos/OAcademicoSemanario/OAcademico.htm
james bond
Hemeroteca Digital
Agora: 18ºC
Seg: 22ºC
Ter: 23ºC
Quar: 23ºC
Qui: 23ºC
LISBOA
INÍCIO
OBJETIVOS
SERVIÇOS
AGENDA
COLEÇÃO
PRODUTOS
0
O ACADEMICO

O academico : semanario illustrado
Chaves, Cândido, ca 18 , ed. lit.

Existências: A. 1, n.º 1 (31 Dez. 1902)-a. 2, n.º 8 (22 Mar. 1903)

Ficha histórica

31 Dez. 1902
N.º 1 — **HTML** — **PDF**

7 Jan. 1903
N.º 2 — **HTML** — **PDF**

14 Jan. 1903
N.º 3 — **HTML** — **PDF**

21 Jan. 1903
N.º 4 — **HTML** — **PDF**

28 Jan. 1903
N.º 5 — **HTML** — **PDF**

4 Fev. 1903
N.º 6 — **HTML** — **PDF**

1 Mar. 1903
N.º 7 — **HTML** — **PDF**

22 Mar. 1903
N.º 8 — **HTML** — **PDF**

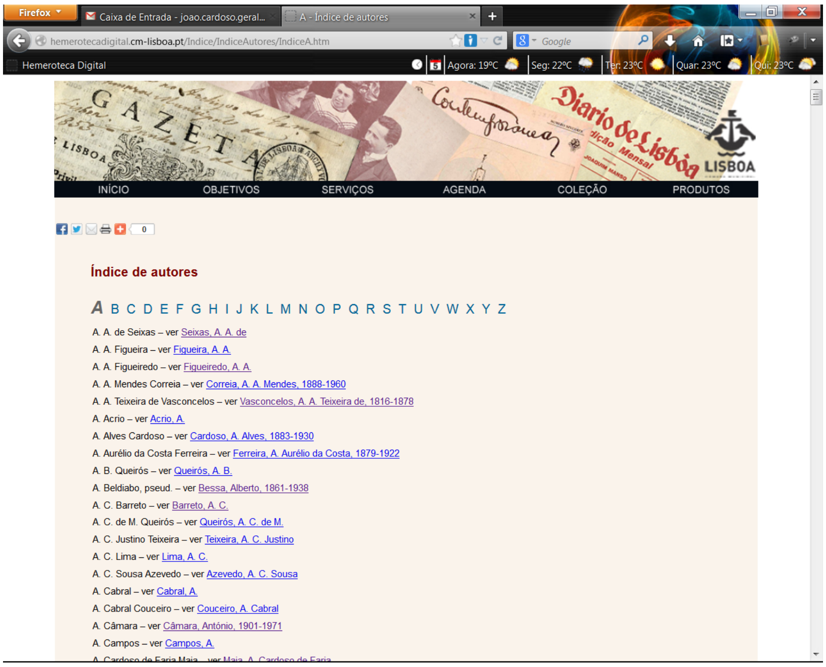
Figure 3.6: An example of a work page

to compile a list of all the authors which collaborated on the publication. This will be used to check whether all the listed authors exist at the author index (see figure 3.7). If any of them does not exist they must first be added to the author index and a new author page must be created for that author (see figure 3.8). Regardless of the author existence in the author index, the work must be added to each individual author page. Again, these task are performed using FrontPage.

The final task is to send all newly generated or edited HTML files to the external server which, as mentioned, is managed by the DMSI-DASI. This is done by using a safe connection through the SFTP, SCP and FTP client for Microsoft Windows WinSCP (see section 3.1).

3.2.5 Process Metadata Sharing (P2)

All objects digitized and published by Hemeroteca are registered at the Hemeroteca's catalogue, which is hosted at a server running the OAI protocol, (see section 2.4.4.A). Hemeroteca's catalogue, despite its individual importance, is only a part of a larger catalogue made up by contributions from all of CML's digital libraries. This catalogue

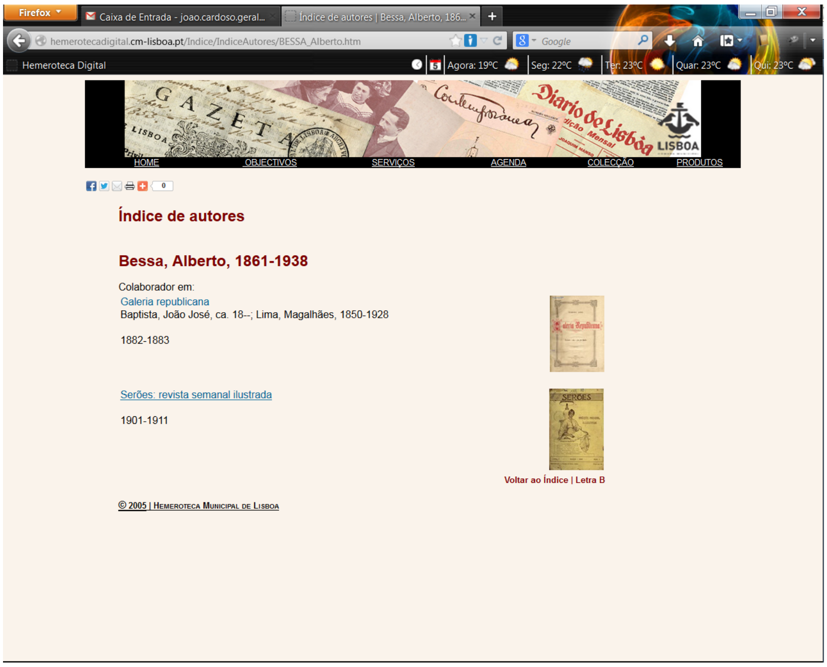


Índice de autores

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

A. A. de Seixas – ver Seixas, A. A. de
A. A. Figueira – ver Figueira, A. A.
A. A. Figueiredo – ver Figueiredo, A. A.
A. A. Mendes Correia – ver Correia, A. A. Mendes, 1888-1960
A. A. Teixeira de Vasconcelos – ver Vasconcelos, A. A. Teixeira de, 1816-1878
A. Acrio – ver Acrio, A.
A. Alves Cardoso – ver Cardoso, A. Alves, 1883-1930
A. Aurélio da Costa Ferreira – ver Ferreira, A. Aurélio da Costa, 1879-1922
A. B. Queirós – ver Queirós, A. B.
A. Beldiabo, pseud. – ver Bessa, Alberto, 1861-1938
A. C. Barreto – ver Barreto, A. C.
A. C. de M. Queirós – ver Queirós, A. C. de M.
A. C. Justino Teixeira – ver Teixeira, A. C. Justino
A. C. Lima – ver Lima, A. C.
A. C. Sousa Azevedo – ver Azevedo, A. C. Sousa
A. Cabral – ver Cabral, A.
A. Cabral Couceiro – ver Couceiro, A. Cabral
A. Câmara – ver Câmara, António, 1901-1971
A. Campos – ver Campos, A.
A. Cardoso de Faria Maia – ver Maia, A. Cardoso de Faria

Figure 3.7: An example of a section of the author index


Firefox
Caixa de Entrada - joao.cardoso.geral...
Índice de autores | Bessa, Alberto, 186...
hemerotecadigital.cm-lisboa.pt/Indice/IndiceAutores/BESSA_Alberto.htm
Google
Hemeroteca Digital
Agora: 19ºC
Seg: 22ºC
Ter: 23ºC
Quar: 23ºC
Qui: 23ºC
LISBOA
HOME
OBJECTIVOS
SERVIÇOS
AGENDA
COLECÇÃO
PRODUTOS
0

Índice de autores

Bessa, Alberto, 1861-1938

Colaborador em:

Galeria republicana
Baptista, João José, ca. 18--; Lima, Magalhães, 1850-1928

1882-1883

Serões: revista semanal ilustrada

1901-1911

Voltar ao Índice | Letra B

© 2005 | Hemeroteca Municipal de Lisboa

Figure 3.8: An example of an author page

is called Bibliotecas Municipais de Lisboa (BLX) [7].

[7] Bibliotecas Municipais de Lisboa website: `http://blx.cm-lisboa.pt/` Retrieved 2012-12-20

Hemeroteca is also taking part on the creation of a national wide catalogue, the Registo Nacional de Objectos Nacionais (RNOD) [8]. RNOD is managed by the BNP, and aims to become the most comprehensive national catalogue. Hemeroteca contributed to this catalogue by merging its own catalogue into RNOD's.

On a continental scale, Hemeroteca has participated in the EuropeanaLocal [9], a now defunct project by the Europeana Fundation. This meant that most of Hemeroteca's records are also available at the Europeana.eu internet portal. However since the end of the EuropeanaLocal project, the staff at Hemeroteca has been unable to update new records to Europeana's catalogue, for they lack the necessary training to do so.

3.3 Consolidated Analysis

This section analyses the problems that are to be tackled throughout this project's execution, (which is detailed in chapter 4).

As stated in section 3.2, the P1 at Hemeroteca comprised four tasks that can be perceived as a collapsed sub-process. The stakeholder defined that both the P2 and the P1.1 should not be considered for optimization. Therefore after studying each of the remainder three sub-processes, it became apparent all of them could suffer improvements that would not only simplify them, but also lead them to become more efficient in terms of time consumption.

The P1.2 presented a challenge because it was based on an outdated software application (see section 3.2.2). PAPAIA, as described in section 3.1, was originally designed for the BND. However it was never able to display its full potential when in use by the digitizing and image services at Hemeroteca. Features such as editing TIFF headers or registering structural metadata were either never performed, or were alternatively completed through the use of different software applications. PAPAIA was solely used for renaming batches of image files, and its interface proved to be so complex, (see figure 3.9), that none of the four staff members of the digitizing and image services fully understood how to take advantage of the renaming abilities that PAPAIA offered. All image files were thus renamed with the same scheme, the only variable

[8] Registo Nacional de Objectos Nacionais website: `http://rnod.bnportugal.pt/rnod/` Retrieved 2012-12-20

[9] EuropeanaLocal Project website: `http://pro.europeana.eu/web/europeanalocal` Retrieved 2013-01-04

being its order number.

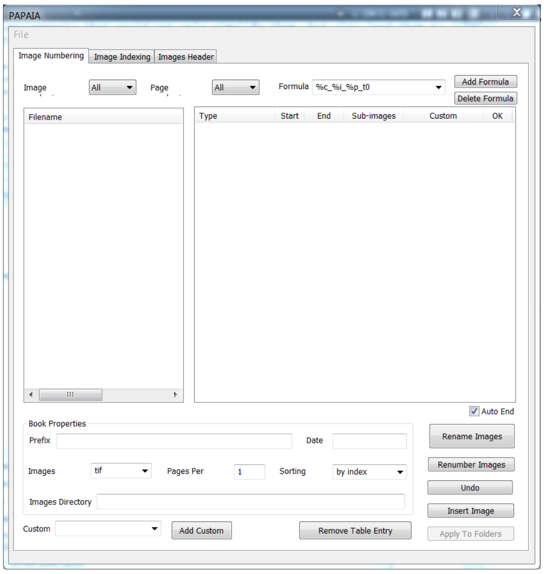


Image: All
Page: All
Formula: %c_%i_%p_t0
☑ Auto End
Images: tif
Pages Per: 1
Sorting: by index

Figure 3.9: PAPAIA's main interface

The P1.3 was riddled with redundant tasks, particularly in what referred to the descriptive metadata editing. There were two different software applications being used to both create and add descriptive metadata to PDF files. In terms of the strucutral metadata addition, the version of the software application ContentE was outdated (The current version is 3.9, whilst the version used at the digitizing and image service is 1.6). The staff at the digitizing and image service had also not been specifically trained for its use, and therefore could not take full advantage of the application's potential.

As can be seen in section 3.2.4, the P1.4 relies heavily on manual procedures and has little automation. This means that the staff members at the digitizing and image services spend a heavy portion of their time editing HTML files, when they could be spending it processing more works. Additionally the manual editing of HTML files leads to syntax errors, which hamper the consistency of data publishing.

4

Design and Solution

This chapter focuses on presenting solutions for the problems stated in section 3.3.

The chapter is divided into three sections. Section 4.1 goes through the solution to problems related to the file name normalization sub-process (see section 3.2.2), and presents the design to the software application that was developed to solve them. Section 4.2 focuses on the measures that were taken to solve the issues related to the metadata editing sub-process (see section 3.2.3). Finally section 4.3 analyses the solution presented for the publishing sub-process (see section 3.2.4), and describes the design of the software application that was developed for that intent.

4.1 File Name Normalization

As mentioned in section 3.3 there were three main issues that needed to be tackled in order to optimize the P1.2. These issues were:

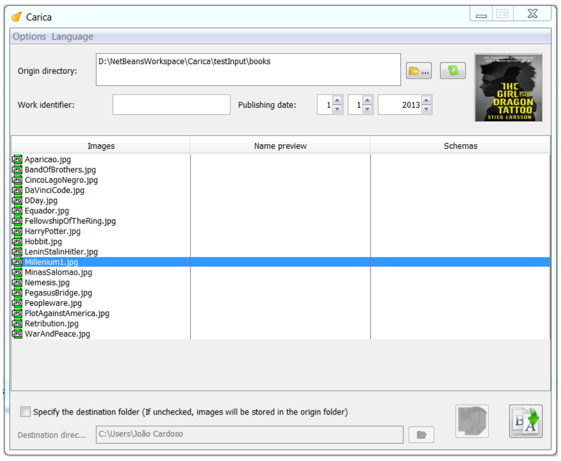
- Image files needed to be renamed according to the naming convention used at the BND.

- PAPAIA was an outdated software application, that had been designed specifically for the BND;

- The staff at the digitizing and image service should be able to understand and take advantage of all of the software application's features.

Having these issues in mind, a decision was made to scrap PAPAIA and replace it for an application that would not only fit the digitizing and image service needs more efficiently, but also allow its staff to take full advantage of its features.

The software application that was to replace PAPAIA was called Carica (see user manual at appendix A). The name Carica is a wordplay based on the fact that Carica is a genus of flowering plants in the family Caricaceae, which includes the papaya. PAPAIA's substitution would not imply any changes to the P1.2 itself, for changes would only occur in terms of how the task was performed and not on the definition of the task itself.



Origin directory: D:\NetBeansWorkspace\Carica\testInput\books

Work identifier:

Publishing date: 1 1 2013

Images	Name preview	Schemas
Aparicao.jpg		
BandOfBrothers.jpg		
CincoLagoNegro.jpg		
DaVinciCode.jpg		
DDay.jpg		
Equador.jpg		
FellowshipOfTheRing.jpg		
HarryPotter.jpg		
Hobbit.jpg		
LeninStalinHitler.jpg		
Millenium1.jpg		
MinasSalomao.jpg		
Nemesis.jpg		
PegasusBridge.jpg		
Peopleware.jpg		
PlotAgainstAmerica.jpg		
Retribution.jpg		
WarAndPeace.jpg		

□ Specify the destination folder (If unchecked, images will be stored in the origin folder)

Destination direc... C:\Users\João Cardoso

Figure 4.1: Carica main dialog

Carica was to replicate PAPAIA's image file renaming feature. However it upgraded it by going a step further and allowing users to create, edit and apply a set of renaming schemas to a batch of image files (see figure 4.1). There are two main concepts behind Carica. These are the concept of schema and work, which are described in table 4.1.

Concept	Description
schema	A schema is a set of options, which derive from BND's naming convention, and that are later applied to a batch of image files, thus renaming them
Work	A work is a set of schemas that are applied in an aggregated fashion to a batch of image files. Works are best suited for processing batches of image files derived from publications, because the structure of the publication remains unaltered regardless of the volume

Table 4.1: Carica's main concepts

4.1.1 Use Cases in Carica

There are four use cases that define interactions between a user and Carica. The four use cases are depicted in figure 4.2.

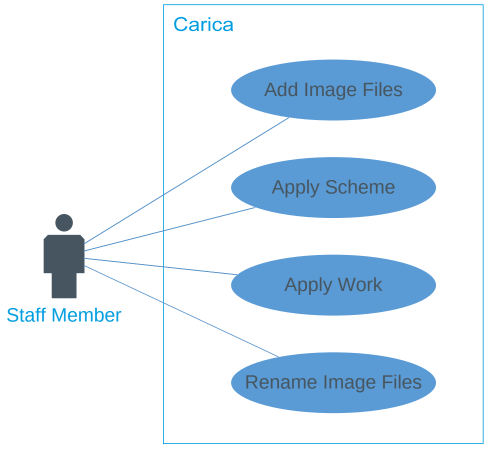


Figure 4.2: Carica's use cases

- **Add image files** Users may add image files to Carica. Once added the user may remove them or rename them, which implies having applied a schema or a work;
- **Apply schema** Users may apply a schema to a batch of image files. This implies having created a schema or edited an existing schema;

- **Applying work** Users may apply a work to a batch of image files. This implies having created a work or edited an existing work;

- **Rename image files** Users can rename batches of image files, by applying schemas or works;

Considering that the schema and work concepts are key in Carica, it becomes necessary to detail how schemas and works are applied to a batch of image files.

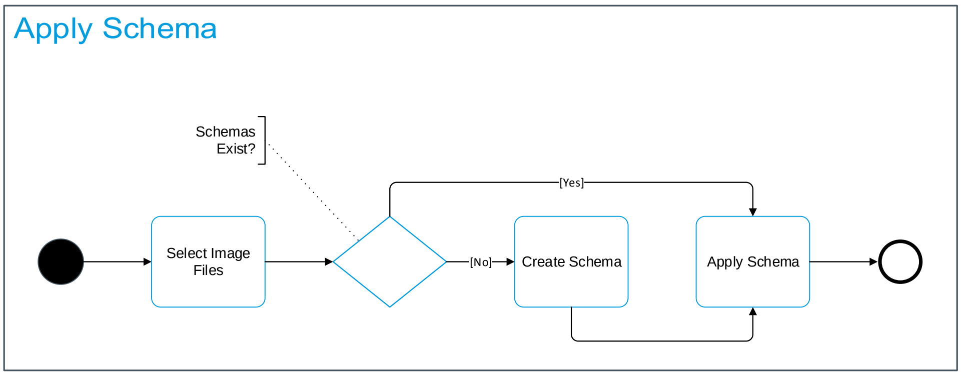
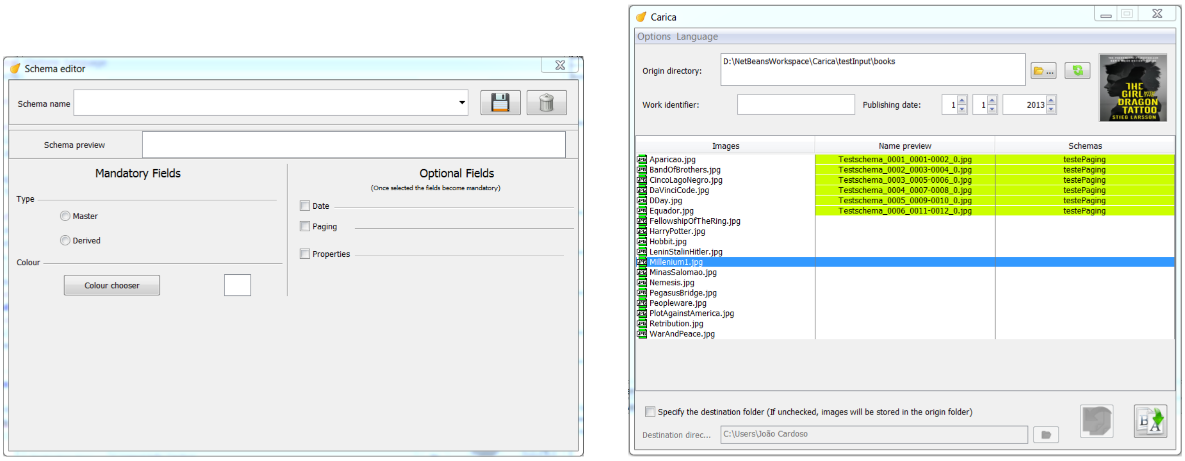


Figure 4.3: Apply schema activity diagram

Figure 4.3 depicts the apply schema activity diagram. The first activity is the selection of the image files to which the schema will be applied. Before progressing to the schema application activity the user is faced with the possibility of not having yet created any schema in Carica. If there aren't any created schemas, the user is asked to create a new schema, which he can do by using Carica's schema editor (see figure 4.4(a)). When there are existing schemas, the user will choose the schema he wishes to apply and the schema will apply itself to each of the image files within the selected batch, as can be seen in figure 4.4(b).

The work application activity diagram can be seen in figure 4.5. The first activity, just as in the scheme application activity diagram, corresponds to the selection of the image files to which the schema will be applied, as can be seen in figure 4.6(b). If any works have been created, the next activity the user performs is the application of a selected work. Otherwise the user must create a new work, as can be seen in figure 4.6(a). However since a work is a set of schemas that are applied in an aggregated fashion to a batch of image files, the user can only create a new work if at least one schema has been created.



Schema editor

Schema name

Schema preview

Mandatory Fields

Type

○ Master

○ Derived

Colour

Colour chooser

Optional Fields

(Once selected the fields become mandatory)

□ Date

□ Paging

□ Properties

Carica

Options Language

Origin directory: D:\NetBeansWorkspace\Carica\testInput\books

Work identifier:

Publishing date: 1 1 2013

Images	Name preview	Schemas
Aparicao.jpg	Testschema_0001_0001-0002_0.jpg	testePaging
BandOfBrothers.jpg	Testschema_0002_0003-0004_0.jpg	testePaging
CincoLagoNegro.jpg	Testschema_0003_0005-0006_0.jpg	testePaging
DaVinciCode.jpg	Testschema_0004_0007-0008_0.jpg	testePaging
DDay.jpg	Testschema_0005_0009-0010_0.jpg	testePaging
Equador.jpg	Testschema_0006_0011-0012_0.jpg	testePaging
FellowshipOfTheRing.jpg		
HarryPotter.jpg		
Hobbit.jpg		
LeninStalinHitler.jpg		
Millenium1.jpg		
MinasSalomao.jpg		
Nemesis.jpg		
PegasusBridge.jpg		
Peopleware.jpg		
PlotAgainstAmerica.jpg		
Retribution.jpg		
WarAndPeace.jpg		

□ Specify the destination folder (If unchecked, images will be stored in the origin folder)

Destination direc... C:\Users\João Cardoso

(a) Carica schema editor

(b) Schema application

Figure 4.4: Schema editor and schema application

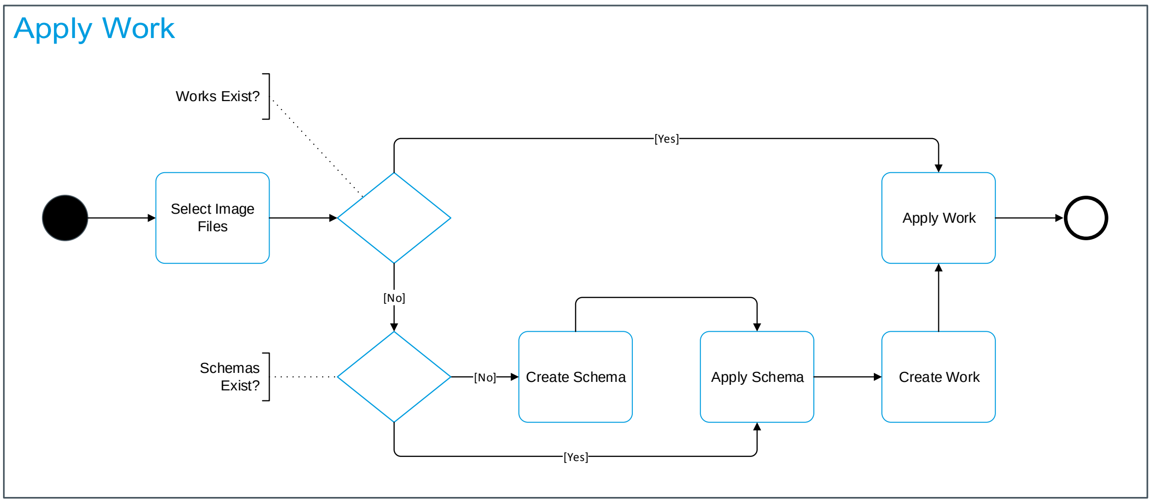


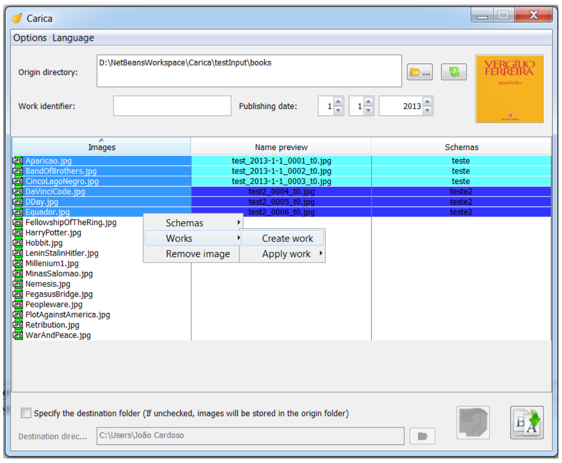
Figure 4.5: Apply work activity diagram

4.1.2 Carica Application Architecture

The Carica software application was developed using the Java programming language[1], using a multi-tier architecture, which in this particular case holds three tiers, as can be seen in figure 4.7.

The data access tier holds the classes that implement domain classes for the ap-

[1] Oracle website for Java developers: `http://www.oracle.com/technetwork/java/index.html` Retrieved 2013-10-08



Carica

Origin directory: D:\NetBeansWorkspace\Carica\testInput\books

Work identifier:

Publishing date: 1 1 2013

Images	Name preview	Schemas
Aparicao.jpg	test_2013-1-1_0001_t0.jpg	teste
BandOfBrothers.jpg	test_2013-1-1_0002_t0.jpg	teste
CincoLagoNegro.jpg	test_2013-1-1_0003_t0.jpg	teste
DaVinciCode.jpg	test2_0004_t0.jpg	teste2
DDay.jpg	test2_0005_t0.jpg	teste2
Equador.jpg	test2_0006_t0.jpg	teste2
FellowshipOfTheRing.jpg		
HarryPotter.jpg		
Hobbit.jpg		
LeninStalinHitler.jpg		
Millenium1.jpg		
MinasSalomao.jpg		
Nemesis.jpg		
PegasusBridge.jpg		
Peopleware.jpg		
PlotAgainstAmerica.jpg		
Retribution.jpg		
WarAndPeace.jpg		

☐ Specify the destination folder (If unchecked, images will be stored in the origin folder)

Destination direc... C:\Users\João Cardoso

(a) Creation of a work in Carica

Carica

Origin directory: D:\NetBeansWorkspace\Carica\testInput\books

Work identifier: teste

Publishing date: 1 1 2012

Images	Name preview	Schemas
Aparicao.jpg		
BandOfBrothers.jpg		
CincoLagoNegro.jpg		
DaVinciCode.jpg		
DDay.jpg		
Equador.jpg		
FellowshipOfTheRing.jpg		
HarryPotter.jpg		
Hobbit.jpg		
LeninStalinHitler.jpg		
Millenium1.jpg		
MinasSalomao.jpg		
Nemesis.jpg		
PegasusBridge.jpg		
Peopleware.jpg		
PlotAgainstAmerica.jpg		
Retribution.jpg		
WarAndPeace.jpg		

☐ Specify the destination folder (If unchecked, images will be stored in the origin folder)

Destination direc... C:\Users\João Cardoso

(b) Application of a work in Carica

Figure 4.6: Work creation and application

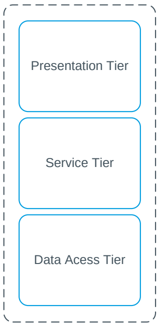


Figure 4.7: Carica multi-tier architecture

plication, such as schemas, names, batches and works. The service tier classes on the other hand interacts with the data access tier, by using its classes to implement the logic behind all the functionalities Carica provides. Finally the presentation tier classes implement a graphical user interface with which the user can interact and take full advantage of Carica's functionalities.

4.2 Metadata Editing

The P1.3 presented three primary issues, which were:

- Two different workflows existed for creating PDF files and enriching them with descriptive metadata;
- The version of the software application ContentE was outdated;
- The staff at the digitizing and image service were not specifically trained to work with ContentE.

The first issue resulted from the fact that only a single Adobe Acrobat licence was held by the digitization and image service at Hemeroteca. This implied that only a staff member was allowed to perform both those tasks using the Adobe Acrobat software tool, whilst the remainder of the staff members had to resort to freeware software applications, respectively JPEGToPDF and BecyMetaPDFEdit. The second issue on the other hand was a consequence of the lack of dedicated IT support to the digitization and image service. This meant that the software application ContentE had never been updated from the originally installed version 1.6, and was therefore outdated. The third issue was the lack of training on the use of ContentE, which implied that the staff at the digitizing and image service could not take full advantage of the functionalities provided by ContentE.

The current version of ContentE (v3.6) is able to both generate PDF files and enrich them with descriptive metadata. This meant that a solution to the first two issues could be achieved if the ContentE version in use at the digitizing and image service was to be updated to the current version. This task was done and the first two issues were solved in this fashion. Additionally Gilberto Pedrosa volunteered to provide assistance in both installing the new version and training staff members in its use, thus solving the third issue.

The optimized P1.3 (see figure 4.8) is therefore completely different from the original P1.3 presented in section 3.2.3. The whole sub-process is performed through the use of ContentE, and implies three tasks. The first task is the creation of PDF files from the existing JPEG files, so that each PDF will reflect the structure of the originally digitized publication. The second task is the editing of the descriptive metadata on each of those PDF files. The final task is to create a structural metadata record which describes the structure of the JPEG image files in order to replicate the structure of the

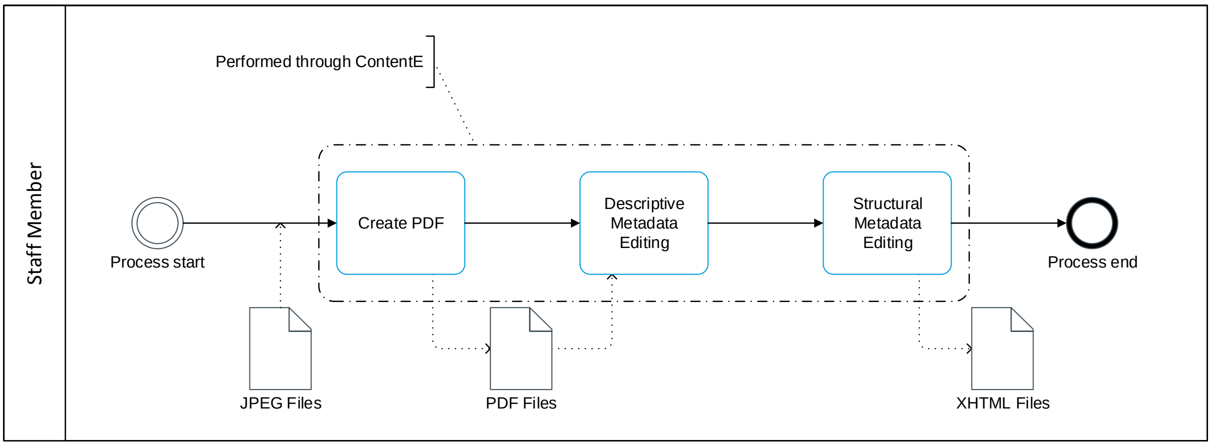

Staff Member
Performed through ContentE
Process start
Create PDF
Descriptive Metadata Editing
Structural Metadata Editing
Process end
JPEG Files
PDF Files
XHTML Files

Figure 4.8: The optimized Sub-Process Metadata Editing (P1.3)

original publication. The outcome of this operation is the creation of a publication copy in XHTML format, which will later be published in the P1.4.

4.3 Publishing

In what refers to the P1.4 there were three issues to solve, which were mentioned in section 3.3. The issues were:

- The sub-process relies heavily on manual procedures;
- There are no automated tasks;
- Manual editing of HTML files leads to syntactic errors and overall lack of consistency.

The challenge was to solve each of three issues by developing a solution which would have to consider the limitations imposed by the digitizing and image service context. This would imply implementing a system which had to be both easy to use and maintain by the staff members of the digitizing and image service.

Obviously the ideal system for this sub-process would be to implement a relational database in which the records that make up the authority file for the digitizing and image service would be stored. Then a software application equipped with a graphical user interface would be used to create the necessary indexes in the form of HTML files, from the records stored within the relational database. Unfortunately since the staff members at the digitizing and image service have only enough knowledge to be able to interact with computerized systems as users, this ideal solution would not work given the current context.

A possible solution to this problem was the implementation of a PURL (see section 2.4.1.B) system similar to the already existing PURL.pt [2], which is currently in use at the BND. This sort of system would be able not only to manage the publishing of new works, but also generate the necessary indexes that feature at Hemeroteca's website.

This solution was initially explored but later abandoned for it was found that the bibliographic records were not attached to the digitized works. That implied the development of a complex application to retrieve bibliographic records to be attached to the existing PURL system. This fact proved to be the tipping point between what would have been a viable solution and what would prove to be too much effort for a marginal gain. Additionally the ever pending constraint of the staff members at the digitizing and image service not being able to manage complex computerized systems, meant that this solution as intelligent as it was would not be functional.

Therefore a compromise had to be made in terms of the adopted solution. Records would be stored not on a relational database but on Microsoft Exel XLS file which would serve as a *de facto* authority file (see section 2.1.1). This of course would bring about some concurrency issues. In case two distinct staff members happen to edit the XLS file at the same time, conflicts might arise. These issues had to be contemplated in the optimized P1.4.

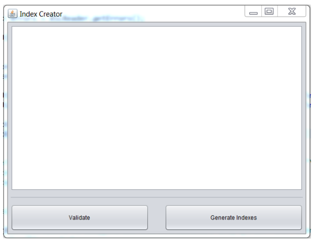
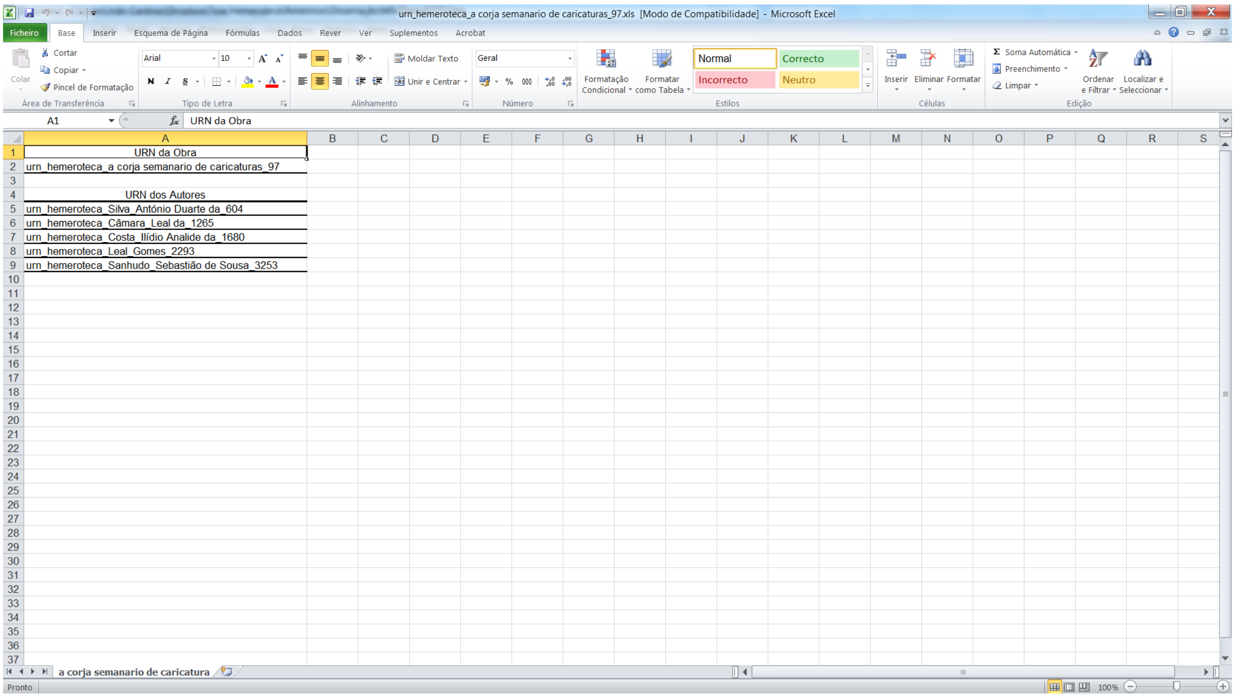


Figure 4.9: Index creator application

The use of an XLS file as a *de facto* authority file would be the starting point for automating the index creation process (see figure 4.11). For a centralized repository holding all the information, meant that indexes could be easily generated. All the staff members needed to do was to create a XLS collaboration file for each newly digitized

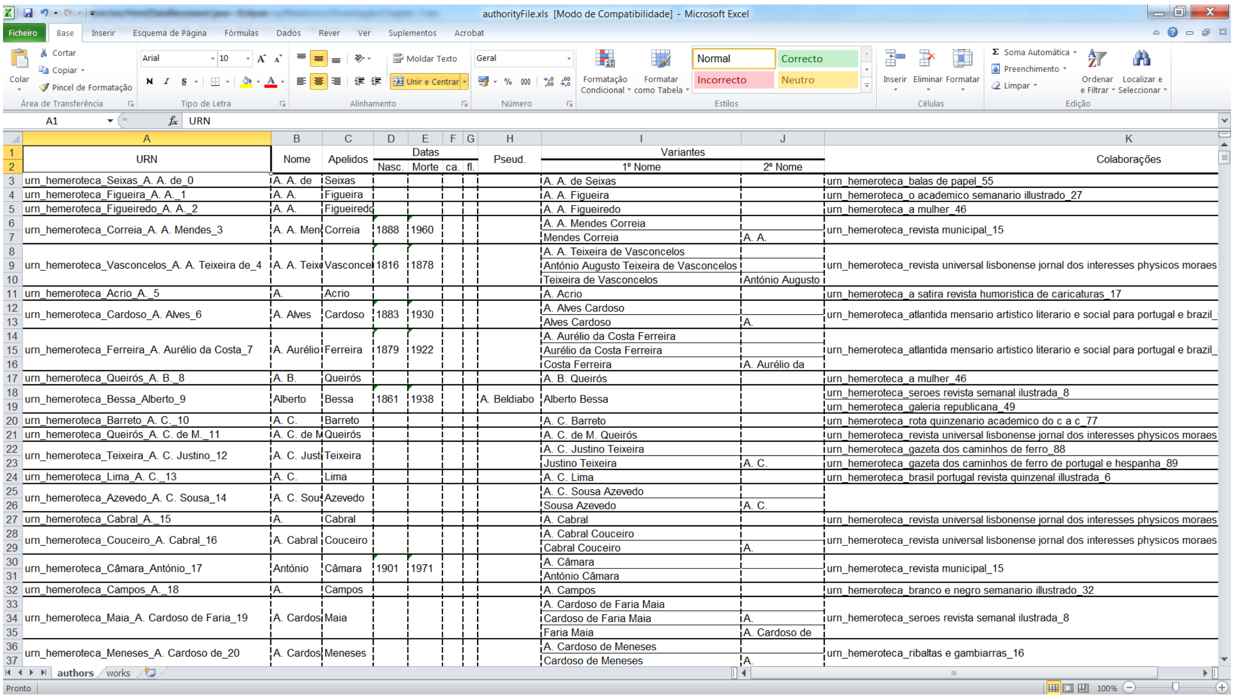
[2] PURL.pt webpage (2013-01-03): `http://purl.pt/index/geral/PT/index.html` Retrieved 2013-01-03

work, indicating which authors collaborated in that particular work (see figure 4.10). The index generation was to be automatically performed through an index creator application developed using the Java programming language (see figure 4.9). This application would parse all the existing XLS collaboration files and update all the necessary entries at the XLS authority file, as well as generating all the necessary HTML files that made up the indexes. The idea was to keep staff members at the digitizing and image service from having to manually edit HTML files. This solution would not only solve all three of the issues mentioned in section 3.3, but also add value to the sub-process.



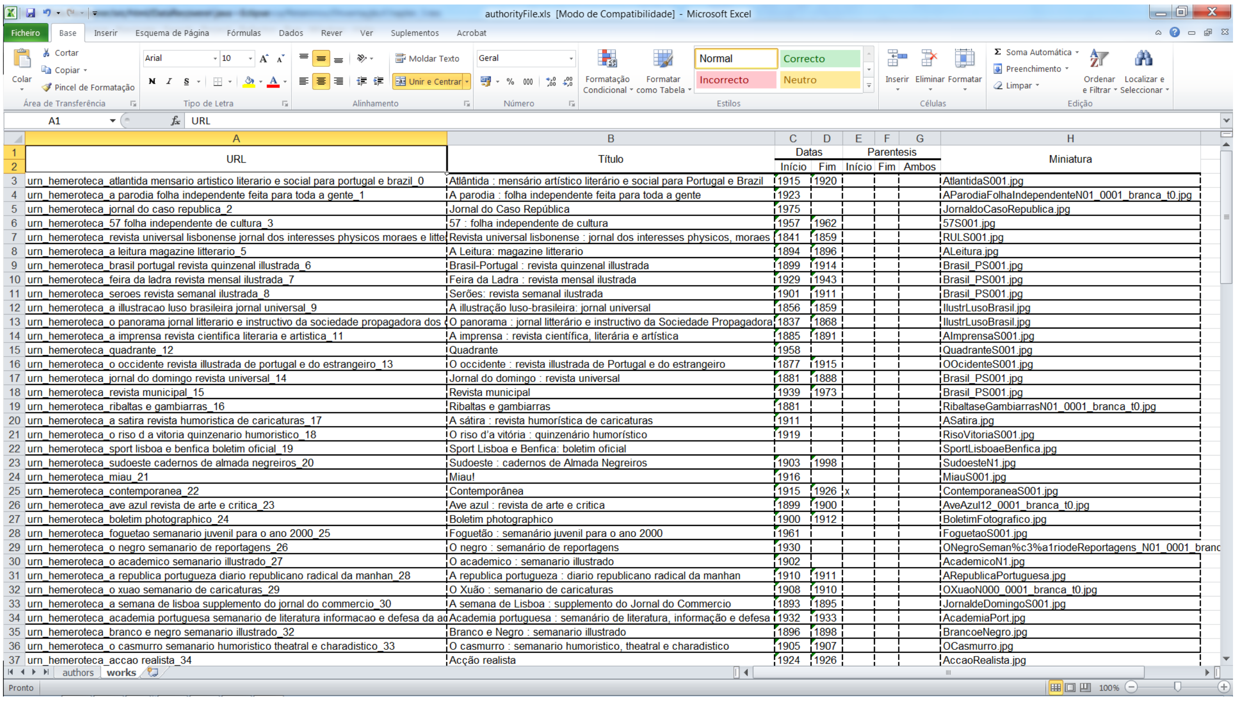
	A
1	URN da Obra
2	urn_hemeroteca_a corja semanario de caricaturas_97
3	
4	URN dos Autores
5	urn_hemeroteca_Silva_António Duarte da_604
6	urn_hemeroteca_Câmara_Leal da_1265
7	urn_hemeroteca_Costa_Ilídio Analide da_1680
8	urn_hemeroteca_Leal_Gomes_2293
9	urn_hemeroteca_Sanhudo_Sebastião de Sousa_3253

Figure 4.10: XLS collaboration file



	A	B	C	D	E	F	G	H	I	J	K
1	URN	Nome	Apelidos	Datas				Pseud.	Variantes		Colaborações
2				Nasc.	Morte	ca.	fl.		1º Nome	2º Nome	
3	urn_hemeroteca_Seixas_A. A. de_0	A. A. de	Seixas						A. A. de Seixas		urn_hemeroteca_balas de papel_55
4	urn_hemeroteca_Figueira_A. A._1	A. A.	Figueira						A. A. Figueira		urn_hemeroteca_o academico semanario illustrado_27
5	urn_hemeroteca_Figueiredo_A. A._2	A. A.	Figueiredo						A. A. Figueiredo		urn_hemeroteca_a mulher_46
6	urn_hemeroteca_Correia_A. A. Mendes_3	A. A. Men	Correia	1888	1960				A. A. Mendes Correia		urn_hemeroteca_revista municipal_15
7									Mendes Correia	A. A.	
8	urn_hemeroteca_Vasconcelos_A. A. Teixeira de_4	A. A. Teixe	Vasconcel	1816	1878				A. A. Teixeira de Vasconcelos		urn_hemeroteca_revista universal lisbonense jornal dos interesses physicos moraes
9									António Augusto Teixeira de Vasconcelos		
10									Teixeira de Vasconcelos	António Augusto	
11	urn_hemeroteca_Acrio_A._5	A.	Acrio						A. Acrio		urn_hemeroteca_a satira revista humoristica de caricaturas_17
12	urn_hemeroteca_Cardoso_A. Alves_6	A. Alves	Cardoso	1883	1930				A. Alves Cardoso		urn_hemeroteca_atlantida mensario artistico literario e social para portugal e brazil_
13									Alves Cardoso	A.	
14	urn_hemeroteca_Ferreira_A. Aurélio da Costa_7	A. Aurélio	Ferreira	1879	1922				A. Aurélio da Costa Ferreira		urn_hemeroteca_atlantida mensario artistico literario e social para portugal e brazil_
15									Aurélio da Costa Ferreira		
16									Costa Ferreira	A. Aurélio da	
17	urn_hemeroteca_Queirós_A. B._8	A. B.	Queirós						A. B. Queirós		urn_hemeroteca_a mulher_46
18	urn_hemeroteca_Bessa_Alberto_9	Alberto	Bessa	1861	1938			A. Beldiabo	Alberto Bessa		urn_hemeroteca_seroes revista semanal ilustrada_8
19											urn_hemeroteca_galeria republicana_49
20	urn_hemeroteca_Barreto_A. C._10	A. C.	Barreto						A. C. Barreto		urn_hemeroteca_rota quinzenario academico do c a c_77
21	urn_hemeroteca_Queirós_A. C. de M._11	A. C. de M	Queirós						A. C. de M. Queirós		urn_hemeroteca_revista universal lisbonense jornal dos interesses physicos moraes
22	urn_hemeroteca_Teixeira_A. C. Justino_12	A. C. Justi	Teixeira						A. C. Justino Teixeira		urn_hemeroteca_gazeta dos caminhos de ferro_88
23									Justino Teixeira	A. C.	urn_hemeroteca_gazeta dos caminhos de ferro de portugal e hespanha_89
24	urn_hemeroteca_Lima_A. C._13	A. C.	Lima						A. C. Lima		urn_hemeroteca_brasil portugal revista quinzenal illustrada_6
25	urn_hemeroteca_Azevedo_A. C. Sousa_14	A. C. Sous	Azevedo						A. C. Sousa Azevedo		
26									Sousa Azevedo	A. C.	
27	urn_hemeroteca_Cabral_A._15	A.	Cabral						A. Cabral		urn_hemeroteca_revista universal lisbonense jornal dos interesses physicos moraes
28	urn_hemeroteca_Couceiro_A. Cabral_16	A. Cabral	Couceiro						A. Cabral Couceiro		urn_hemeroteca_revista universal lisbonense jornal dos interesses physicos moraes
29									Cabral Couceiro	A.	
30	urn_hemeroteca_Câmara_António_17	António	Câmara	1901	1971				A. Câmara		urn_hemeroteca_revista municipal_15
31									António Câmara		
32	urn_hemeroteca_Campos_A._18	A.	Campos						A. Campos		urn_hemeroteca_branco e negro semanario illustrado_32
33	urn_hemeroteca_Maia_A. Cardoso de Faria_19	A. Cardos	Maia						A. Cardoso de Faria Maia		urn_hemeroteca_seroes revista semanal ilustrada_8
34									Cardoso de Faria Maia	A.	
35									Faria Maia	A. Cardoso de	
36	urn_hemeroteca_Meneses_A. Cardoso de_20	A. Cardos	Meneses						A. Cardoso de Meneses		urn_hemeroteca_ribaltas e gambiarras_16
37									Cardoso de Meneses	A.	

(a) The authors sheet



	A	B	C	D	E	F	G	H
1	URL	Título	Datas		Parentesis			Miniatura
2			Início	Fim	Início	Fim	Ambos	
3	urn_hemeroteca_atlantida mensario artistico literario e social para portugal e brazil_0	Atlântida : mensário artístico literário e social para Portugal e Brazil	1915	1920				AtlantidaS001.jpg
4	urn_hemeroteca_a parodia folha independente feita para toda a gente_1	A parodia : folha independente feita para toda a gente	1923					AParodiaFolhaIndependenteN01_0001_branca_t0.jpg
5	urn_hemeroteca_jornal do caso republica_2	Jornal do Caso República	1975					JornaldoCasoRepublica.jpg
6	urn_hemeroteca_57 folha independente de cultura_3	57 : folha independente de cultura	1957	1962				57S001.jpg
7	urn_hemeroteca_revista universal lisbonense jornal dos interesses physicos moraes e litte	Revista universal lisbonense : jornal dos interesses physicos, moraes	1841	1859				RULS001.jpg
8	urn_hemeroteca_a leitura magazine litterario_5	A Leitura: magazine litterario	1894	1896				ALeitura.jpg
9	urn_hemeroteca_brasil portugal revista quinzenal illustrada_6	Brasil-Portugal : revista quinzenal illustrada	1899	1914				Brasil_PS001.jpg
10	urn_hemeroteca_feira da ladra revista mensal ilustrada_7	Feira da Ladra : revista mensal ilustrada	1929	1943				Brasil_PS001.jpg
11	urn_hemeroteca_seroes revista semanal ilustrada_8	Serões: revista semanal ilustrada	1901	1911				Brasil_PS001.jpg
12	urn_hemeroteca_a illustracao luso brasileira jornal universal_9	A illustração luso-brasileira: jornal universal	1856	1859				IlustrLusoBrasil.jpg
13	urn_hemeroteca_o panorama jornal litterario e instructivo da sociedade propagadora dos	O panorama : jornal litterário e instructivo da Sociedade Propagadora	1837	1868				IlustrLusoBrasil.jpg
14	urn_hemeroteca_a imprensa revista cientifica literaria e artistica_11	A imprensa : revista científica, literária e artística	1885	1891				AImprensaS001.jpg
15	urn_hemeroteca_quadrante_12	Quadrante	1958					QuadranteS001.jpg
16	urn_hemeroteca_o occidente revista illustrada de portugal e do estrangeiro_13	O occidente : revista illustrada de Portugal e do estrangeiro	1877	1915				OOcidenteS001.jpg
17	urn_hemeroteca_jornal do domingo revista universal_14	Jornal do domingo : revista universal	1881	1888				Brasil_PS001.jpg
18	urn_hemeroteca_revista municipal_15	Revista municipal	1939	1973				Brasil_PS001.jpg
19	urn_hemeroteca_ribaltas e gambiarras_16	Ribaltas e gambiarras	1881					RibaltaseGambiarrasN01_0001_branca_t0.jpg
20	urn_hemeroteca_a satira revista humoristica de caricaturas_17	A sátira : revista humorística de caricaturas	1911					ASatira.jpg
21	urn_hemeroteca_o riso d a vitoria quinzenario humoristico_18	O riso d'a vitória : quinzenário humorístico	1919					RisoVitoriaS001.jpg
22	urn_hemeroteca_sport lisboa e benfica boletim oficial_19	Sport Lisboa e Benfica: boletim oficial						SportLisboaeBenfica.jpg
23	urn_hemeroteca_sudoeste cadernos de almada negreiros_20	Sudoeste : cadernos de Almada Negreiros	1903	1998				SudoesteN1.jpg
24	urn_hemeroteca_miau_21	Miau!	1916					MiauS001.jpg
25	urn_hemeroteca_contemporanea_22	Contemporânea	1915	1926	x			ContemporaneaS001.jpg
26	urn_hemeroteca_ave azul revista de arte e critica_23	Ave azul : revista de arte e critica	1899	1900				AveAzul12_0001_branca_t0.jpg
27	urn_hemeroteca_boletim photographico_24	Boletim photographico	1900	1912				BoletimFotografico.jpg
28	urn_hemeroteca_foguetao semanario juvenil para o ano 2000_25	Foguetão : semanário juvenil para o ano 2000	1961					FoguetaoS001.jpg
29	urn_hemeroteca_o negro semanario de reportagens_26	O negro : semanário de reportagens	1930					ONegroSeman%c3%a1riodeReportagens_N01_0001_branc
30	urn_hemeroteca_o academico semanario illustrado_27	O academico : semanario illustrado	1902					AcademicoN1.jpg
31	urn_hemeroteca_a republica portugueza diario republicano radical da manhan_28	A republica portugueza : diario republicano radical da manhan	1910	1911				ARepublicaPortuguesa.jpg
32	urn_hemeroteca_o xuao semanario de caricaturas_29	O Xuão : semanario de caricaturas	1908	1910				OXuaoN000_0001_branca_t0.jpg
33	urn_hemeroteca_a semana de lisboa supplemento do jornal do commercio_30	A semana de Lisboa : supplemento do Jornal do Commercio	1893	1895				JornaldeDomingoS001.jpg
34	urn_hemeroteca_academia portuguesa semanario de literatura informacao e defesa da ad	Academia portuguesa : semanário de literatura, informação e defesa	1932	1933				AcademiaPort.jpg
35	urn_hemeroteca_branco e negro semanario illustrado_32	Branco e Negro : semanario illustrado	1896	1898				BrancoeNegro.jpg
36	urn_hemeroteca_o casmurro semanario humoristico theatral e charadistico_33	O casmurro : semanario humoristico, theatral e charadistico	1905	1907				OCasmurro.jpg
37	urn_hemeroteca_accao realista_34	Acção realista	1924	1926				AccaoRealista.jpg

(b) The works sheet

Figure 4.11: XLS authority file displaying both the authors and works sheet

4.3.1 Optimized Sub-Process Publishing (P1.4)

Considering the adopted solution, the optimized P1.4 would be composed of three tasks, as can be seen in figure 4.12.

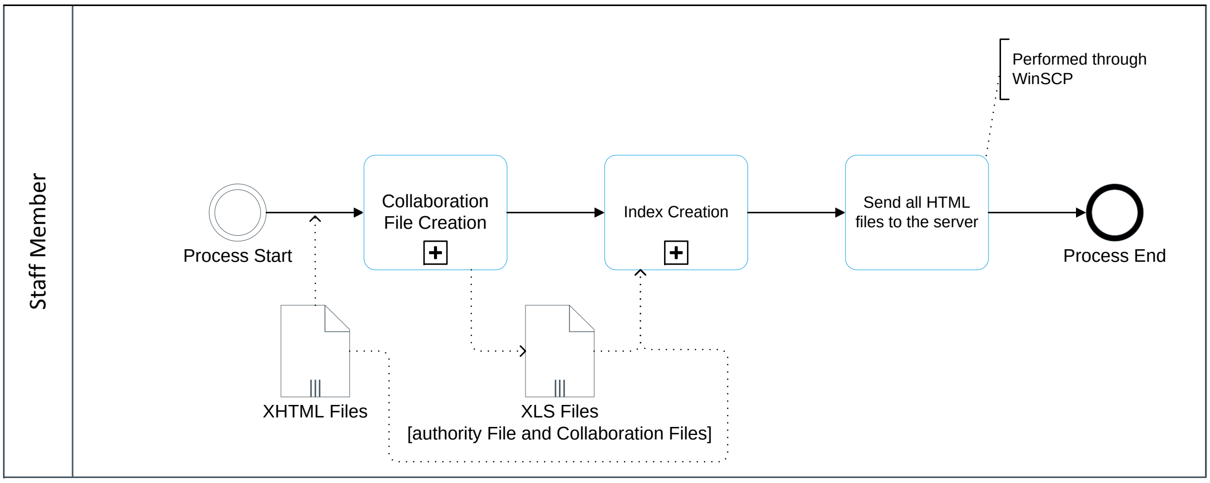

Staff Member
Performed through WinSCP
Process Start
Collaboration File Creation
Index Creation
Send all HTML files to the server
Process End
XHTML Files
XLS Files
[authority File and Collaboration Files]

Figure 4.12: The optimized Sub-Process Publishing (P1.4)

The collaboration file creation task is a sub-process on its own (see figure 4.13). It starts by asking the staff member to verify whether the work to be added already exists at the XLS authority file. If it does not, then the first task is to place the work's thumbnail in the correct folder, followed by the creation of a new work entry at the XLS authority file. Regardless, the next task is to compile a list of authors which collaborated in the work that is being processed. The next step is for the staff member to verify whether all the listed authors are present at the XLS authority file, adding them if not present. The last task is the creation a new XLS collaboration file containing the work identifier, and identifiers for all the collaborating authors.

The second task refers to the actual P1.4.2. Its first task is to validate the index creation, this means running the index creation application and performing a check run looking for possible errors in both the XLS authority file, and all the XLS collaboration files. If errors are found, then they must be corrected and the validation task must be performed again. Once the validation shows no errors, the index creation task is executed and all the HTML files are created.

The last task of the optimized P1.4 to be performed is sending all the HTML files which were created to the remote server which is managed by the DMSI-DASI. A task which is accomplished by using a safe connection through the SFTP, SCP and FTP client for Microsoft Windows WinSCP (see section 3.1).

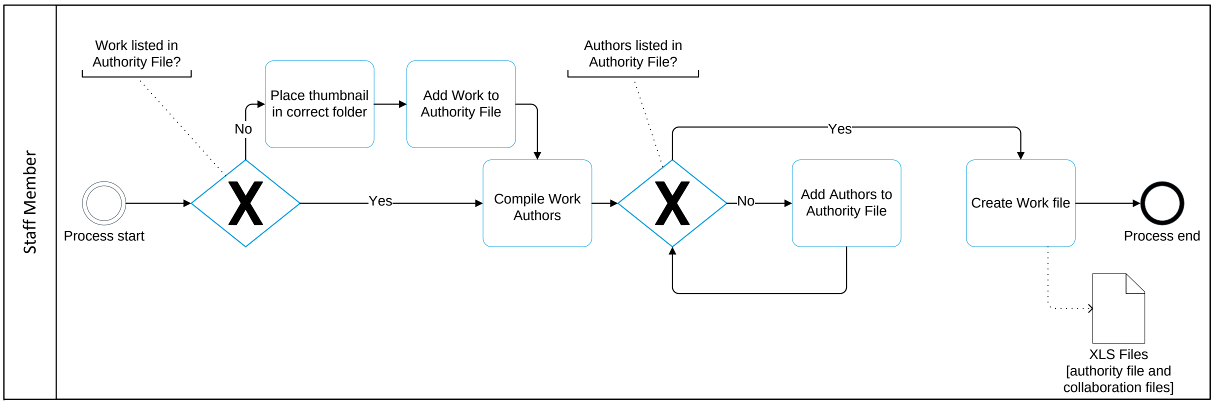

Staff Member
Work listed in Authority File?
Authors listed in Authority File?
Process start
No
Yes
Place thumbnail in correct folder
Add Work to Authority File
Compile Work Authors
Yes
No
Add Authors to Authority File
Create Work file
Process end
XLS Files [authority file and collaboration files]

Figure 4.13: The Sub-Process Collaboration File Creation (P1.4.1)

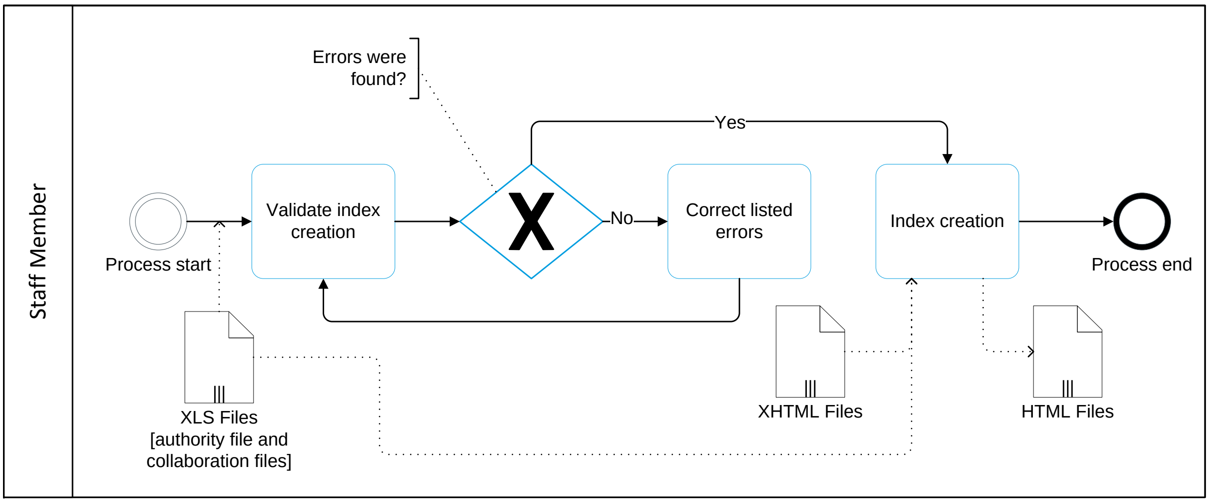


Figure 4.14: The Sub-Process Index Creation (P1.4.2)

4.3.2 Software Applications

There were two software applications that had to be developed for the optimization of the P1.4.

The first software application was simply a data recoverer which was meant to recover all data stored at the currently existing indexes, and generate new ones based on the sub-process tasks previously described. The development of this application was a lengthy process due to the lack of format consistency of the existing HTML index pages.

The second software application to be developed was intended to be instated permanently within the publishing sub-process as an index creator application. It has two use cases, as can be seen in figure 4.15.

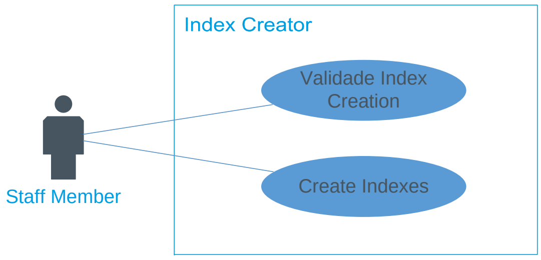


Figure 4.15: Index creator application use cases

- **Validate index creation** Users may perform a validation on the artefacts needed to create indexes (XLS authority file and XLS collaboration files), in order to check if there are no errors that would impede the indexes from being created;

- **Create indexes** Users may create indexes. An operation which generates all the necessary HTML index files.

The index creator software application was developed using the Java programming language, using a multi-tier architecture, just as was done in the case of the Carica software application. The application is structured in three tiers, which can be seen in figure 4.7.

The data access tier holds the classes that implement domain classes for the application, such as authors and works. The service tier classes on the other hand interact with the data access tier, by using its classes to implement the logic behind the two functionalities the index creator application provides. Finally the presentation tier classes implement a graphical user interface with which the user can interact to both validate index creation and to create indexes.

5
Validation

This chapter aims to present the results of the validation to the two software applications that were developed as a result of the solutions implemented during this project.

Due to delays during the execution of the project it proved to be impossible to validate every functionality in a real world context. However this did not impede that tests from being made to ensure that all functionalities performed according to the solution design.

This chapter is organized in two sections. In section 5.1 the validation of each of the eight use cases from the Carica application are presented. Whilst in section 5.2 the two use cases from the index creator application are presented.

5.1 Carica Application Validation

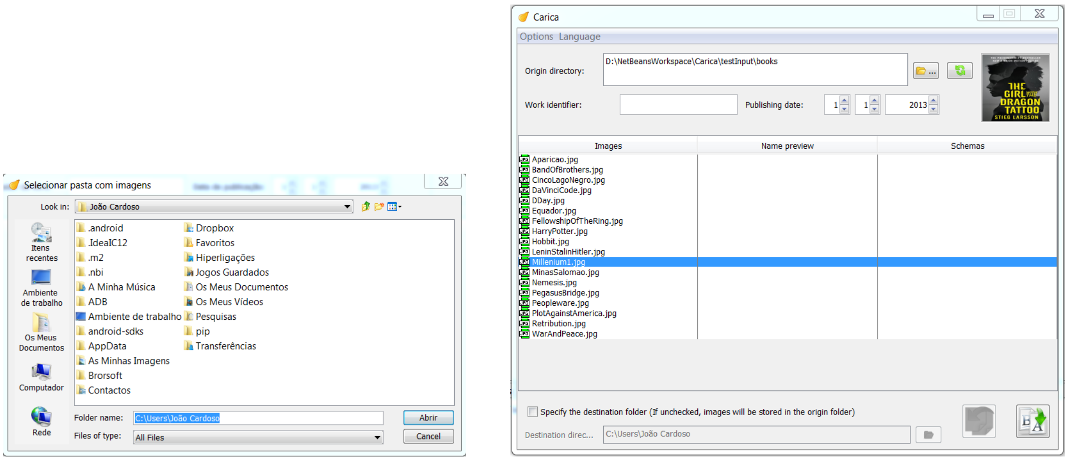
This section presents the tests that were used to validate each of the four use cases found in the Carica application. The four use cases are described in section 4.1.1.

5.1.1 Add Image Files

The first use case to be validated was the addition of a batch of image files to the Carica application. Users should be able to add image files to Carica, and once added the user should be able to remove them as well. To test this use case, a batch of image files was added to Carica, and subsequently removed.

The test procedure was simply adding a batch of image files by clicking the 📂... (browse) button, which lead to the opening of a browse dialog such as the one seen in figure 5.1(a). Once the image folder was selected the images were added to Carica as can be seen in figure 5.1(b). This ensured that images were being added correctly.

The second part of the test required a batch of image files to be removed from Carica. This implied selecting the required images files and selecting the "remove image" option in the Carica menu, as can be seen in figure 5.2(b). Once this option was selected the images were removed, as can be seen in figure 5.2(b), thus validating that images could be removed as was specified in the use case.



Look in: João Cardoso

.android
.IdealC12
.m2
.nbi
A Minha Música
ADB
Ambiente de trabalho
android-sdks
AppData
As Minhas Imagens
Brorsoft
Contactos
Dropbox
Favoritos
Hiperligações
Jogos Guardados
Os Meus Documentos
Os Meus Vídeos
Pesquisas
pip
Transferências

Folder name: C:\Users\João Cardoso

Files of type: All Files

Origin directory: D:\NetBeansWorkspace\Carica\testInput\books

Work identifier:

Publishing date: 1 1 2013

Images	Name preview	Schemas
Aparicao.jpg		
BandOfBrothers.jpg		
CincoLagoNegro.jpg		
DaVinciCode.jpg		
DDay.jpg		
Equador.jpg		
FellowshipOfTheRing.jpg		
HarryPotter.jpg		
Hobbit.jpg		
LeninStalinHitler.jpg		
Millenium1.jpg		
MinasSalomao.jpg		
Nemesis.jpg		
PegasusBridge.jpg		
Peopleware.jpg		
PlotAgainstAmerica.jpg		
Retribution.jpg		
WarAndPeace.jpg		

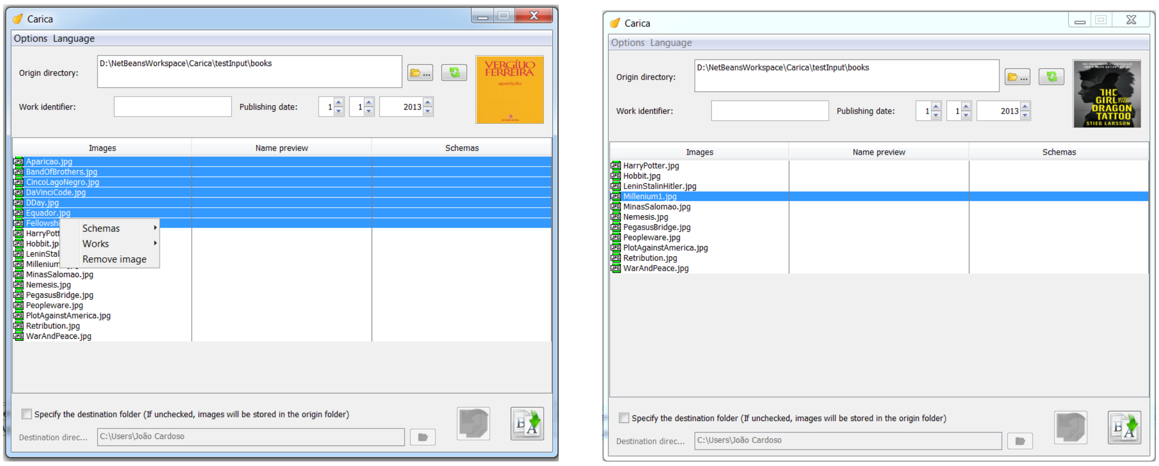
□ Specify the destination folder (If unchecked, images will be stored in the origin folder)

Destination direc... C:\Users\João Cardoso

(a) Selecting the image files to add to Carica

(b) Adding image files to Carica

Figure 5.1: Adding a batch of image files to Carica



Origin directory: D:\NetBeansWorkspace\Carica\testInput\books

Work identifier:

Publishing date: 1 1 2013

Images	Name preview	Schemas
Aparicao.jpg		
BandOfBrothers.jpg		
CincoLagoNegro.jpg		
DaVinciCode.jpg		
DDay.jpg		
Equador.jpg		
Fellowshi		
HarryPott		
Hobbit.jp		
LeninStal		
Millenium		
MinasSalomao.jpg		
Nemesis.jpg		
PegasusBridge.jpg		
Peopleware.jpg		
PlotAgainstAmerica.jpg		
Retribution.jpg		
WarAndPeace.jpg		

□ Specify the destination folder (If unchecked, images will be stored in the origin folder)

Destination direc... C:\Users\João Cardoso

(a) Removing a batch of image files from Carica

Origin directory: D:\NetBeansWorkspace\Carica\testInput\books

Work identifier:

Publishing date: 1 1 2013

Images	Name preview	Schemas
HarryPotter.jpg		
Hobbit.jpg		
LeninStalinHitler.jpg		
Millenium1.jpg		
MinasSalomao.jpg		
Nemesis.jpg		
PegasusBridge.jpg		
Peopleware.jpg		
PlotAgainstAmerica.jpg		
Retribution.jpg		
WarAndPeace.jpg		

□ Specify the destination folder (If unchecked, images will be stored in the origin folder)

Destination direc... C:\Users\João Cardoso

(b) Carica after the removal of a batch of image files

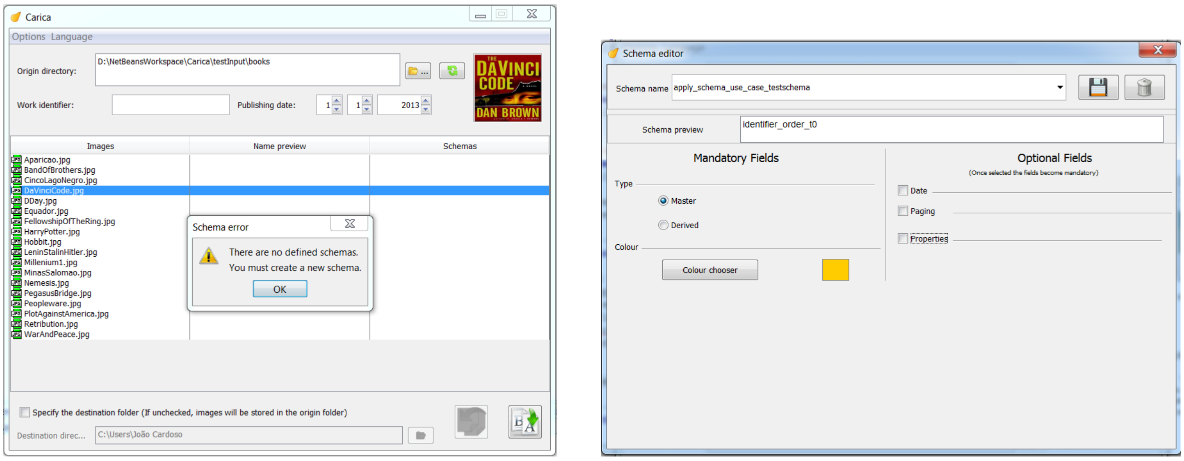
Figure 5.2: Removing a batch of image files from Carica

5.1.2 Apply Schema

The second use case to be validated was the application of a schema. Users should be able to apply a schema to a batch of image files. This implies having created a schema or edited an existing schema, as was specified in the UML activity schema shown in figure 4.3, which is presented in section 4.1.1. In order to test this use case there were three tasks that needed to be checked. Firstly the fact that no schemas could be applied if there were no schemas defined was to be verified. Secondly in order to be able to apply a schema, a new schema was to be created. And finally the newly created schema was to be applied to a batch of image files.

To verify the first step of the procedure all that was required was to attempt to apply a schema. Therefore a batch of image files was selected and the Carica menu was called, prompting an error message that warned the user to the need of creating a new schema, as can be seen in figure 5.3(a). To complete the second step of the test procedure, a new schema was created by filling only the mandatory fields in the schema edit dialog, as can be seen in figure 5.3(b). The third and last implied selecting a batch of image files and then choosing the previously created schema from the Carica menu, in order for it to be applied to the selected batch of image files (see figure 5.4(a)).

The results of the schema application, which validate the use case, can be seen in figure 5.4(b).



Origin directory: D:\NetBeansWorkspace\Carica\testInput\books

Publishing date: 1 1 2013

Images: Aparicao.jpg, BandOfBrothers.jpg, CincoLagoNegro.jpg, DaVinciCode.jpg, DDay.jpg, Equador.jpg, FellowshipOfTheRing.jpg, HarryPotter.jpg, Hobbit.jpg, LeninStalinHitler.jpg, Millenium1.jpg, MinasSalomao.jpg, Nemesis.jpg, PegasusBridge.jpg, Peopleware.jpg, PlotAgainstAmerica.jpg, Retribution.jpg, WarAndPeace.jpg

There are no defined schemas.
You must create a new schema.

□ Specify the destination folder (If unchecked, images will be stored in the origin folder)

Destination direc... C:\Users\João Cardoso

Schema name apply_schema_use_case_testschema

Schema preview identifier_order_t0

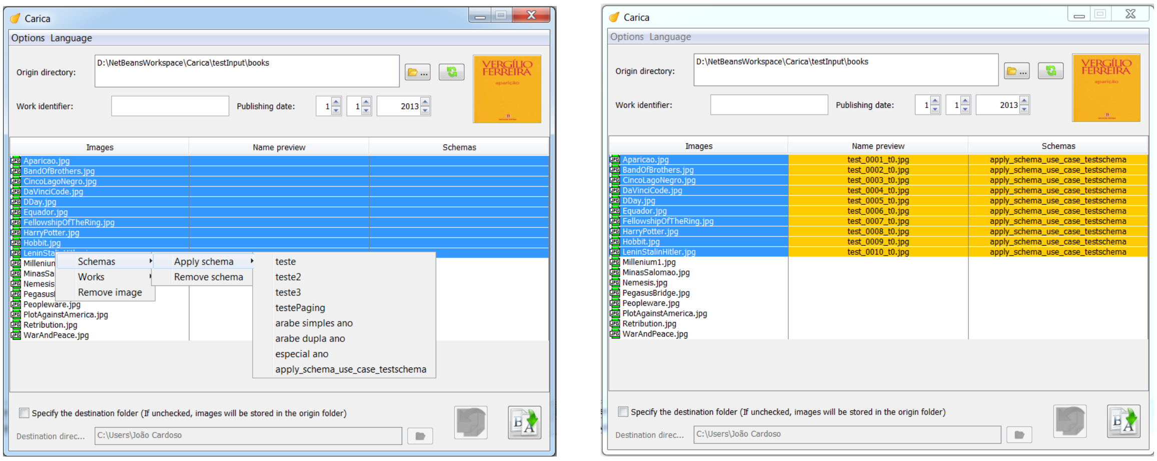
Type: ● Master ○ Derived

□ Date
□ Paging
□ Properties

(a) The error message that is presented when there are no existing schemas

(b) Creating a new schema using the Carica schema edit dialog

Figure 5.3: Error message warning to the lack of existing schemas and schema creation



Origin directory: D:\NetBeansWorkspace\Carica\testInput\books

Publishing date: 1 1 2013

Apply schema: teste, teste2, teste3, testePaging, arabe simples ano, arabe dupla ano, especial ano, apply_schema_use_case_testschema

Images	Name preview	Schemas
Aparicao.jpg	test_0001_t0.jpg	apply_schema_use_case_testschema
BandOfBrothers.jpg	test_0002_t0.jpg	apply_schema_use_case_testschema
CincoLagoNegro.jpg	test_0003_t0.jpg	apply_schema_use_case_testschema
DaVinciCode.jpg	test_0004_t0.jpg	apply_schema_use_case_testschema
DDay.jpg	test_0005_t0.jpg	apply_schema_use_case_testschema
Equador.jpg	test_0006_t0.jpg	apply_schema_use_case_testschema
FellowshipOfTheRing.jpg	test_0007_t0.jpg	apply_schema_use_case_testschema
HarryPotter.jpg	test_0008_t0.jpg	apply_schema_use_case_testschema
Hobbit.jpg	test_0009_t0.jpg	apply_schema_use_case_testschema
LeninStalinHitler.jpg	test_0010_t0.jpg	apply_schema_use_case_testschema
Millenium1.jpg		
MinasSalomao.jpg		
Nemesis.jpg		
PegasusBridge.jpg		
Peopleware.jpg		
PlotAgainstAmerica.jpg		
Retribution.jpg		
WarAndPeace.jpg		

□ Specify the destination folder (If unchecked, images will be stored in the origin folder)

Destination direc... C:\Users\João Cardoso

(a) Application of a schema to a batch of image files

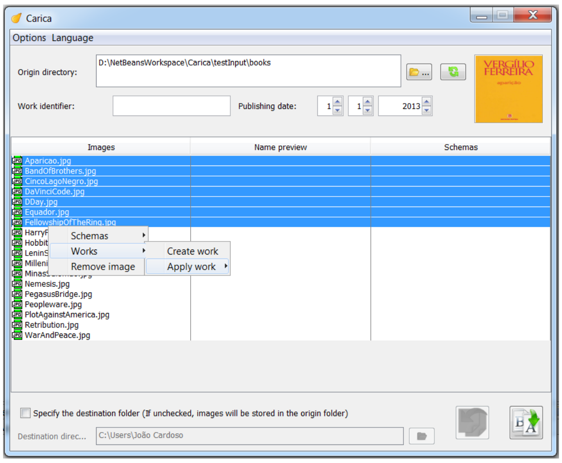
(b) Carica after the application of a schema to a batch of image files

Figure 5.4: Applying a schema to a batch of image files

5.1.3 Apply Work

The third use case to be validated was the application of a work. As stated in section 4.1.1, users should be able to apply a work to a batch of image files. This application implies having previously created or edited an existing work, as was specified in the UML activity schema shown in figure 4.6(b), which is presented in section 4.1.1. As some of the tasks that were required for the testing of this use case had already been tested in section 5.1.2, the tests presented in this section are related to tasks which haven't been previously tested. As such, the tasks to be checked were firstly the validation that work application could not be executed without having previously created a work. Secondly, the creation of a work had to be validated. And thirdly, that works could indeed be applied to a batch of image files.

To validate that no work application could be executed without having previously created a work, a batch of image files was selected and the Carica menu was called in order to apply a work. However, since no work had yet been created, no work was available for application, thus validating the task as can be seen in figure 5.5. The second task to validate was the creation of a work. In order to create a work, two schemas were applied to a batch of image files. Subsequently the work was created by selecting all the files to which schemas had been applied and selecting the option "create work" in the Carica menu, as can be seen in figure 5.6. Finally the last task to validate was the application of a work to a batch of image files. This was done by selecting a batch of image files and selecting a previously created work from the Carica menu, as can be seen in figure 5.7(a). The final outcome of this task can be seen in figure 5.7(b).



Origin directory: D:\NetBeansWorkspace\Carica\testInput\books

Work identifier:

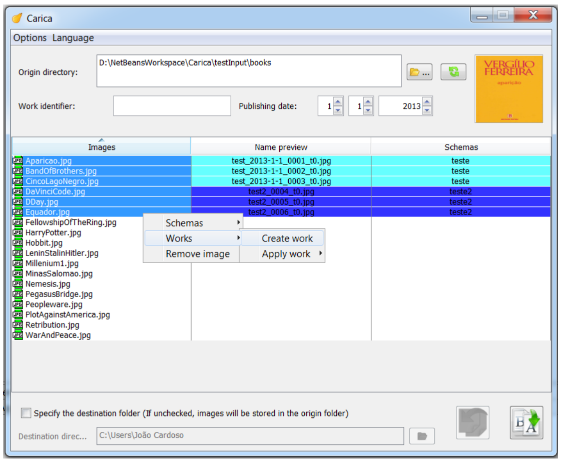
Publishing date: 1 1 2013

Images	Name preview	Schemas
Aparicao.jpg		
BandOfBrothers.jpg		
CincoLagoNegro.jpg		
DaVinciCode.jpg		
DDay.jpg		
Equador.jpg		
FellowshipOfTheRing.jpg		
HarryP		
Hobbit		
LeninS		
Milleni		
Minas		
Nemesis.jpg		
PegasusBridge.jpg		
Peopleware.jpg		
PlotAgainstAmerica.jpg		
Retribution.jpg		
WarAndPeace.jpg		

☐ Specify the destination folder (If unchecked, images will be stored in the origin folder)

Destination direc... C:\Users\João Cardoso

Figure 5.5: Carica menu with no works available for application



Origin directory: D:\NetBeansWorkspace\Carica\testInput\books

Work identifier:

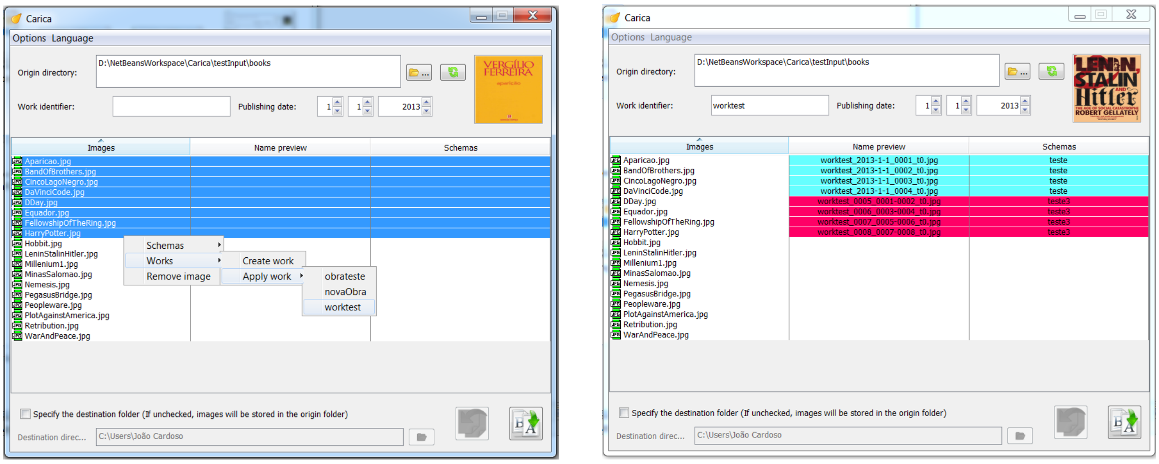
Publishing date: 1 1 2013

Images	Name preview	Schemas
Aparicao.jpg	test_2013-1-1_0001_t0.jpg	teste
BandOfBrothers.jpg	test_2013-1-1_0002_t0.jpg	teste
CincoLagoNegro.jpg	test_2013-1-1_0003_t0.jpg	teste
DaVinciCode.jpg	test2_0004_t0.jpg	teste2
DDay.jpg	test2_0005_t0.jpg	teste2
Equador.jpg	test2_0006_t0.jpg	teste2
FellowshipOfTheRing.jpg		
HarryPotter.jpg		
Hobbit.jpg		
LeninStalinHitler.jpg		
Millenium1.jpg		
MinasSalomao.jpg		
Nemesis.jpg		
PegasusBridge.jpg		
Peopleware.jpg		
PlotAgainstAmerica.jpg		
Retribution.jpg		
WarAndPeace.jpg		

☐ Specify the destination folder (If unchecked, images will be stored in the origin folder)

Destination direc... C:\Users\João Cardoso

Figure 5.6: The creation of a work



Origin directory: D:\NetBeansWorkspace\Carica\testInput\books

Work identifier:

Publishing date: 1 1 2013

Images	Name preview	Schemas
Aparicao.jpg		
BandOfBrothers.jpg		
CincoLagoNegro.jpg		
DaVinciCode.jpg		
DDay.jpg		
Equador.jpg		
FellowshipOfTheRing.jpg		
HarryPotter.jpg		
Hobbit.jpg		
LeninStalinHitler.jpg		
Millenium1.jpg		
MinasSalomao.jpg		
Nemesis.jpg		
PegasusBridge.jpg		
Peopleware.jpg		
PlotAgainstAmerica.jpg		
Retribution.jpg		
WarAndPeace.jpg		

☐ Specify the destination folder (If unchecked, images will be stored in the origin folder)

Destination direc... C:\Users\João Cardoso

(a) Choosing a work to apply from the Carica menu

Origin directory: D:\NetBeansWorkspace\Carica\testInput\books

Work identifier: worktest

Publishing date: 1 1 2013

Images	Name preview	Schemas
Aparicao.jpg	worktest_2013-1-1_0001_t0.jpg	teste
BandOfBrothers.jpg	worktest_2013-1-1_0002_t0.jpg	teste
CincoLagoNegro.jpg	worktest_2013-1-1_0003_t0.jpg	teste
DaVinciCode.jpg	worktest_2013-1-1_0004_t0.jpg	teste
DDay.jpg	worktest_0005_0001-0002_t0.jpg	teste3
Equador.jpg	worktest_0006_0003-0004_t0.jpg	teste3
FellowshipOfTheRing.jpg	worktest_0007_0005-0006_t0.jpg	teste3
HarryPotter.jpg	worktest_0008_0007-0008_t0.jpg	teste3
Hobbit.jpg		
LeninStalinHitler.jpg		
Millenium1.jpg		
MinasSalomao.jpg		
Nemesis.jpg		
PegasusBridge.jpg		
Peopleware.jpg		
PlotAgainstAmerica.jpg		
Retribution.jpg		
WarAndPeace.jpg		

☐ Specify the destination folder (If unchecked, images will be stored in the origin folder)

Destination direc... C:\Users\João Cardoso

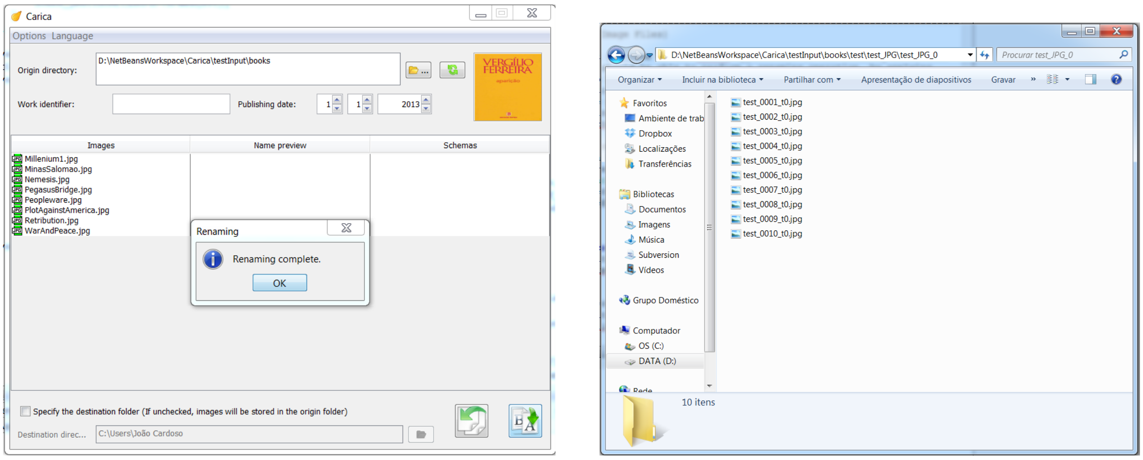
(b) The batch of images files after the work has been applied

Figure 5.7: Applying a work to a batch of image files

5.1.4 Rename Image Files

The fourth and last use case of the Carica application to be validated was the renaming of batches image files. Users should be able to perform a renaming operation, by applying schemas or works to batches of image files. In order to validate this use case, the procedure was to apply a schema to a batch of image files, and then to check that the image file names would be changed according to the schema which was applied. However since the first step has already been validated in section 5.1.2, this section focuses solely on the second step.

As such the second step required having already applied a schema to a batch of image files, as can be seen in figure 5.4(b). Once the schema had been applied, the renaming operation was executed by clicking on the BA (rename) button. This causes the batch of renamed image files to both be removed from Carica, and for the images to be renamed, as can be seen in both figure 5.8(a) and figure 5.8(b).



Origin directory: D:\NetBeansWorkspace\Carica\testInput\books

Work identifier:

Publishing date: 1 1 2013

Millenium1.jpg
MinasSalomao.jpg
Nemesis.jpg
PegasusBridge.jpg
Peopleware.jpg
PlotAgainstAmerica.jpg
Retribution.jpg
WarAndPeace.jpg

Renaming complete.

□ Specify the destination folder (If unchecked, images will be stored in the origin folder)

Destination direc... C:\Users\João Cardoso

D:\NetBeansWorkspace\Carica\testInput\books\test\test_JPG\test_JPG_0

Procurar test_JPG_0

test_0001_t0.jpg
test_0002_t0.jpg
test_0003_t0.jpg
test_0004_t0.jpg
test_0005_t0.jpg
test_0006_t0.jpg
test_0007_t0.jpg
test_0008_t0.jpg
test_0009_t0.jpg
test_0010_t0.jpg

10 itens

(a) Carica after renaming a batch of image files

(b) The batch of image files after ranaming

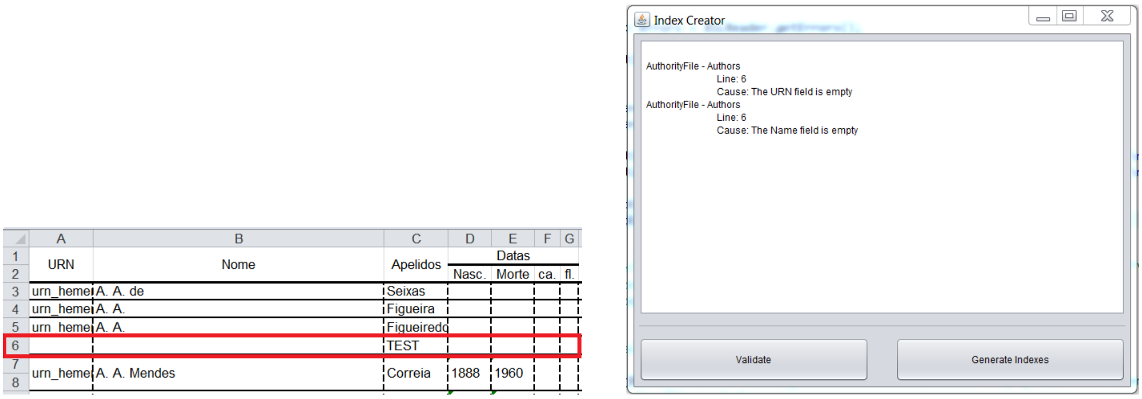
Figure 5.8: The results of renaming a batch of image files

5.2 Index Creator Application Validation

This section presents the tests that were used to validate each of the two use cases found in the index creator application. The two use cases are described in section 4.3.2.

5.2.1 Validate Index Creation

The first use case of the index creator application was the validation of index creation. Users should be able to perform a validation on the artefacts needed to create indexes (XLS authority file and XLS collaboration files), in order to check if there are no errors that would impede the indexes from being created. In order to validate this, the first step was to purposely corrupt the XLS authority file, in order to cause an error to appear during the index creator validation operation (see figure 5.9(a)). The next step was to run the index creation validation operation through the index creator application. The result of this operation was an error report, declaring both the errors that had been originally inserted into the XLS authority file (see figure 5.9(b)).



	A	B	C	D	E	F	G
1	URN	Nome	Apelidos	Datas			
2				Nasc.	Morte	ca.	fl.
3	urn_heme	A. A. de	Seixas				
4	urn_heme	A. A.	Figueira				
5	urn_heme	A. A.	Figueiredo				
6			TEST				
7	urn_heme	A. A. Mendes	Correia	1888	1960		
8							

```
AuthorityFile - Authors
          Line: 6
          Cause: The URN field is empty
AuthorityFile - Authors
          Line: 6
          Cause: The Name field is empty
```

(a) An entry with both the URN and Name fields empty

(b) The error report in the index creator application

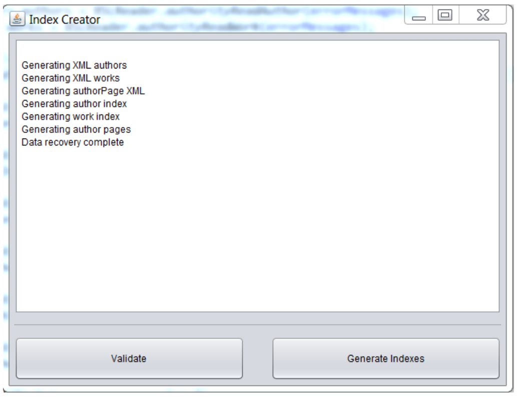
Figure 5.9: The error in the authority file and its report in the index creator application

5.2.2 Create Indexes

In regards to the index creator application, the second use case to be validated was the index creation. This use case stated that users should be able to create indexes, an operation which generates all the necessary HTML index files (author index, work index and author pages). In order to validate this use case, two steps were required.

First the indexes had to be generated through the index creator application. And second, the resulting HTML pages should present the same information as the existing indexes currently at use at the Hemeroteca.

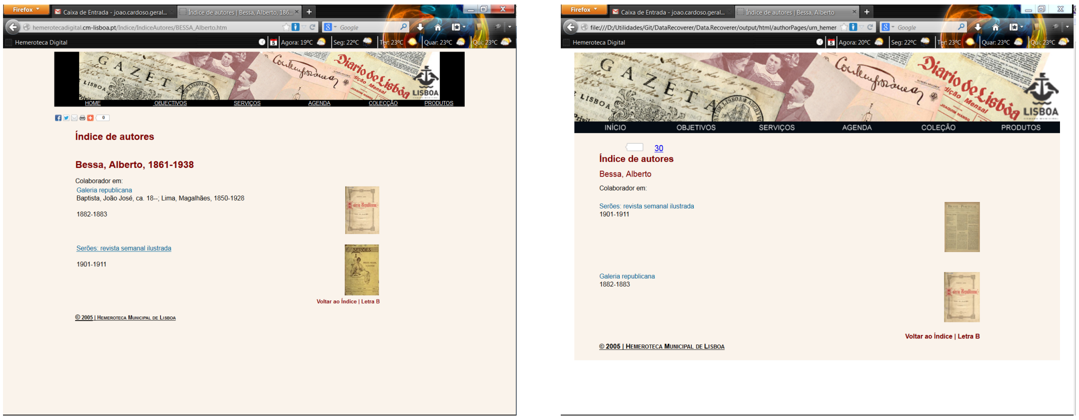
Therefore the first step was to generate the indexes through the use of the index creator application. The results of this operation, as they are displayed to the user, can be seen in figure 5.10. The second step was to verify that the resulting indexes



Generating XML authors
Generating XML works
Generating authorPage XML
Generating author index
Generating work index
Generating author pages
Data recovery complete

Figure 5.10: Index creation using the index creator application

completely replicated the original indexes. To this effect each of the three types of indexes that are generated by the index creator application was compared to their manually created counterpart. The results show that the new index creator application correctly replicates the existing indexes, as can be seen in figures 5.12, 5.13 and 5.11.


Firefox
Caixa de Entrada - joao.cardoso.geral...
Índice de autores | Bessa, Alberto, 186...
hemerotecadigital.cm-lisboa.pt/Indice/IndiceAutores/BESSA_Alberto.htm
Hemeroteca Digital
HOME
OBJECTIVOS
SERVIÇOS
AGENDA
COLECÇÃO
PRODUTOS
file:///D:/Utilidades/Git/DataRecoverer/Data.Recoverer/output/html/authorPages/urn_hemer
INÍCIO
OBJETIVOS
SERVIÇOS
AGENDA
COLEÇÃO
PRODUTOS
30

Índice de autores

Bessa, Alberto, 1861-1938

Colaborador em:

Galeria republicana
Baptista, João José, ca. 18--; Lima, Magalhães, 1850-1928

1882-1883

Serões: revista semanal ilustrada

1901-1911

Voltar ao Índice | Letra B

© 2005 | Hemeroteca Municipal de Lisboa

Índice de autores

Bessa, Alberto

Colaborador em:

Serões: revista semanal ilustrada
1901-1911

Galeria republicana
1882-1883

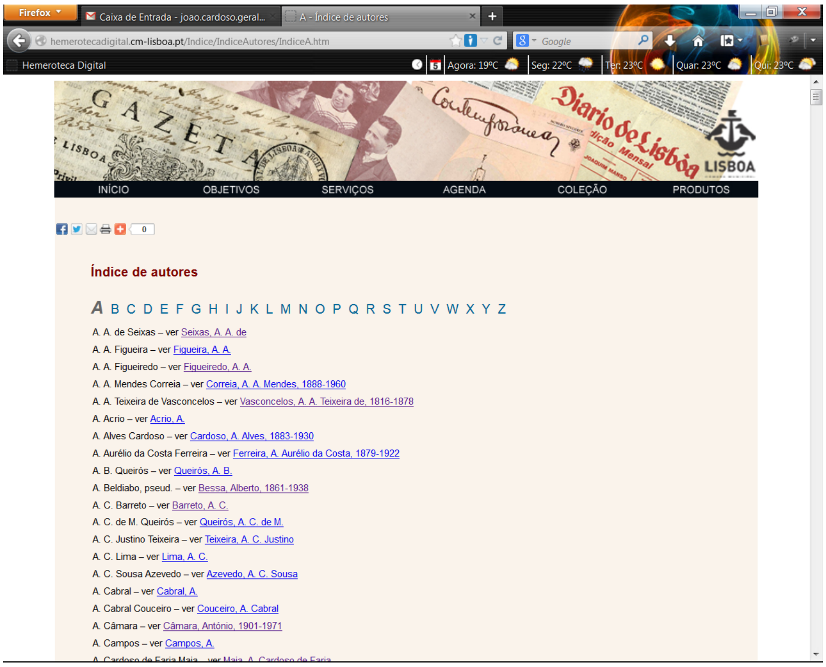
Voltar ao Índice | Letra B

© 2005 | Hemeroteca Municipal de Lisboa

(a) An example of a manually created author page

(b) An example of an automatically created author page

Figure 5.11: Comparison between author pages

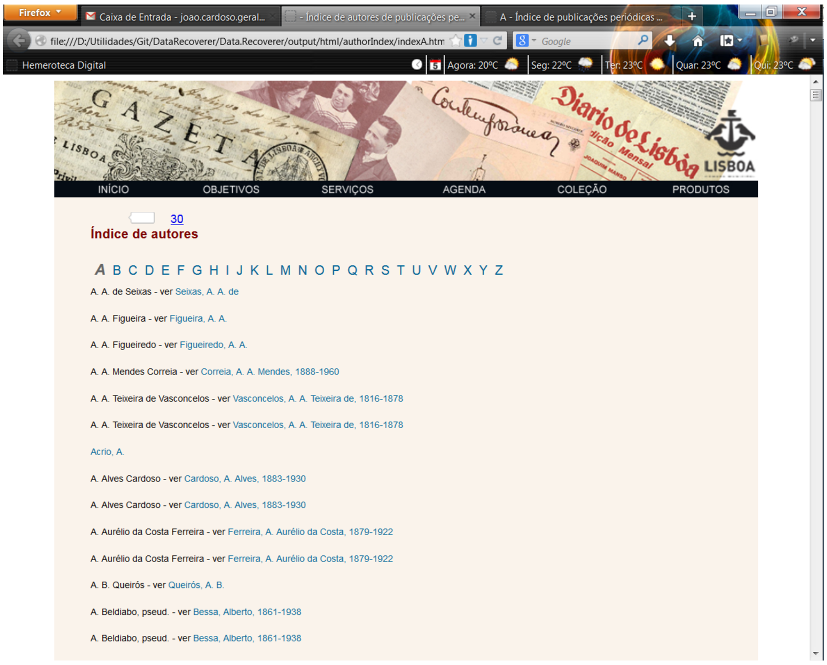


Índice de autores

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

A. A. de Seixas – ver Seixas, A. A. de
A. A. Figueira – ver Figueira, A. A.
A. A. Figueiredo – ver Figueiredo, A. A.
A. A. Mendes Correia – ver Correia, A. A. Mendes, 1888-1960
A. A. Teixeira de Vasconcelos – ver Vasconcelos, A. A. Teixeira de, 1816-1878
A. Acrio – ver Acrio, A.
A. Alves Cardoso – ver Cardoso, A. Alves, 1883-1930
A. Aurélio da Costa Ferreira – ver Ferreira, A. Aurélio da Costa, 1879-1922
A. B. Queirós – ver Queirós, A. B.
A. Beldiabo, pseud. – ver Bessa, Alberto, 1861-1938
A. C. Barreto – ver Barreto, A. C.
A. C. de M. Queirós – ver Queirós, A. C. de M.
A. C. Justino Teixeira – ver Teixeira, A. C. Justino
A. C. Lima – ver Lima, A. C.
A. C. Sousa Azevedo – ver Azevedo, A. C. Sousa
A. Cabral – ver Cabral, A.
A. Cabral Couceiro – ver Couceiro, A. Cabral
A. Câmara – ver Câmara, António, 1901-1971
A. Campos – ver Campos, A.
A. Cardoso de Faria Maia – ver Maia, A. Cardoso de Faria

(a) An example of a manually created author index


Firefox
Caixa de Entrada - joao.cardoso.geral...
- Índice de autores de publicações pe...
A - Índice de publicações periódicas ...
file:///D:/Utilidades/Git/DataRecoverer/Data.Recoverer/output/html/authorIndex/indexA.htm
Google
Hemeroteca Digital
Agora: 20ºC
Seg: 22ºC
Ter: 23ºC
Quar: 23ºC
Qui: 23ºC
GAZETA
LISBOA
INÍCIO
OBJETIVOS
SERVIÇOS
AGENDA
COLEÇÃO
PRODUTOS
30

Índice de autores

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

A. A. de Seixas - ver Seixas, A. A. de

A. A. Figueira - ver Figueira, A. A.

A. A. Figueiredo - ver Figueiredo, A. A.

A. A. Mendes Correia - ver Correia, A. A. Mendes, 1888-1960

A. A. Teixeira de Vasconcelos - ver Vasconcelos, A. A. Teixeira de, 1816-1878

A. A. Teixeira de Vasconcelos - ver Vasconcelos, A. A. Teixeira de, 1816-1878

Acrio, A.

A. Alves Cardoso - ver Cardoso, A. Alves, 1883-1930

A. Alves Cardoso - ver Cardoso, A. Alves, 1883-1930

A. Aurélio da Costa Ferreira - ver Ferreira, A. Aurélio da Costa, 1879-1922

A. Aurélio da Costa Ferreira - ver Ferreira, A. Aurélio da Costa, 1879-1922

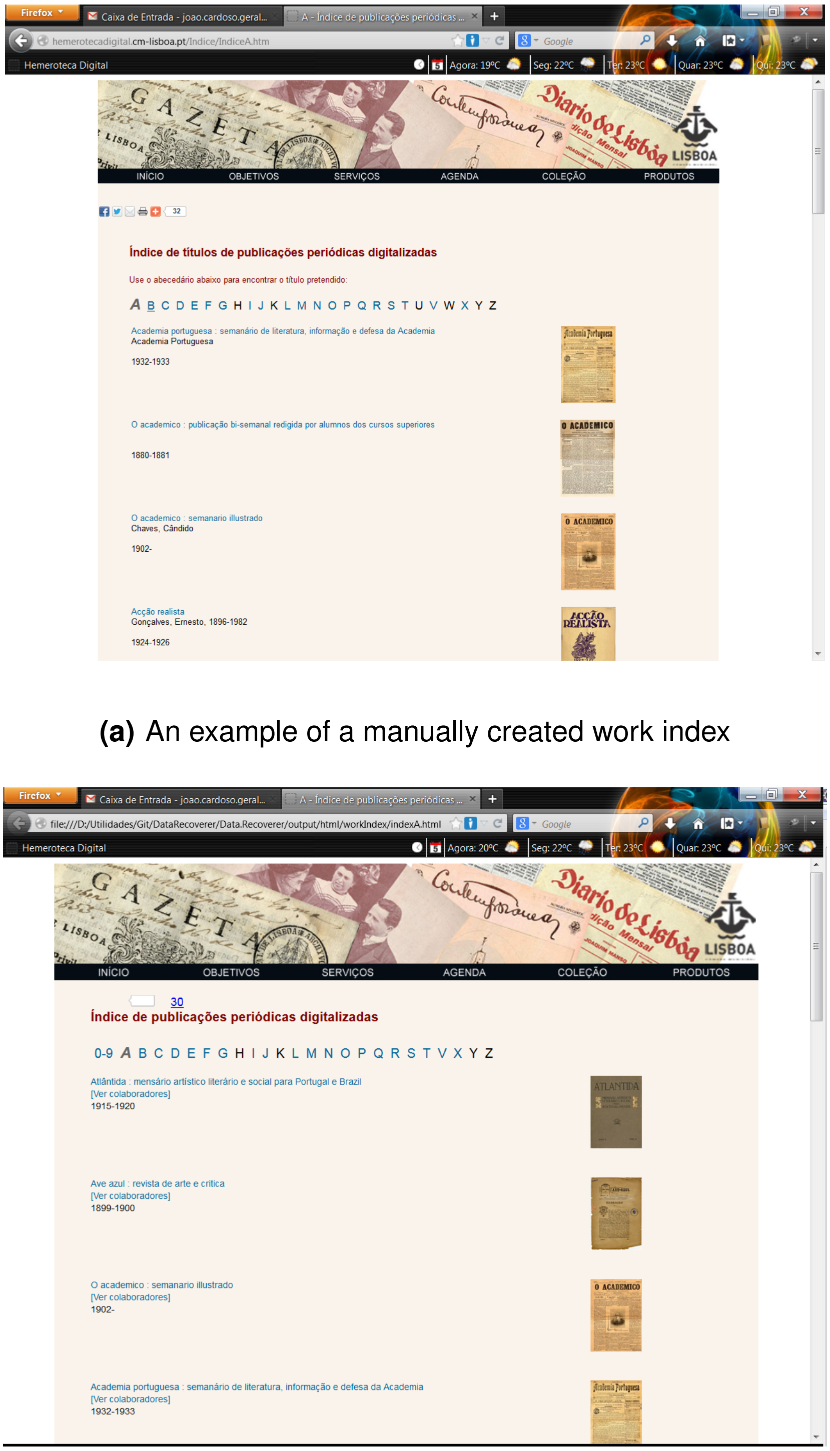
A. B. Queirós - ver Queirós, A. B.

A. Beldiabo, pseud. - ver Bessa, Alberto, 1861-1938

A. Beldiabo, pseud. - ver Bessa, Alberto, 1861-1938

(b) An example of an automatically created author index

Figure 5.12: Comparison between work indexes



Índice de títulos de publicações periódicas digitalizadas

Use o abecedário abaixo para encontrar o título pretendido:

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Academia portuguesa : semanário de literatura, informação e defesa da Academia
Academia Portuguesa

1932-1933

O academico : publicação bi-semanal redigida por alumnos dos cursos superiores

1880-1881

O academico : semanario illustrado
Chaves, Cândido

1902-

Acção realista
Gonçalves, Ernesto, 1896-1982

1924-1926

(a) An example of a manually created work index

Firefox
Caixa de Entrada - joao.cardoso.geral...
A - Índice de publicações periódicas ...
file:///D:/Utilidades/Git/DataRecoverer/Data.Recoverer/output/html/workIndex/indexA.html
Google
Hemeroteca Digital
Agora: 20ºC
Seg: 22ºC
Ter: 23ºC
Quar: 23ºC
Qui: 23ºC
LISBOA
INÍCIO
OBJETIVOS
SERVIÇOS
AGENDA
COLEÇÃO
PRODUTOS
30

Índice de publicações periódicas digitalizadas

0-9 A B C D E F G H I J K L M N O P Q R S T V X Y Z

Atlântida : mensário artístico literário e social para Portugal e Brazil
[Ver colaboradores]
1915-1920

Ave azul : revista de arte e critica
[Ver colaboradores]
1899-1900

O academico : semanario illustrado
[Ver colaboradores]
1902-

Academia portuguesa : semanário de literatura, informação e defesa da Academia
[Ver colaboradores]
1932-1933

(b) An example of an automatically created work index

Figure 5.13: Comparison between author indexes

6

Conclusions and Future Work

This chapter serves two distinct purposes. On one hand it provides closure to this report, by drawing conclusions on the outcome of this project. Whilst on the other it keeps the subject open by stating what could be done to improve that which has been implemented as a result of the work detailed in this report.

This chapter is divided into two distinct sections, section 6.1 presents the conclusions while section 6.2 goes through the future work that could be developed.

6.1 Conclusions

It is always tough to draw conclusions on a project whose execution extended for close to nine months. One could say that the overall objective of the project was achieved, for solutions were found and implemented that resulted in optimizations to a digital library's business processes and consequently to its workflows.

In what refers to the particular case of Hemeroteca's digitizing and image service business process, the issues pointed out in section 3.3 were all met throughout the execution of the project. Their solutions were sometimes not ideal but the deviations from the ideal solution were always brought about by constraints from having to solve a real problem. Nonetheless the staff members of the digitizing and image service can now go about their work on more effective and efficient manner.

The optimization of the P1.2 occurred without any significant problems. This was a bit unexpected, because although the overall objective of the Carica software application was quite simple, the technical challenges to achieve it were rarely so. Obviously this process while mostly uneventful was far from smooth. There were delays during its execution, most of which were down to the technical inexperience of the student to whom this master thesis refers.

Another point worth mentioning was the optimization of the P1.4. The implementation of the solution designed for this sub-process was a lengthy affair. As mentioned in section 4.3.2 the solution implied implementing two software applications. One to recover the existing data, and a second one which would be integrated into the P1.4.

Unfortunately data recovery soon turned into a quagmire, for the lack of consistency between HTML files slowed the process to a crawl. This is why a task which was supposedly simple, ended up extending itself for close to three months. This delay on the conclusion of the execution phase of the project meant that the testing phase of the index creator application could not be as extensive as would have been ideal. Which means that support will have to be given to the staff of the digitizing and image service long before the term of this project has ended.

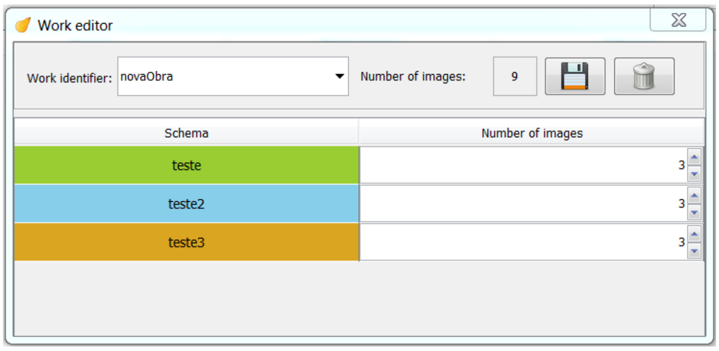
On balance one can say that the quality of the work developed within the realm of this thesis project was satisfactory. However given the circumstances, and considering that all objectives were achieved, one can not in good faith criticise the commitment to the tasks at hand. This is not to say that given more time both better solutions and an overall greater quality of work couldn't have been achieved, they obviously could as will be detailed in section 6.2.

6.2 Future Work

This section describes possible upgrades that could be made to the implemented solutions. These could be vary from adding functionalities, to the whole rethinking of the proposed solution.

6.2.1 Improvements to Carica

The Carica software application has a very simple objective, which is to create, edit and apply a set of renaming schemas to a batch of image files. One of the features that could be upgraded is the edition of works (see figure 6.1). This feature although implemented was never fully functional and often led to to the occurrence of errors during the work application, and during the work editing itself.



Work editor

Work identifier: novaObra

Number of images: 9

Schema	Number of images
teste	3
teste2	3
teste3	3

Figure 6.1: Carica work edit dialog

This is why an upgrade to Carica would always have to start by sorting out the issues with the work edit functionality. This would be mostly a technical task, and would not require much rethinking of the overall solution.

6.2.2 Improvements to the P1.4

The optimized P1.4 has already been extensively described both in chapters 4 and 5. However this section does not aim to further describe what has been accomplished, but what could be done to add value to the implemented solution.

One of the possibilities was to completely rethink the current solution for the creation of indexes. The existence of XLS files which serve as both the authority file and the collaboration files should be eliminated. The new solution would involve a three tier software application, as can be seen in figure 6.2.

The first tier would be the data access tier, and would implement methods that would link to a relational database which would implement the digitizing and image service authority file. The second tier would implement the application logic. All the application's functionalities would be implemented in this tier. The final tier would be a graphical user interface through which the staff members at the digitizing and image

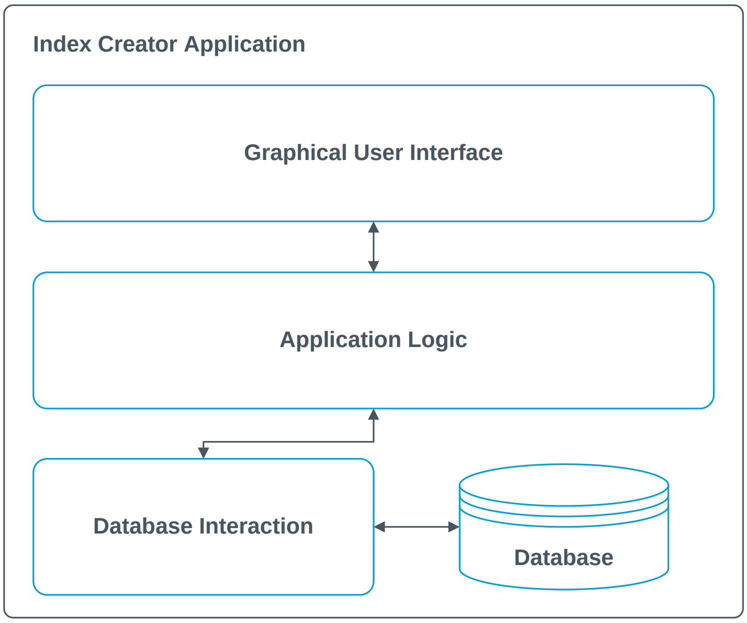


Figure 6.2: The architecture of a possible solution for the index creator application involving a relational database

service could edit the authority records, and create all the indexes that can be created based on the information present in the authority record.

This solution would completely eliminate all the concurrency issues that are present in the implemented solution.

A possible upgrade which wouldn't involve major changes to the current solution, was to promote changes to the fields within the authority file. By linking author pseudonyms with a certain work, work pages could be able to display the correct names of the authors which collaborated in them, something that currently is impossible to be done. Another change could be adding a field identifying locations mentioned in the work, so that a locations index could be created. Other types of indexes could be generated from data retrieved from the authority file.

Bibliography

[1] W. Y. Arms, Digital Libraries. Cambridge, Massachusetts: MIT Press, 2000.

[2] D. H. Clack, Authority control : principles, applications, and instructions. American Library Association Chicago, 1990.

[3] The Library of Congress, “TIFF, Revision 6.0,” http://www.digitalpreservation.gov/formats/fdd/fdd000022.shtml Retrieved 2012-12-05.

[4] ISO, “ISO 32000-1:2008 Document management – Portable document format – Part 1: PDF 1.7,” http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=51502, 2008.

[5] Adobe Systems, “Adobe PDF 101 - Quick overview of PDF file format,” http://partners.adobe.com/public/developer/tips/topic_tip31.html Retrieved 2012-12-06.

[6] Wallace, Gregory K., “The JPEG still picture compression standard,” Commun. ACM, vol. 34, no. 4, pp. 30–44, apr 1991. [Online]. Available: http://doi.acm.org/10.1145/103085.103089

[7] Ecma International, “Technical Report TR/98: JPEG File Interchange Format (JFIF) ,” http://www.ecma-international.org/publications/techreports/E-TR-098.htm, 2009.

[8] Camera & Imaging Products Association, “Exchangeable image file format for digital still cameras: Exif Version 2.3,” http://www.cipa.jp/english/hyoujunka/kikaku/pdf/DC-008-2010_E.pdf, 2010.

[9] Joint Photographic Experts Group, “JPEG 2000 official page,” http://www.jpeg.org/jpeg2000/index.html Retrieved 2012-12-11.

[10] The Free On-line Dictionary of Computing, “metadata,” http://dictionary.reference.com/browse/metadata, October 2012.

[11] NISO, “Understanding metadata,” 4733 Bethesda Avenue, Suite 300, Bethesda, MD 20814 USA, 2004.

[12] W3C, “Extensible markup language (XML) 1.0 (fifth edition),” W3C Recommendation 26 November 2008, November 2008.

[13] S. Weibel, J. Godby, E. Miller, and R. Daniel, “DC1: OCLC/NCSA metadata workshop: The essential elements of network object description,” in OCLC/NCSA Metadata Workshop Report, March 1995.

[14] S. Weibel, “The dublin core: A simple content description format for electronic resources,” Bul. Am. Soc. Inf. Tech., vol. 24, no. 1, pp. 9–11, October/November 1997.

[15] ISO, “ISO 2709:2008 information and documentation – format for information exchange,” September 2011.

[16] IFLA, “UNIMARC - consise bibliographic format,” http://archive.ifla.org/VI/8/unimarc-concise-bibliographic-format-2008.pdf, 2008.

[17] Library of Congress, “MARC 21 format for bibliographical data,” http://www.loc.gov/marc/bibliographic/ecbdhome.html, 1999.

[18] ——, “MARCXML MARC 21 XML Schema,” http://www.loc.gov/standards/marcxml/ Retrieved 2013-01-04.

[19] ISO, “ISO/DIS 25577 - information and documentation - MarcXchange,” http://www.loc.gov/standards/iso25577/, 2006.

[20] Library of Congress, “MODS Metadata Object Description Schema,” http://www.loc.gov/standards/mods/ Retrieved 2013-01-04.

[21] ——, “METS: Introdução & Tutorial,” http://www.loc.gov/standards/mets/METSOverview.V2_port.html Retrieved 2013-01-04.

[22] ISO/IEC JTC1/SC29/WG11/N5231, “MPEG-21 Overview v.5,” http://mpeg.chiariglione.org/standards/mpeg-21/mpeg-21.htm, October 2002.

[23] PREMIS Editorial Committee, “Data dictionary for preservation metadata: PREMIS version 2.0,” http://www.loc.gov/standards/premis, 2008.

[24] H. Ableson, B. Adida, M. Linksvayer, and N. Yergler, “ccREL: The Creative Commons Rights Expression Language,” 2008.

[25] IETF, “Uniform resource identifier (uri): Generic syntax,” http://tools.ietf.org/html/rfc3986, January 2005.

[26] ——, “Urn syntax,” http://tools.ietf.org/html/rfc2141, May 1997.

[27] ——, “Uniform resource locators (url),” http://tools.ietf.org/html/rfc1738, December 1994.

[28] B. Rosenblatt, “The digital object identifier: Solving the dilemma of copyright protection online,” The Journal of Electronic Publishing, vol. 3, no. 2, 1997.

[29] N. Paskin, “The digital object identifier system: digital technology meets content management,” Interlending & Document Supply, vol. 27, no. 1, pp. 13–16, 1999.

[30] Tim Berners-Lee, “Design Issues: Linked Data,” http://www.w3.org/DesignIssues/LinkedData.html Retrieved 2013-01-04.

[31] ANSI/NISO, “Information retrieval (Z39.50): Application service definition and protocol specification,” http://www.loc.gov/z3950/agency/Z39-50-2003.pdf, 2003.

[32] Library of Congress, “SRU version 1.1 archive,” http://www.loc.gov/standards/sru/sru1-1archive/index.html, July 2007.

[33] ——, “CQL: Contextual query language (SRU version 1.2 specifications),” http://www.loc.gov/standards/sru/specs/cql.html Retrieved 2013-01-04.

[34] ——, “Search/Retrieve Response Type,” http://www.loc.gov/standards/sru/sru1-1archive/xml-files/srw-types.xsd Retrieved 2012-11-20.

[35] ——, “SRW: Search/Retrieve Web Service,” http://www.loc.gov/standards/sru/sru1-1archive/srw.html, July 2007.

[36] W3C, “SOAP recomendation,” http://www.w3.org/TR/soap/ Retrieved 2013-01-04.

[37] C. Lagoze and H. V. de Sompel, “The open archives initiative: Building a low-barrier interoperability framework,” Proceedings of the 1st ACM/IEEE-CS Joint Conf. on Digital Libraries (JCDL’01), pp. 54–62, 2001.

[38] Borbinha, José and Gil, João and Pedrosa, Gilberto and Penas, João", “The Case of the Digitized Works at a National Digital Library,” in Proceedings of the 2nd Int. Conf. on Document Image Analysis for Libraries, ser. DIAL ’06. Washington, DC, USA: IEEE Computer Society, 2006, pp. 116–125. [Online]. Available: http://dx.doi.org/10.1109/DIAL.2006.42

[39] Borbinha, José and Pedrosa, Gilberto and Penas, João, “ContentE: flexible publication of digitised works with METS,” in Proceedings of the 9th European Conf. on Res. and Adv. Tech. for Digital Libraries, ser. ECDL’05. Berlin, Heidelberg: Springer-Verlag, 2005, pp. 537–538. [Online]. Available: http://dx.doi.org/10.1007/11551362_71

[40] Network Working Group, “RFC 1512: MIME (Multipurpose Internet Mail Extensions) Part One: Mechanisms for Specifying and Describing the Format of Internet Message Bodies,” http://tools.ietf.org/html/rfc1521, 1993.

[41] José Borbinha, João Gil, “Regras para Estruturas de Directórios e Nomes de Ficheiros de Imagens Digitalizadas.”

Appendix A - Carica Manual

A.1 Introdução ao Carica

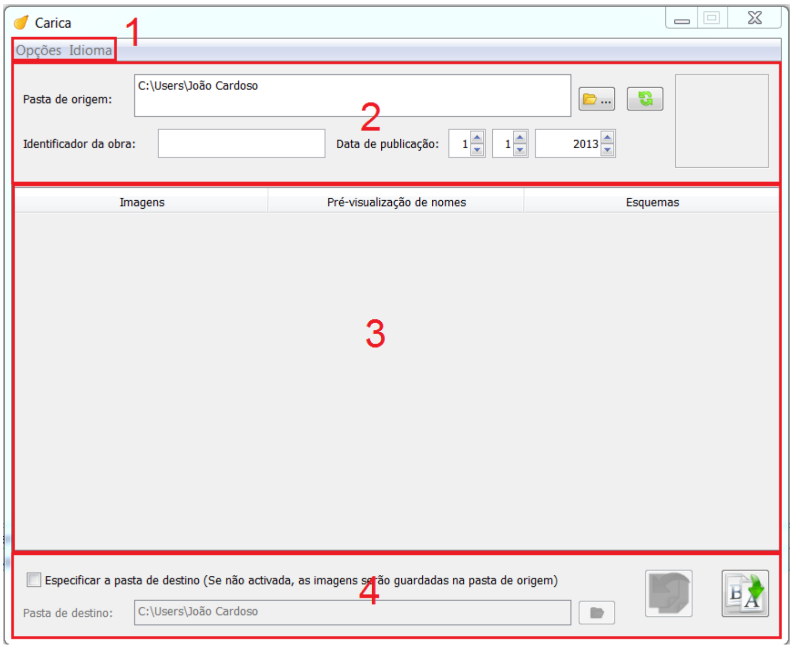
No processo de organização de uma obra digitalizada são inúmeras as tarefas para as quais é possível e vantajoso criar processos automáticos que facilitem a sua realização. A aplicação Carica foi portanto desenvolvida com o objectivo de automatizar o processo de normalização de nomes dos ficheiros de imagens de páginas digitalizadas segundo a sintaxe definida para a Biblioteca Nacional Digital.

A.1.1 Termos Importantes

Esquema Um esquema no contexto do Carica é um conjunto de opções, derivadas da sintaxe de normalização de nomes definida para a Biblioteca Nacional Digital, que serão aplicadas aos nomes dos ficheiros de imagens a ser normalizados.

Obra Uma obra no contexto do Carica é um conjunto de esquemas que são aplicados de forma agregada a um conjunto de nomes de ficheiros de imagem.

A.1.2 Ecrã Principal



Pasta de origem: C:\Users\João Cardoso

Identificador da obra:

Data de publicação: 1 1 2013

☐ Especificar a pasta de destino (Se não activada, as imagens serão guardadas na pasta de origem)

Pasta de destino: C:\Users\João Cardoso

Figure A.1: Ecrã principal do Carica

O ecrã principal do Carica pode ser dividido em 4 secções, tal como pode ser visto na figura A.1. As secções são, respectivamente:

1. Opções de edição de esquemas e de obras, e selecção de idioma;

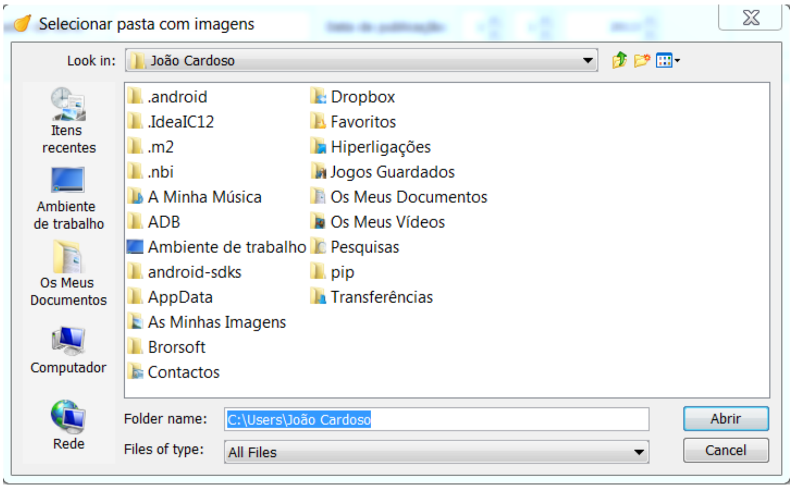
2. Campos de importação de ficheiros de imagem e preenchimento de dados para normalização dos seus nomes;

3. Visualizador de ficheiros de imagem que permite a aplicação de esquemas e obras;

4. Opções de renomeação de ficheiros de imagem.

A.2 Importar Imagens

O primeiro passo para a normalização de nomes de ficheiros de imagem é importar os ficheiros para o Carica. Para tal é necessário clicar no botão 📂... (procurar), que irá levar a que se abra um menu de selecção de ficheiros (ver figura A.2).



Look in: João Cardoso

Folder name: C:\Users\João Cardoso

Files of type: All Files

Figure A.2: Menu de selecção de ficheiros

Uma vez nesse menu basta seleccionar a pasta que contém os ficheiros e clicar no botão 'Abrir'. O Carica irá assumir o caminho para essa pasta como referência para futuras importações de ficheiros.

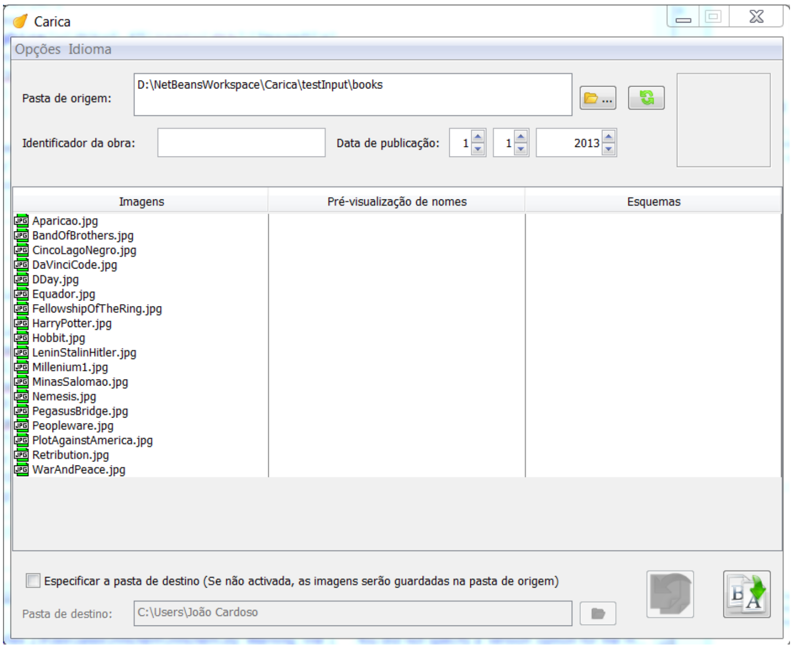
Uma vez importados os ficheiros irão ser listados na coluna de 'Imagens' do visualizador, tal como pode ser visto na figura A.3.

A.3 Esquemas

Um dos conceitos fundamentais do Carica é o esquema. Um esquema no contexto do Carica é um conjunto de opções, derivadas da sintaxe de normalização de nomes definida para a Biblioteca Nacional Digital, que serão aplicadas aos nomes dos ficheiros de imagens a ser normalizados.

O Carica possui um menu de criação e edição de esquemas. Este pode ser acedido através do menu de opções > editar esquemas (ver figura A.4).

O menu de criação e edição de esquemas permite ao utilizador escolher uma série de campos que são derivados da sintaxe de normalização de nomes em uso na Bib-



Pasta de origem: D:\NetBeansWorkspace\Carica\testInput\books

Identificador da obra:

Data de publicação: 1 1 2013

Imagens	Pré-visualização de nomes	Esquemas
Aparicao.jpg		
BandOfBrothers.jpg		
CincoLagoNegro.jpg		
DaVinciCode.jpg		
DDay.jpg		
Equador.jpg		
FellowshipOfTheRing.jpg		
HarryPotter.jpg		
Hobbit.jpg		
LeninStalinHitler.jpg		
Millenium1.jpg		
MinasSalomao.jpg		
Nemesis.jpg		
PegasusBridge.jpg		
Peopleware.jpg		
PlotAgainstAmerica.jpg		
Retribution.jpg		
WarAndPeace.jpg		

□ Especificar a pasta de destino (Se não activada, as imagens serão guardadas na pasta de origem)

Pasta de destino: C:\Users\João Cardoso

Figure A.3: Final do processo de importação de ficheiros de imagem

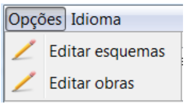


Figure A.4: Como aceder ao editor de esquemas

lioteca Nacional Digital. Para um melhor entendimento do significado de cada um campos, aconselha-se a leitura do documento "Regras para Estruturas de Directórios e Nomes de Ficheiros de Imagens Digitalizadas".

A figura A.5 representa o menu de criação e edição de esquemas:

1. Nome do esquema a ser criado, ou selecção do esquema para editar da listagem de esquemas existentes;
2. Botão de guardar esquemas;
3. Botão de remoção de esquemas;
4. Barra de pré-visualização do esquema, onde se podem ver os efeitos dos campos que forem seleccionados;
5. Campos de teor obrigatório;

6. Campos de teor opcional.

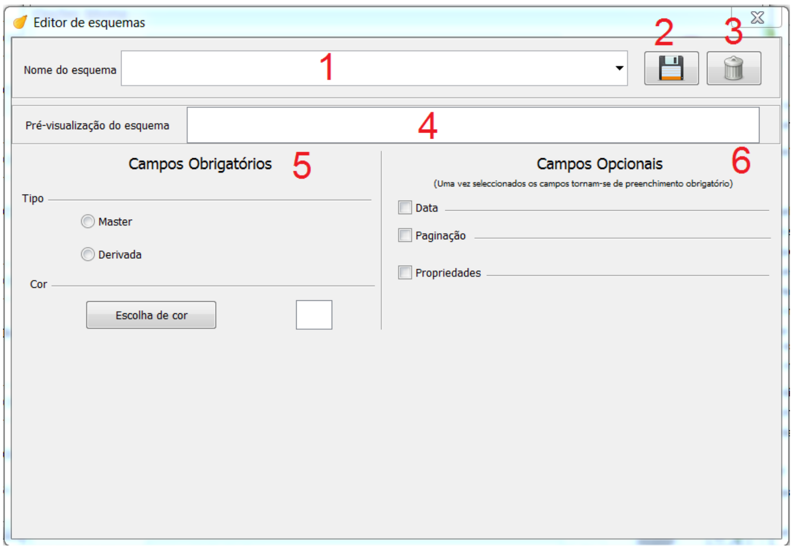


Figure A.5: Menu de edição e criação de esquemas

A.3.1 Criar Esquemas

De forma a criar um novo esquema o primeiro passo é definir um nome de esquema, sendo que em seguida o utilizador é forçado a preencher alguns campos que podem ser vistos na figura A.6.

Os campos obrigatórios são:

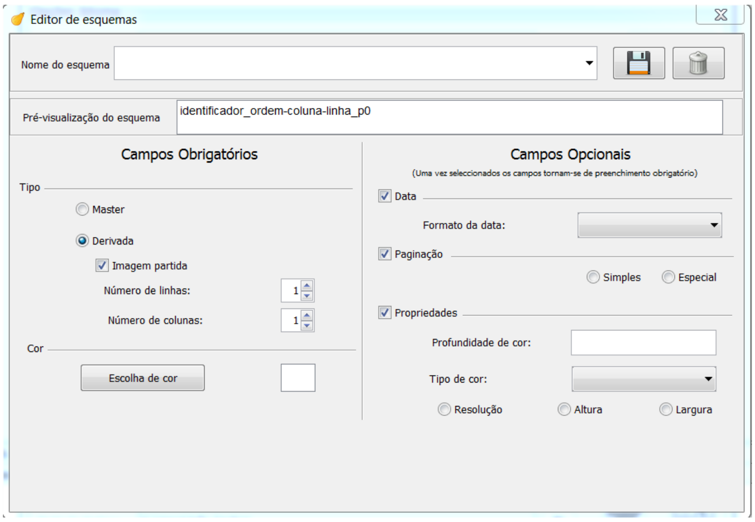
Tipo Identifica o tipo de codificação da imagem, com algumas das suas propriedades. Tais como a possibilidade de ser uma imagem que esteja partida em vários ficheiros;

Cor Utilizada para identificar o esquema no visualizador do ecrã principal do Carica.

O utilizador também poderá preencher campos opcionais, que são:

Data O formato com que a data deverá ser representada neste esquema;

Paginação Representa o número ou referência de série que se encontre ou se queira registar para cada imagem;



Editor de esquemas

Nome do esquema

Pré-visualização do esquema: identificador_ordem-coluna-linha_p0

Campos Obrigatórios

Tipo

○ Master

● Derivada

☑ Imagem partida

Número de linhas: 1

Número de colunas: 1

Cor

Escolha de cor

Campos Opcionais

(Uma vez seleccionados os campos tornam-se de preenchimento obrigatório)

☑ Data

Formato da data:

☑ Paginação

○ Simples ○ Especial

☑ Propriedades

Profundidade de cor:

Tipo de cor:

○ Resolução ○ Altura ○ Largura

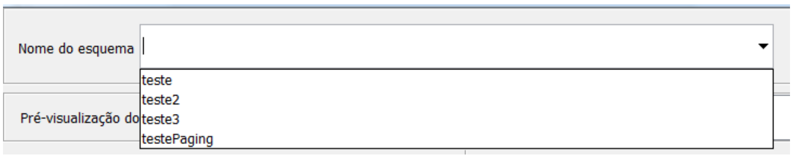
Figure A.6: Menu de edição e criação de esquemas com os campos visíveis

Propriedades Consiste numa sequência de campos registando características da imagem.

Uma vez definidos os campos que compõem o esquema, este é guardado clicando no botão 💾 (guardar).

A.3.2 Editar Esquemas

A edição de esquemas faz-se seleccionando um dos esquemas existentes e mostrados ao utilizador no campo do nome do esquema, tal como pode ser visto na figura A.7.



Nome do esquema

teste
teste2
teste3
testePaging

Pré-visualização do

Figure A.7: Escolha de um esquema para editar

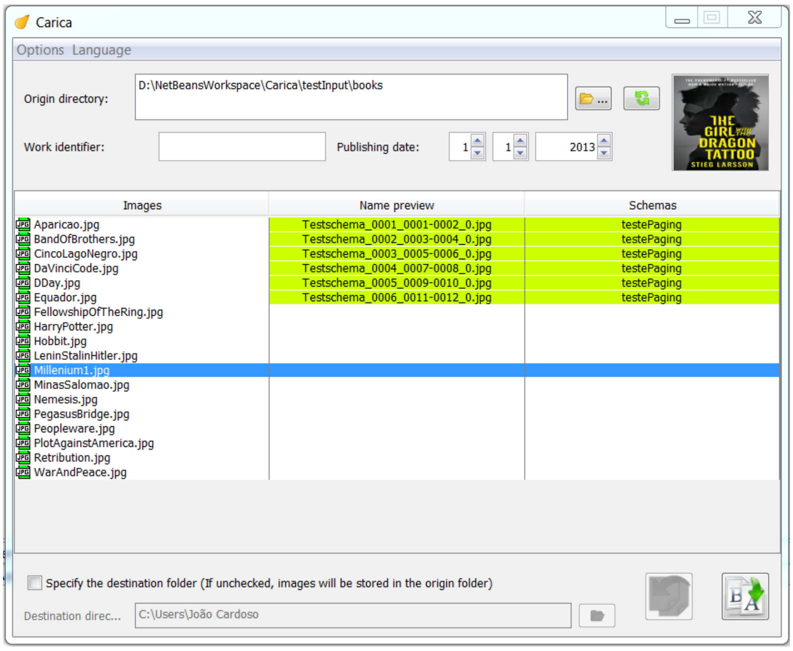
Uma vez escolhido o esquema é carregado no menu de criação e edição de esquemas, onde poderá ser editado sendo que qualquer alteração poderá levar à criação de um novo esquema (sendo para esse efeito necessária a atribuição de um novo nome), ou a substituição do esquema existente. Em ambos os casos qualquer alteração deverá ser guardada, clicando no botão 💾 (guardar).

Uma vez seleccionados e carregados no menu de criação e edição de esquemas, os esquemas também poderão ser completamente removidos. Sendo somente necessário clicar no botão 🗑 (remover).

A.3.3 Aplicar Esquemas

Uma vez criado um esquema, este passa a estar disponível para ser aplicado a ficheiros de imagem no visualizador do ecrã principal do Carica. Para tal deverão ser seleccionados os) ficheiros aos quais será aplicado o esquema e o utilizador deverá então abrir o menu de interacção com os ficheiros clicando com o botão direito do rato nos ficheiros seleccionados.

No menu de interacção deverá seleccionar a opção Esquemas >Aplicar Esquema sendo que depois seleccionará um dos esquemas disponíveis para aplicação, tal como pode ser visto na figura A.8.



Origin directory: D:\NetBeansWorkspace\Carica\testInput\books

Work identifier:

Publishing date: 1 1 2013

Images	Name preview	Schemas
Aparicao.jpg	Testschema_0001_0001-0002_0.jpg	testePaging
BandOfBrothers.jpg	Testschema_0002_0003-0004_0.jpg	testePaging
CincoLagoNegro.jpg	Testschema_0003_0005-0006_0.jpg	testePaging
DaVinciCode.jpg	Testschema_0004_0007-0008_0.jpg	testePaging
DDay.jpg	Testschema_0005_0009-0010_0.jpg	testePaging
Equador.jpg	Testschema_0006_0011-0012_0.jpg	testePaging
FellowshipOfTheRing.jpg		
HarryPotter.jpg		
Hobbit.jpg		
LeninStalinHitler.jpg		
Millenium1.jpg		
MinasSalomao.jpg		
Nemesis.jpg		
PegasusBridge.jpg		
Peopleware.jpg		
PlotAgainstAmerica.jpg		
Retribution.jpg		
WarAndPeace.jpg		

☐ Specify the destination folder (If unchecked, images will be stored in the origin folder)

Destination direc... C:\Users\João Cardoso

Figure A.8: Menu de interacção para aplicação de um esquema

Uma vez seleccionado o esquema um novo menu irá ser mostrado ao utilizador (ver figura A.9). No qual este terá de preencher alguns campos que variam de aplicação de esquema para aplicação de esquema. Os campos não são todos obrigatórios, e podem ser:

Identificador da Obra Um campo obrigatório, com um valor que deve ser único, extraído do exemplar digitalizado ou pela entidade detentora da obra, e como tal mantém-se constante para todas as imagens da mesma obra.

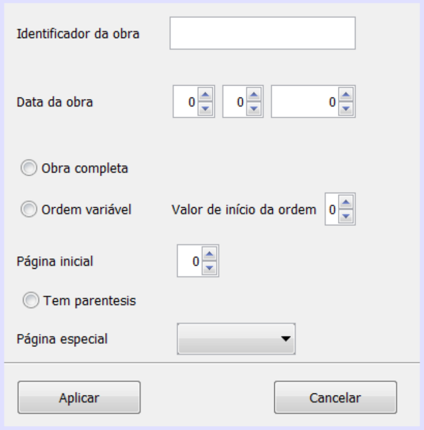
Data da obra Regista a data de publicação ou na sua falta qualquer outra data relevante para a obra e utilizada na sua descrição;

Ordem da obra Regista a ordem com que a imagem criada, devendo esta seguir a ordem do artefacto original. Este campo é obrigatório, sendo constituído por um número seguido opcionalmente por uma letra ou por um identificador de célula, no caso da imagem estar dividida em vários ficheiros. No caso da obra ser completa, o valor de ordem será 0000;

Página inicial O valor inicial da paginação dos ficheiros de imagem;

Parêtesis Indica se a paginação se encontra rodeada de parêntesis ou não;

Página especial Indica a página especial que se aplica neste esquema, caso essa opção esteja definida no esquema.



Identificador da obra

Data da obra 0 0 0

○ Obra completa

○ Ordem variável Valor de início da ordem 0

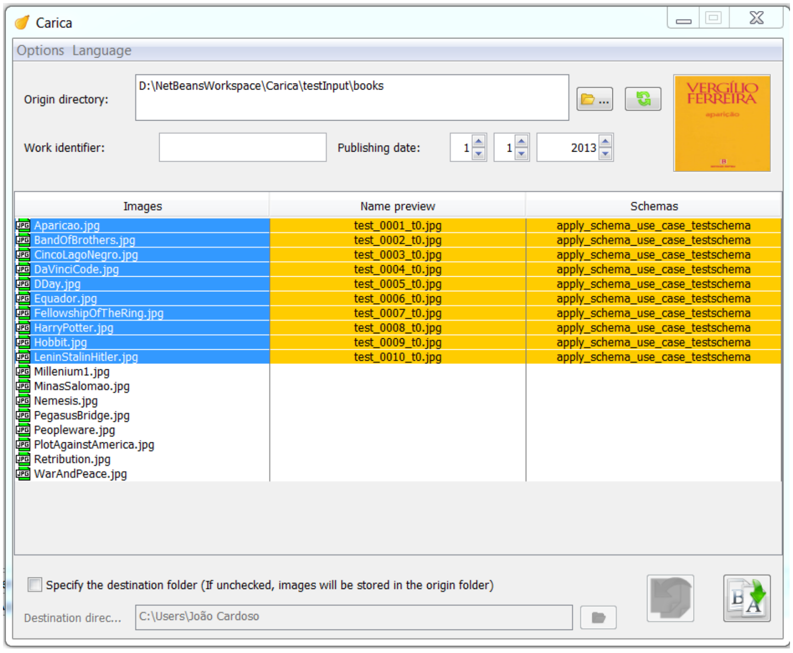
Página inicial 0

○ Tem parentesis

Página especial

Figure A.9: Menu de aplicação de esquemas

Uma vez terminado o preenchimento dos campos necessários, o utilizador deverá carregar no botão Aplicar (aplicar) o que levará a que o esquema seja aplicado aos ficheiros de imagem seleccionados, tal como pode ser visto na figura A.10.



Origin directory: D:\NetBeansWorkspace\Carica\testInput\books

Work identifier:

Publishing date: 1 1 2013

Images	Name preview	Schemas
Aparicao.jpg	test_0001_t0.jpg	apply_schema_use_case_testschema
BandOfBrothers.jpg	test_0002_t0.jpg	apply_schema_use_case_testschema
CincoLagoNegro.jpg	test_0003_t0.jpg	apply_schema_use_case_testschema
DaVinciCode.jpg	test_0004_t0.jpg	apply_schema_use_case_testschema
DDay.jpg	test_0005_t0.jpg	apply_schema_use_case_testschema
Equador.jpg	test_0006_t0.jpg	apply_schema_use_case_testschema
FellowshipOfTheRing.jpg	test_0007_t0.jpg	apply_schema_use_case_testschema
HarryPotter.jpg	test_0008_t0.jpg	apply_schema_use_case_testschema
Hobbit.jpg	test_0009_t0.jpg	apply_schema_use_case_testschema
LeninStalinHitler.jpg	test_0010_t0.jpg	apply_schema_use_case_testschema
Millenium1.jpg		
MinasSalomao.jpg		
Nemesis.jpg		
PegasusBridge.jpg		
Peopleware.jpg		
PlotAgainstAmerica.jpg		
Retribution.jpg		
WarAndPeace.jpg		

□ Specify the destination folder (If unchecked, images will be stored in the origin folder)

Destination direc... C:\Users\João Cardoso

Figure A.10: Resultado final da aplicação de um esquema

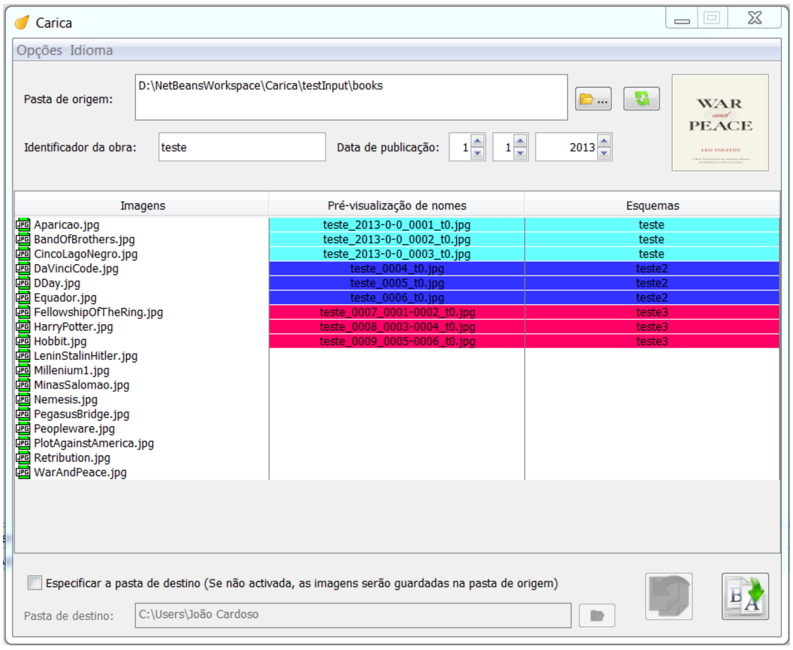
A.4 Obras

O conceito de obra no contexto do Carica é um conjunto de esquemas que são aplicados de forma agregada a um conjunto de nomes de ficheiros de imagem. Este conceito é extremamente útil quando se estão a processar ficheiro de imagem que representem obras com estruturas regulares. Dessa forma evita-se a aplicação repetitiva de esquemas, trocando-a por uma aplicação única de uma obra.

A.4.1 Criar Obras

De forma a criar uma obra é necessário primeiro definir os vários esquemas que a compõem, tal como foi descrito na secção A.3.1. É posteriormente feita uma aplicação dos vários esquemas aos ficheiros de imagem que irão servir de representação à obra, como foi descrito na secção A.3.3. O ponto de partida para a criação de uma obra deverá ser semelhante ao que pode ser visto na figura A.11.

Neste ponto, basta seleccionar a totalidade dos ficheiros que irão compor a obra e clicar com o botão do lado direito do rato no visualizador do ecrã principal do Carica, e escolhendo no menu de interacção a opção Obras >Criar obra (ver figura A.12).



Pasta de origem: D:\NetBeansWorkspace\Carica\testInput\books

Identificador da obra: teste

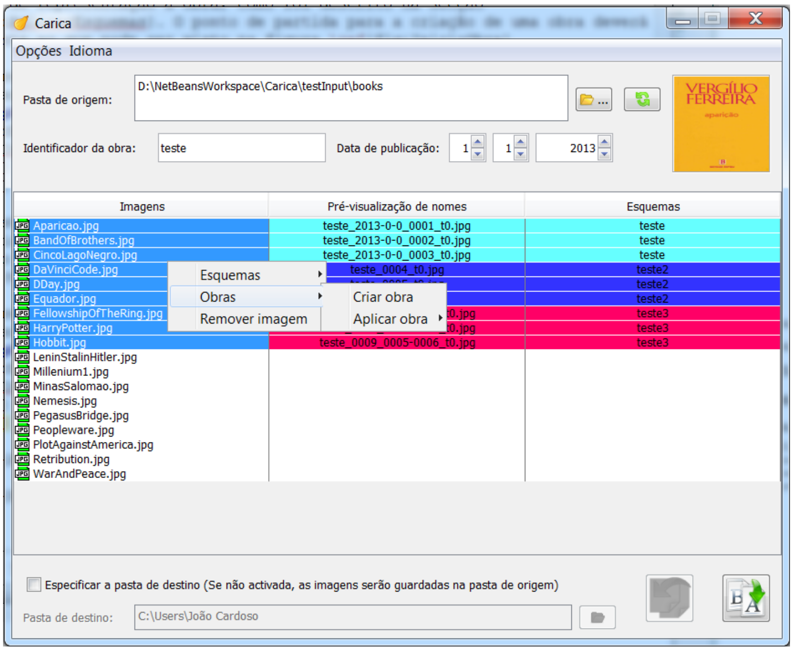
Data de publicação: 1 1 2013

Imagens	Pré-visualização de nomes	Esquemas
Aparicao.jpg	teste_2013-0-0_0001_t0.jpg	teste
BandOfBrothers.jpg	teste_2013-0-0_0002_t0.jpg	teste
CincoLagoNegro.jpg	teste_2013-0-0_0003_t0.jpg	teste
DaVinciCode.jpg	teste_0004_t0.jpg	teste2
DDay.jpg	teste_0005_t0.jpg	teste2
Equador.jpg	teste_0006_t0.jpg	teste2
FellowshipOfTheRing.jpg	teste_0007_0001-0002_t0.jpg	teste3
HarryPotter.jpg	teste_0008_0003-0004_t0.jpg	teste3
Hobbit.jpg	teste_0009_0005-0006_t0.jpg	teste3
LeninStalinHitler.jpg		
Millenium1.jpg		
MinasSalomao.jpg		
Nemesis.jpg		
PegasusBridge.jpg		
Peopleware.jpg		
PlotAgainstAmerica.jpg		
Retribution.jpg		
WarAndPeace.jpg		

□ Especificar a pasta de destino (Se não activada, as imagens serão guardadas na pasta de origem)

Pasta de destino: C:\Users\João Cardoso

Figure A.11: Ficheiros de imagem com vários esquemas aplicados



Pasta de origem: D:\NetBeansWorkspace\Carica\testInput\books

Identificador da obra: teste

Data de publicação: 1 1 2013

Imagens	Pré-visualização de nomes	Esquemas
Aparicao.jpg	teste_2013-0-0_0001_t0.jpg	teste
BandOfBrothers.jpg	teste_2013-0-0_0002_t0.jpg	teste
CincoLagoNegro.jpg	teste_2013-0-0_0003_t0.jpg	teste
DaVinciCode.jpg	teste_0004_t0.jpg	teste2
DDay.jpg		teste2
Equador.jpg		teste2
FellowshipOfTheRing.jpg	0.jpg	teste3
HarryPotter.jpg	0.jpg	teste3
Hobbit.jpg	teste_0009_0005-0006_t0.jpg	teste3
LeninStalinHitler.jpg		
Millenium1.jpg		
MinasSalomao.jpg		
Nemesis.jpg		
PegasusBridge.jpg		
Peopleware.jpg		
PlotAgainstAmerica.jpg		
Retribution.jpg		
WarAndPeace.jpg		

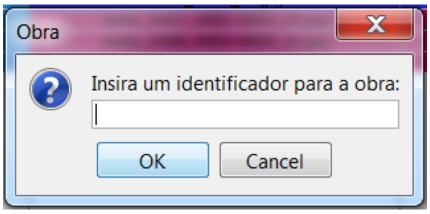
□ Especificar a pasta de destino (Se não activada, as imagens serão guardadas na pasta de origem)

Pasta de destino: C:\Users\João Cardoso

Figure A.12: Menu de interacção para criação de uma obra

Após seleccionar a opção "Criar obra" no menu de interacção sera pedido ao utilizador que defina um identificador para a obra, tal como pode ser visto na figura A.13.

A obra criada passará a estar disponível para aplicação.



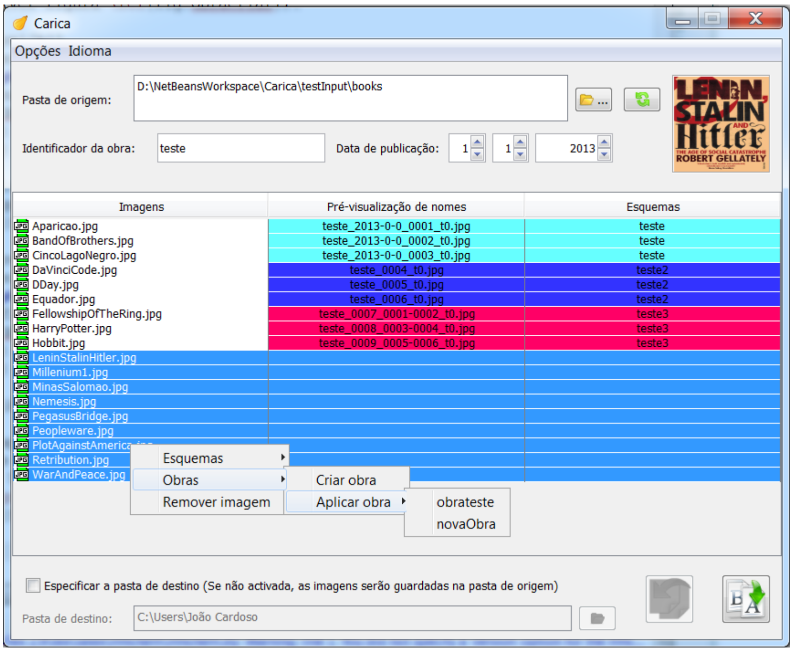
Obra

Insira um identificador para a obra:

Figure A.13: Definição do identificador da obra

A.4.2 Aplicar Obras

De forma a aplicar uma obra o utilizador deverá seleccionar o número necessário de ficheiros de imagem e clicar no botão direito do rato, invocando o menu de interacção. Uma vez no menu de interacção o utilizador deverá seleccionar a opção Obras >Aplicar obra, escolhendo a obra que pretende aplicar (ver figura A.14).



Carica

Pasta de origem: D:\NetBeansWorkspace\Carica\testInput\books

Identificador da obra: teste

Data de publicação: 1 1 2013

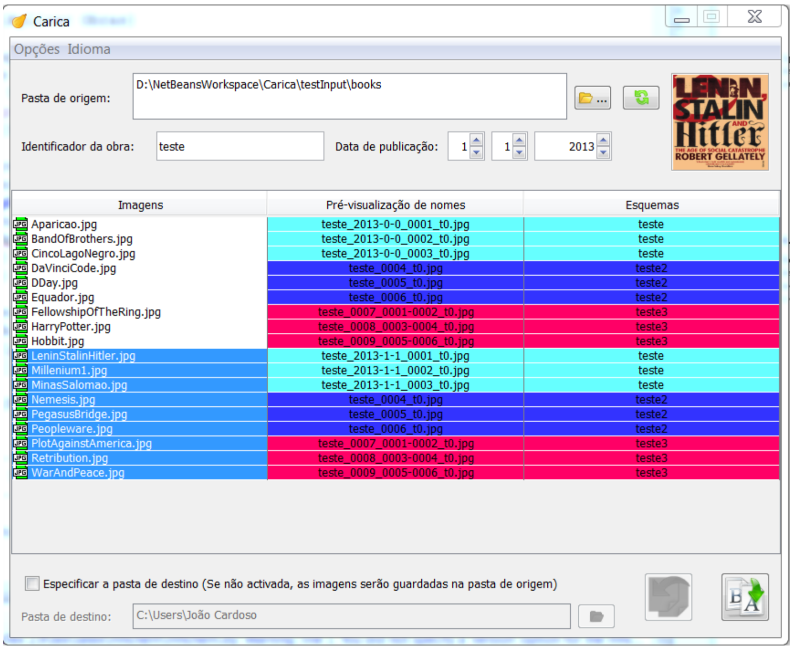
Imagens	Pré-visualização de nomes	Esquemas
Aparicao.jpg	teste_2013-0-0_0001_t0.jpg	teste
BandOfBrothers.jpg	teste_2013-0-0_0002_t0.jpg	teste
CincoLagoNegro.jpg	teste_2013-0-0_0003_t0.jpg	teste
DaVinciCode.jpg	teste_0004_t0.jpg	teste2
DDay.jpg	teste_0005_t0.jpg	teste2
Equador.jpg	teste_0006_t0.jpg	teste2
FellowshipOfTheRing.jpg	teste_0007_0001-0002_t0.jpg	teste3
HarryPotter.jpg	teste_0008_0003-0004_t0.jpg	teste3
Hobbit.jpg	teste_0009_0005-0006_t0.jpg	teste3
LeninStalinHitler.jpg		
Millenium1.jpg		
MinasSalomao.jpg		
Nemesis.jpg		
PegasusBridge.jpg		
Peopleware.jpg		
PlotAgainstAmerica.jpg		
Retribution.jpg		
WarAndPeace.jpg		

☐ Especificar a pasta de destino (Se não activada, as imagens serão guardadas na pasta de origem)

Pasta de destino: C:\Users\João Cardoso

Figure A.14: Menu de interacção para aplicação de uma obra

Uma vez seleccionada a obra assumirá os valores de "Identificador de obra" e "Data de publicação" inseridos nos campos respectivos do ecrã principal do Carica (ver secção A.6.1. O resultado final pode ser visível na figura A.15.



Pasta de origem: D:\NetBeansWorkspace\Carica\testInput\books

Identificador da obra: teste

Data de publicação: 1 1 2013

Imagens	Pré-visualização de nomes	Esquemas
Aparicao.jpg	teste_2013-0-0_0001_t0.jpg	teste
BandOfBrothers.jpg	teste_2013-0-0_0002_t0.jpg	teste
CincoLagoNegro.jpg	teste_2013-0-0_0003_t0.jpg	teste
DaVinciCode.jpg	teste_0004_t0.jpg	teste2
DDay.jpg	teste_0005_t0.jpg	teste2
Equador.jpg	teste_0006_t0.jpg	teste2
FellowshipOfTheRing.jpg	teste_0007_0001-0002_t0.jpg	teste3
HarryPotter.jpg	teste_0008_0003-0004_t0.jpg	teste3
Hobbit.jpg	teste_0009_0005-0006_t0.jpg	teste3
LeninStalinHitler.jpg	teste_2013-1-1_0001_t0.jpg	teste
Millenium1.jpg	teste_2013-1-1_0002_t0.jpg	teste
MinasSalomao.jpg	teste_2013-1-1_0003_t0.jpg	teste
Nemesis.jpg	teste_0004_t0.jpg	teste2
PegasusBridge.jpg	teste_0005_t0.jpg	teste2
Peopleware.jpg	teste_0006_t0.jpg	teste2
PlotAgainstAmerica.jpg	teste_0007_0001-0002_t0.jpg	teste3
Retribution.jpg	teste_0008_0003-0004_t0.jpg	teste3
WarAndPeace.jpg	teste_0009_0005-0006_t0.jpg	teste3

□ Especificar a pasta de destino (Se não activada, as imagens serão guardadas na pasta de origem)

Pasta de destino: C:\Users\João Cardoso

Figure A.15: Resultado da aplicação de uma obra

A.4.3 Editar Obras

O editor de obras do Carica ainda não se encontra totalmente desenvolvido na versão actual do Carica. No entanto é possível remover obras criadas, bastando para isso aceder ao editor de obras através do menu "Opções" no ecrã principal do Carica (ver figura A.16).

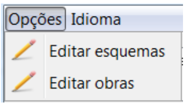
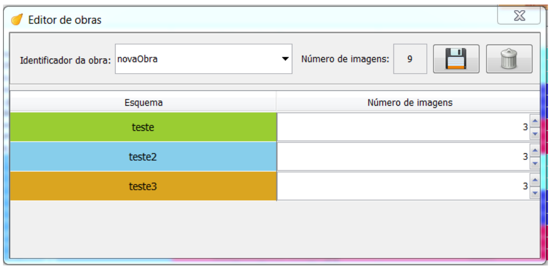


Figure A.16: Como aceder ao editor de obras

Uma vez no editor de obras basta ao utilizador seleccionar a obra que deseja remover, de forma análoga à selecção de esquemas no editor de esquemas (ver secção A.3.2), e clicar no botão 🗑 (remover) (ver figura A.17).



Editor de obras

Identificador da obra: novaObra

Número de imagens: 9

Esquema	Número de imagens
teste	3
teste2	3
teste3	3

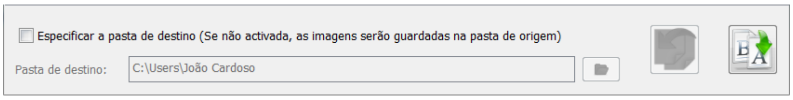
Figure A.17: Editor de obras

De momento a edição de obras a partir do editor é desaconselhada pois é uma funcionalidade que ainda está em desenvolvimento, podendo o seu uso levar a erros de sistema.

A.5 Normalização de Ficheiros de Imagem

Uma vez que a aplicação Carica foi portanto desenvolvida com o objectivo de automatizar o processo de normalização de nomes dos ficheiros de imagens de páginas digitalizadas, torna-se fundamental que haja uma operação de renomeação das imagens. Isto é, que a aplicação dos esquemas ou obras se traduza em alterações concretas aos nomes dos ficheiros de imagem.

Como tal, existe no ecrã principal do Carica uma secção que permite renomear ficheiros de imagem aos quais já tenha sido aplicado um esquema, oferecendo também a possibilidade ao utilizador de escolher a pasta de destino dos ficheiros de imagem renomeados (ver figura A.18).



□ Especificar a pasta de destino (Se não activada, as imagens serão guardadas na pasta de origem)

Pasta de destino: C:\Users\João Cardoso

Figure A.18: Secção de renomeação de ficheiros de imagem

A.5.1 Renomear Ficheiros de Imagem

Para renomear ficheiros de imagem, o utilizador tem primeiro de aplicar um esquema ou uma obra a uma selecção de ficheiros de imagem (ver secções A.3.3 e

A.4.2). Em seguida pode opcionalmente escolher a pasta de destino dos ficheiros de imagem renomeados. Sendo que para renomear os ficheiros de imagem basta ao utilizador clicar no botão BA (renomear).

O processo de renomeação irá, por definição, colocar os ficheiros de imagem renomeados dentro de uma estrutura de pastas na pasta de origem dos ficheiros de imagem. Essa estrutura de pastas reflecte o esquema que foi aplicado aos ficheiros de imagem.

A estrutura de pastas é a seguinte:

```
<identificador>
<identificador>_<formato>
<identificador>_<formato>_<propriedades>
```

Em que:

Identificador É o identificador da obra;

Formato É o formato do ficheiro de imagem (JPG, TIFF, PDF, etc.);

Propriedades São as propriedades do ficheiro de imagem, que correspondem ao campo opcional de propriedades do esquema que foi aplicado aos ficheiros de imagem.

A.5.2 Anular Renomeação

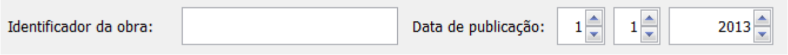
O Carica permite ao utilizador anular a última operação de renomeação de ficheiros de imagem. Para tal basta ao utilizador, após uma renomeação, clicar no botão ↶ (anular).

Ao anular a última operação de renomeação o Carica irá repor os ficheiros de imagem na pasta de origem, e eliminar a estrutura de pastas que tinha sido criada. Adicionalmente é permitido ao utilizador escolher se pretende recuperar quaisquer aplicações de esquema que tivessem sido feitas.

A.6 Outras Funcionalidades

A.6.1 Campos de aplicação geral

É possível ao utilizador especificar dois campos de forma geral, de modo a que não os tenha de introduzir sempre que aplicar um novo esquema ou obra. Para tal, basta ao utilizador simplesmente preencher os campos "Identificador da obra" e "Data de publicação" no ecrã principal do carica (ver figura A.19).



Identificador da obra:

Data de publicação: 1 1 2013

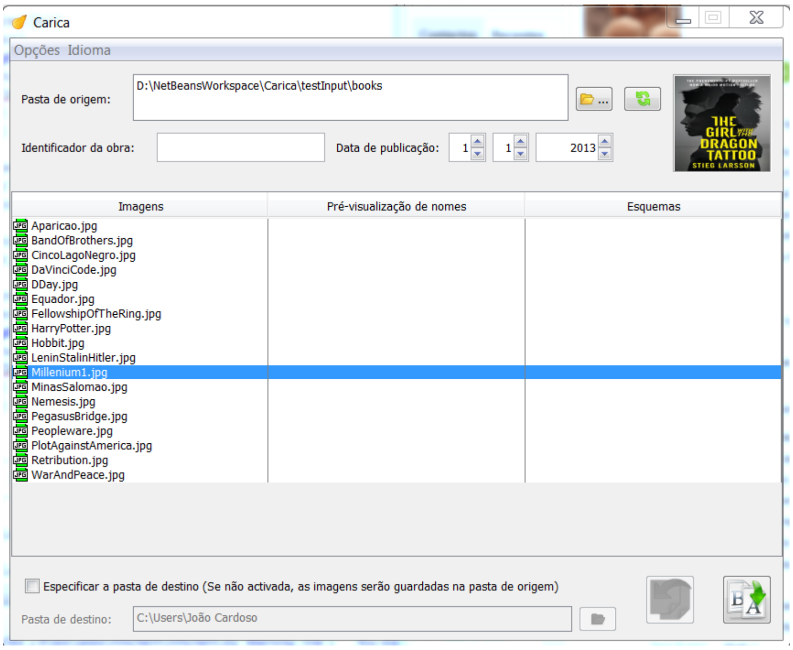
Figure A.19: Campos de aplicação geral

A.6.2 Pré-Visualização dos Ficheiros de Imagem

Uma das funcionalidades que estão implementadas no Carica, é a pré-visualização dos ficheiros de imagem. Para tal, basta seleccionar o ficheiro e uma miniatura do mesmo sera mostrada no local apropriado do ecrã principal do Carica, tal como pode ser visto na figura A.20. Adicionalmente essa miniatura pode ser expandida se for clicada duas vezes com o botão do lado esquerdo do rato.

A.6.3 Mudança de Idioma

Actualmente o Carica vem equipado com dois idiomas, Português e Inglês. Para fazer a mudança de idioma basta no ecrã principal do Carica seleccionar o menu idioma, que uma vez expandido permite a selecção do idioma Português ou Inglês (ver figura A.21).



Pasta de origem: D:\NetBeansWorkspace\Carica\testInput\books

Identificador da obra:

Data de publicação: 1 1 2013

Imagens	Pré-visualização de nomes	Esquemas
Aparicao.jpg		
BandOfBrothers.jpg		
CincoLagoNegro.jpg		
DaVinciCode.jpg		
DDay.jpg		
Equador.jpg		
FellowshipOfTheRing.jpg		
HarryPotter.jpg		
Hobbit.jpg		
LeninStalinHitler.jpg		
Millenium1.jpg		
MinasSalomao.jpg		
Nemesis.jpg		
PegasusBridge.jpg		
Peopleware.jpg		
PlotAgainstAmerica.jpg		
Retribution.jpg		
WarAndPeace.jpg		

□ Especificar a pasta de destino (Se não activada, as imagens serão guardadas na pasta de origem)

Pasta de destino: C:\Users\João Cardoso

Figure A.20: Exemplo de pré-visualização de ficheiro de imagem

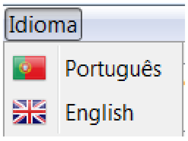


Figure A.21: Como seleccionar um outro idioma