Software Requirements Specification

1. Introduction

2. User Classes and Characteristics

The CSRepo tool will be used by a user profile, considered as a searcher, who wants to obtain metadata of scientific articles of computing. A second user profile, considered as a client system, can obtain the metadata of scientific articles of computing in an automated way. The last user profile is the administrator, which can change metadata such as see the notifications sent by other users too.

3. Concepts Definition

In this section are described the main concepts relevant to the system domain.
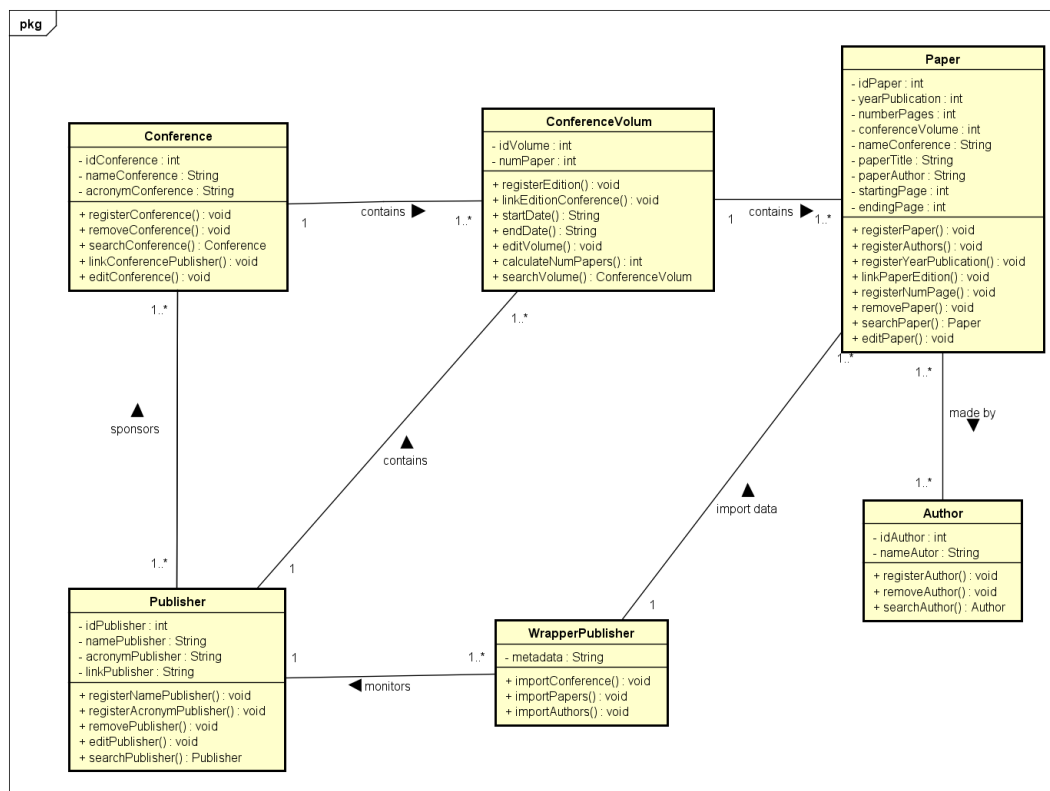
4. Software Requirements

This section describes the textual requirements of the product. Functional requirements are described in Section 4.1. Section 4.2 describes non-functional requirements.

4.1 Functional Requirements

[FR01] When the client system has access to the repository and requests the metadata exportation of an existing article in the repository, the system must ensure the exportation of the desired metadata in export patterns like as BibTex or XML.

[FR02] When the client system has access to the repository and requests the metadata exportation of an inexistent article in the repository, the system must return a message informing the absence of the metadata in the repository.

[FR03] When the client system has access to the repository and requests the metadata exportation of existing papers from a conference edition, the system must ensure the exportation of the desired metadata in only one archive all the metadata of this conference edition in export patterns like as BibTex or XML.

[FR04] When the client system has access to the repository and requests the metadata exportation of an inexistent conference edition, the system must return a message informing the absence of the edition metadata in the repository.

[FR05] When the researcher has access to the repository and requests the metadata exportation of papers from an inexistent conference edition, the system must return a message informing the absence of the conference edition in the repository.

[FR06] When the researcher has access to the repository and requests the metadata exportation of an existing article, the system must ensure the exportation of the desired metadata in an archive with export patterns like as BibTex or XML.

[FR07] When the researcher has access to the repository and request the metadata exportation of an inexistent paper in the repository, the system must return a message informing the absence of the paper in the repository.

[FR08] When the researcher has access to the repository and request a list of papers from an existent conference in the repository, the system must ensure the formation of a list of articles published at the desired conference, sorted by edition.

[FR09] When the researcher has access to the repository and request a list of papers from an inexistent conference in the repository, the system must return a message informing the absence of the conference in the repository.

[FR10] When the researcher has access to the repository and request a list by categories of conferences, the system must ensure the formation of a list of categories of conferences.

[FR11] When the researcher has access to the repository and report inconsistencies, the system must record the inconsistency, show a message that the inconsistency was registered and that will be forwarded for analysis.

[FR12] When the administrator has access to the repository and receive an inconsistency message from the repository, the system must ensure that all problems can be properly analyzed and resolved.

[FR13] When a report is received, the system must collect all the information about this report, so that the administrator can resolve the inconsistency.

[FR14] When the researcher has access to the repository and request the identification of all volumes added by the last update of the repository, the system must ensure the formation of a list with all volumes added by the last update and inform the date of update.

4.2 Non-Functional Requirements

4.2.1 Usability

[NFR01] The interface would be friendly, simple and intuitive. Error messages must be explained, in order to show how the user should act.

4.2.2 Reliability

[NFR02] If an error occurs in the processing of a transaction, the system must return to a previous consistent state, without compromising the consistency of the stored data.

4.2.3 Security

[NFR03] Only the administrator will have access to the system operations through a login provided by the system.

4.2.4 Correctness

[NFR04] The system should always return valid data to the user.

5. Requirements traceability

In this document, the traceability matrix of the System CSRepo is registered. Each tab in the worksheet represents bi-directional traceability between two different artifacts. The caption for understanding relationships is given below.

5.1 Traceability Matrix: User Story x Functional Requirement

US X	Functional Requirement													
	FR01	FR02	FR03	FR04	FR05	FR06	FR07	FR08	FR09	FR10	FR11	FR12	FR13	FR14
01	X	X	X	X										
02					X	X	X							
03						X	X							
04								X	X					
05										X				
06											X			
07												X	X	
08														X

US- User Story

FR- Functional Requirement