Beyond Borders

Software Requirements Specification

Version1

31st Jan 2014

Trupti Saundar
Vraj Shah
Abhinav Marda
Anmol Arora

Prepared for
Open Access Academic Journal Project
Instructor: Dr. Abhishek Srivastava
Client: Dr. Nirmala Menon
Spring2014.

Table of Contents

Introduction

Beyond Borders is an online journal management and publishing system to expand and improve the access of research. It is open source software made freely available to research scholars worldwide for the purpose of making open access publishing a viable option for more journals, as open access can increase a journal's readership as well as its contribution to the public good on a global scale. It will provide the facility of online submission of journals so that authors can get constructive feedback from expert reviewers and they could improve their research. It offers opportunity to readers to have an access to read journals of good quality on various topics of humanities.

1.1 Purpose

The purpose of this document is to present a detailed description of the Open Access Academic Journal System. It will explain the features of the system, the interfaces of the system, what the system will do and the constraints under which it must operate.
The intended audience for this SRS is client who wants the software to be built and developers.

1.2 Scope

The SRS is intended to produce a website to be called Beyond Borders. It will have many features that will help research scholars to help break down barriers that hinder communication and collaboration among researchers and students of Digital Humanities, around the world. Some of the intended features which this platform will provide are:

1. Online Submission by the author and constructive feedback
2. Preformatted reply forms will be used in every stage of articles' progress, the location of these forms will be configurable via applicant's maintenance options.
3. Communication between authors, reviewers and editor through email
4. Login and registering
5. Browse by different categories
6. Latest news
7. Relational database containing list of Authors, Reviewers and Articles and information associated with them
8. Facility to view latest and most read articles
9. Submission of comments by Reviewers to the Author
10. Readers can download the article under *Creative Commons License*

1.3 Definitions, Acronyms, and Abbreviations

Term	Definition
Author	*Person submitting an article to be reviewed. In case of multiple authors, this term refers to the* principal author*, with whom all communication is made.*
Active Article	*The document that is tracked by the system; it is a narrative that is planned to be posted to the public website.*
Database	*Collection of all the information monitored by this system.*
Editor	*Person who receives articles, sends articles for review, and makes final judgments for publications.*
Humanities Society Database	*The existing membership database (also HS database).*
Member	*A member of the Humanities Society listed in the HS database.*
Reader	*Anyone visiting the site to read articles.*
Review	*A written recommendation about the appropriateness of an article for publication; may include suggestions for improvement.*
Reviewer	*A person that examines an article and has the ability to recommend approval of the article for publication or to request that changes be made in the article.*
Software Requirements Specification	*A document that completely describes all of the functions of a proposed system and the constraints under which it must operate. For example, this document.*
User	*Reviewer or Author.*

1.4 References

IEEE. *IEEE STD 830-1998 IEEE Recommended Practice for Software Requirements Specifications*. IEEE Computer Society, 1998

1.5. Overview of Document

The next chapter, the General Description Section, of this document gives an overview of functionality of the product. It describes the informal requirements and is used to establish a context for the technical requirements in the next chapter.

The third chapter, Specific Requirements section, of this document is written primarily for the developers and describes in technical terms the details of the functionality of the product.

Both sections of the document describe the same software product, but are intended for different audiences and thus use different language.

2. General Description

2.1 Product Perspective

This Product is entirely a new, self-contained product. The Open Access Academic Journal System has 4 main Actors (Author, Reviewer, Editor, Reader) and one cooperative system. The Author, Reader and reviewer access the Online Journal through Internet. Any Author or Reviewer communication with the system is through email. The Editor accesses the entire system directly. This it requires a web browser with an internet connection and server supporting PHP and a MySQL database.

2.2 Product Functions

This section outlines the use cases for each of the active readers separately. The reader, the author and the reviewer have only one use case apiece while the editor is main actor in this system

2.2.1 Reader Use Case

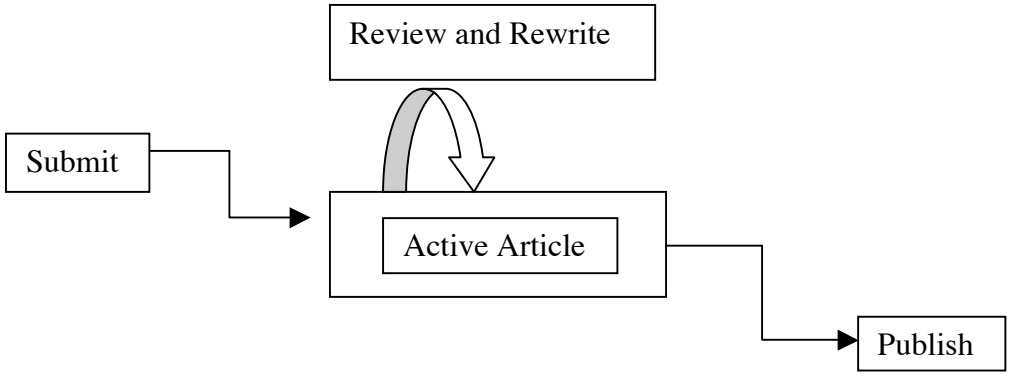
Use Case: Search Article

Brief Description

The Reader accesses the Online Journal Website, searches for an article and downloads it to his/her machine.

Initial Step-By-Step Description

1. The Reader chooses to search by author name, date of issue or title.
2. The system displays the choices to the Reader.
3. The Reader selects the article desired.
4. The system presents the abstract of the article to the reader.
5. The Reader chooses to download the article.
6. The system provides the requested article.



The *Article Submission Process* state-transition diagram summarizes the use cases listed below. An Author submits an article for consideration. The Editor enters it into the system and assigns it to and sends it to at least three reviewers. The Reviewers return their comments, which are used by the Editor to make a decision on the article. Either the article is accepted as written, declined, or the Author is asked to make some changes based on the reviews. If it is accepted, possibly after a revision, the Editor sends a copyright form to the Author. When that form is returned, the article is published to the Online Journal. Not shown in the above is the removal of a declined article from the system.

2.2.2 Author Use Case

Use Case: Submit Article

Brief Description The author either submits an original article or resubmits an edited article.

Initial Step-By-Step Description

1. The Author will sign up for an account and get registered
2. The Author chooses the *Email Editor* Button.
3. The System uses the *send to* HTML tag to bring up the user's email system.
4. The Author fills in the Subject line and attaches the files as directed and emails them.
5. The System generates and sends an email acknowledgement.

2.2.3 Reviewer Use Case

Use Case: Submit Review

Brief Description
The reviewer submits a review of an article.

Initial Step-By-Step Description

1. The Reviewer chooses the *Email Editor* Button.
2. The System uses the *send to* HTML tag to bring up the user's email system.
3. The Reviewer fills in the Subject line and attaches the file as directed and emails it.
4. The System generates and sends an email acknowledgement.

2.2.4 Editor Use Case

Use Case1: Assign Reviewer

Brief Description
The Editor assigns one or more reviewers to an article.

Initial Step-By-Step Description

1. The Editor selects to *Assign Reviewer button*.
2. The system presents a list of Reviewers .The Editor selects a Reviewer.
3. The Editor repeats step 2 until sufficient reviewers are assigned.
4. The system emails the Reviewers, attaching the article and requesting that they do the review.

Use Case2: Send Response

Brief Description
The Editor sends a response to an Author.

Initial Step-By-Step Description

1. The Editor selects to *Send Response button*.
2. The system calls the email system and puts the Author's email address in the Recipient line and the name of the article on the subject line.
3. The Editor fills out the email text and sends the message.

Use Case3: Send Copyright

Brief Description
The Editor sends a copyright form to an Author.

Initial Step-By-Step Description

1. The Editor selects to *Send Copyright button*.
2. The system calls the email system and puts the Author's email address in the Recipient line, the name of the article on the subject line, and attaches the copyright form.
3. The Editor fills out the email text and sends the message.

Use Case 4: Publish Article

Brief Description

The Editor transfers an accepted article to the Online Journal.

Initial Step-By-Step Description

1. The Editor selects to *Publish Article*.
2. The system transfers the article to the Online Journal and updates the search information there.

2.3User Characteristics

The Reader is expected to be Internet literate and be able to use search engine. The Author and Reviewer are expected to be Internet literate and to be able to use email with attachments. The Editor is expected to be able to use email with attachments and also several buttons.

2.4 General Constraints

Constraints for the website include:

1. Time(less than 2 minutes to load the website)
2. Limited availability of budget to buy a domain
3. Availability of browser with internet connection and server supporting PHP and MySQL Database to use this software

2.5 Assumptions and Dependencies

It is assumed that device (computer, mobile) is having a web browser with an internet connection and server supporting PHP and a MySQL database to use this software.

3. Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

The website should work and be tested against IE, Firefox, Google Chrome and Netscape.

Whenever user (reader, editor, reviewer) visits website he/she should be able to see main page. The home tab should show the main page layout as shown in fig 1(main page).

About tab should show the window shown in fig 2(About Page)

If user wants to register to become a member of humanities society he/she should fill details shown in Register Page (e.g. Username, password, full name, email id etc.)

For login into already created account user should click on login tab and should provide username and password. This can also by using user section on the right of main page.

To search an article user can use search tab. To browse articles by name, title, date user should select corresponding appropriate tab under browse by section on the right of main page. Tabs on left section of main page can be used to view archives, latest articles, most read article and current issues.
User can send an email to editor by using contact us tab.

Fig 1-Main Page

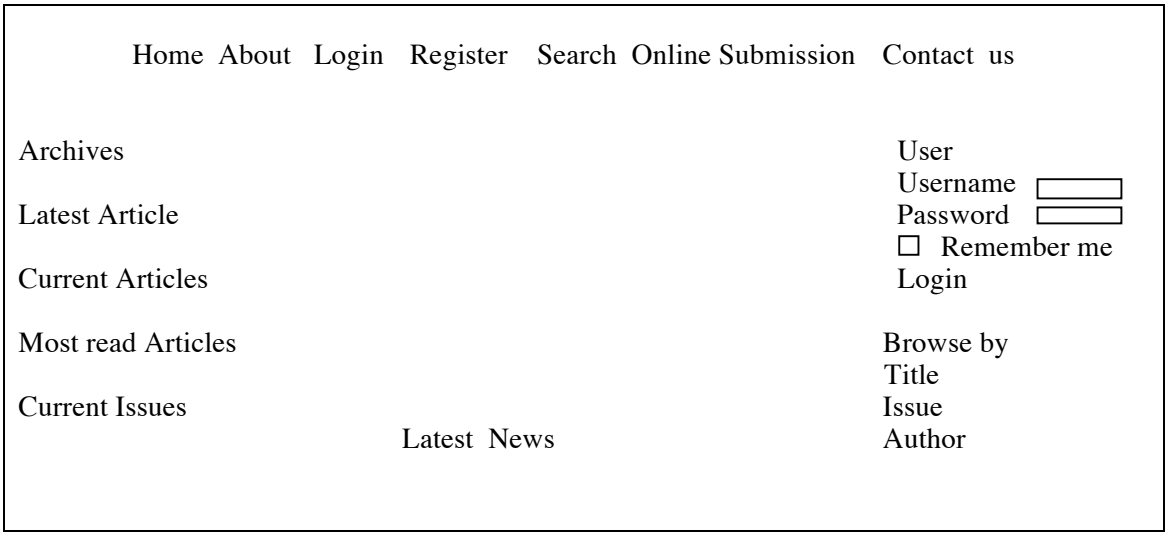
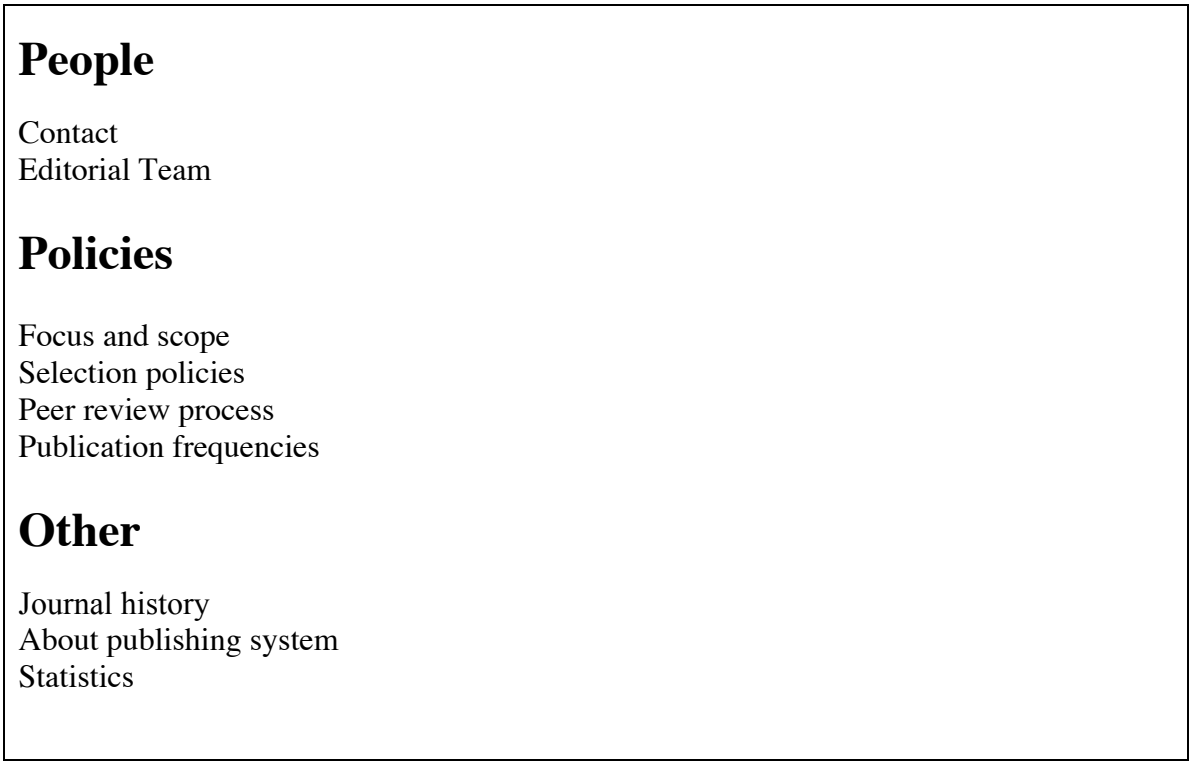


Fig 2-About page



The only link to an external system is the link to the Humanities Society (HS) Database to verify the membership of a Reviewer. The Editor believes that a society member is much more likely to be an effective reviewer and has imposed a membership requirement for a Reviewer. The HS Database fields of interest to the Web Publishing Systems are member's name, membership (ID) number, and email address (an optional field for the HS Database)

The *Assign Reviewer* use case sends the Reviewer ID to the HS Database and a Boolean is returned denoting membership status.

3.1.2 Hardware Interfaces

This software will be developed in PHP, MySQL and HTML5. These technologies require a minimum hardware configuration, but no other hardware requirements are needed. There is no explicit interaction with the client's hardware.

3.1.3 Software Interfaces

There is no special software interface requirement. This website only needs a browser with internet connection and server supporting PHP, MySQL.

3.1.4 Communications Interfaces

All communications should be done using email system. There is no special communication interfaces.

3.2 Functional Requirements

3.2.1 Search Article

Use Case Name	Search Article
Reference	Section 2.2.1, Search Article
Trigger	The Reader accesses the Online Journal Website
Precondition	The Web is displayed with grids for searching
Basic Path	1. The Reader chooses how to search the articles. The choices are by Author, by Title, and by Issue. 2. If the search is by Author, the system creates and presents an alphabetical list of all authors in the database. In the case of an article with multiple authors, each is contained in the list. 3. The Reader selects an author. 4. The system creates and presents a list of all articles by that author in the database. 5. The Reader selects an article. 6. The system displays the Abstract for the article. 7. The Reader selects to download the article or to return to the article list or to the previous list.
Alternative Paths	In step 2, if the Reader selects to search by issue, the system creates and presents a list of all issues of articles in the database. 3. The Reader selects an issue. 4. Return to step 6 5. In step 2, if the Reader selects to browse by title, the 6. The system presents a list of all such articles with title in the database. Return to step 5.
Post condition	The selected article is downloaded to the client machine.
Exception Paths	The Reader may abandon the search at any time.
Actor	Reader

3.2.2 Submit article

Use Case Name	Submit article
Reference	Section 2.2.2, Submit Article
Trigger	The user selects online submission tab
Precondition	The user is on the Online Journal Main Page.
Basic Path	1. The Author chooses the *Email Editor* button. 2. The System uses the *send to* HTML tag to bring up the user's email system. 3. The Author fills in the Subject line and attaches the files as directed and emails them.

Alternative Paths	None.
Post condition	The System generates and sends an email acknowledgement.
Exception Paths	The attempt may be abandoned at any time.
Actor	Author

3.2.3 Submit Review

Use Case Name	Submit review
Reference	Section 2.2.3 Submit Article
Trigger	The user selects Email Editor Button.
Precondition	The user selects the contact us tab on main page
Basic Path	1. The Reviewer chooses the *Email Editor* button. 2. The System uses the *send to* HTML tag to bring up the user's email system. 3. The Reviewer fills in the Subject line and attaches the file as directed and emails it.
Alternative Paths	None
Post condition	The message is sent.
Exception Paths	The attempt may be abandoned at any time.
Actor	reviewer

3.2.4 Assign Reviewer

Use Case Name	Assign Reviewer
Reference	Section 2.2.4, Assign Reviewer
Trigger	The Editor selects to assign a reviewer to an article.
Precondition	The Editor has accessed the Article Manager main screen and the article is already in the database. .
Basic Path	1. The Editor selects the article 2. The system presents an alphabetical list of reviewers with their information. 3. The Editor selects a reviewer for the article. 4. The system updates the article database entry and emails the reviewer with the standard message and attaches the text of the article without author information. 5. The Editor has the option of repeating this use case from step 2.
Alternative Paths	None.
Post condition	At least one reviewer has been added to the article information and the appropriate communication has been sent.
Exception Paths	The Editor may abandon the operation at any time.
Actor	editor

3.2.5 Send response or copyright

Use Case Name	Send response or copyright
Reference	Section 2.2.4, Send Response; Section 2.2.4, Send Copyright
Trigger	The editor selects to send a communication to an author.
Precondition	The Editor has accessed the Article Manager main screen.
Basic Path	1. The system presents an alphabetical list of authors. 2. The Editor selects an author. 3. The system invokes the Editor's email system entering the author's email address into the *To:* entry. 4. The Editor uses the email facility.
Alternative Paths	None.
Post condition	The communication has been sent.
Exception Paths	The Editor may abandon the operation at any time.
Other	The standard copyright form will be available in the Editor's directory for attaching to the email message, if desired.
Actor	editor

3.2.6 Publish Article

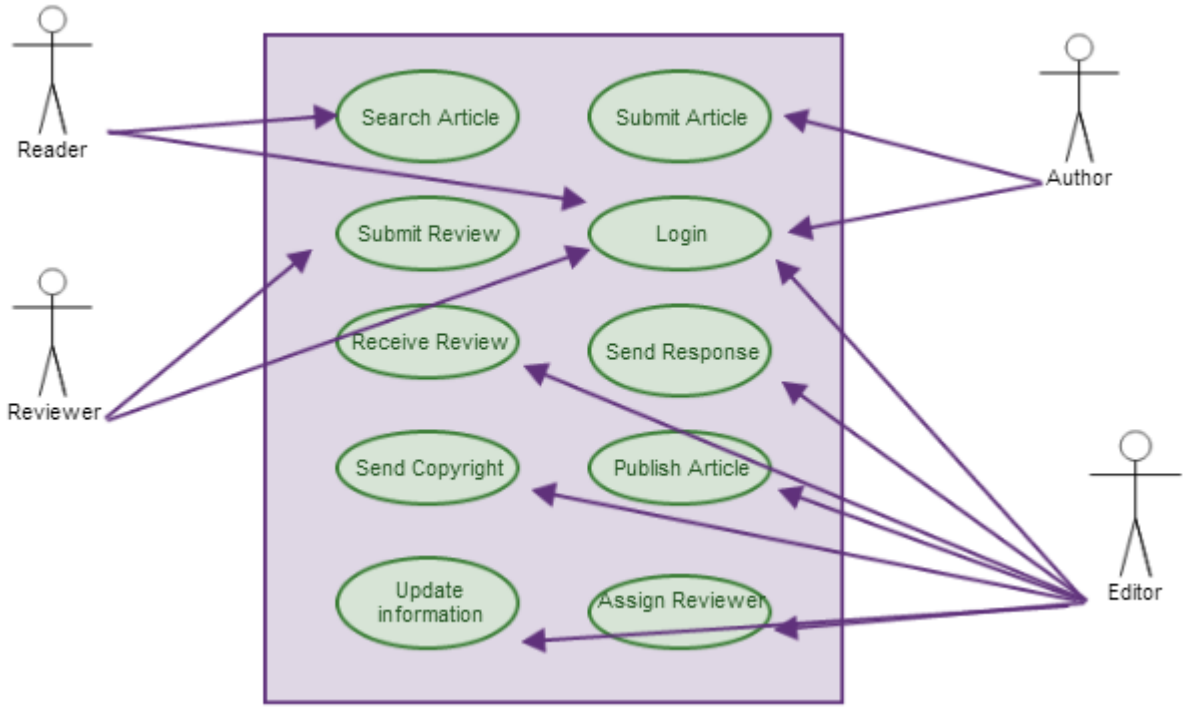
Use Case Name	Publish Article
Reference	Section 2.2.4, Publish Article
Trigger	The Editor selects to transfer an approved article to the Online Journal.
Precondition	The Editor has accessed the Article Manager main screen.
Basic Path	1. The system creates and presents an alphabetical list of the active articles that are flagged as having their copyright form returned. 2. The Editor selects an article to publish. 3. The system accesses the Online Database and transfers the article and its accompanying information to the Online Journal database. 4. The article is removed from the active article database.
Alternative Paths	None.
Post condition	The article is properly transferred.
Exception Paths	The Editor may abandon the operation at any time.
Other	Find out from the Editor to see if the article information should be archived somewhere.
Actor	editor

3.3 Use Cases

Use Case Index:

User case id	Use case name	Actor	complexity
1	Login	Reader, author, reviewer, editor	All these actors can login into already created account or signup for a new account.
2	Search article	reader	The Reader chooses how to search the Web site. The choices are by name of Author, by title, Keyword and by date.
3	Submit article	author	The author either submits an original article or resubmits an edited article.
4	Assign reviewer	editor	The Editor assigns one or more reviewers to an article after receiving Articles submitted by author.
5	Submit review	reviewer	The reviewer submits a review of an article.
6	Receive review	editor	The Editor enters a review into the system and receives reviews.
7	Send response	editor	The Editor sends a response to an Author.
8	Send copyright	editor	The Editor sends a copyright form to an Author.
9	Publish articles	editor	The Editor transfers an accepted article to the Online Journal.
10	Update information on website	editor	Editor can update the information on website.

Further description of use cases refer (3.2 Functional requirements) and use case diagram is given below:



3.4 Classes / Objects

- **3.4.1 <Class / Object #1>**

3.4.1.1 Attributes
3.4.1.2 Functions
<Reference to functional requirements and/or use cases>

3.4.2 <Class / Object #2>

3.5 Non-functional Requirements

The Online Journal will be on a server with high speed Internet capability. The physical machine to be used will be determined by the client. The speed of the Reader's connection will depend on the hardware used rather than characteristics of this system. The Article Manager will run on the editor's PC and will contain an Access database. Access should be installed on that computer and should be a Windows operating system.

The data to be stored in the internal Article Manager database is given below.

Author Data Entity

Data Item	Description
Name	Name of principle author
Email Address	Internet address
Article	Article entity

Reviewer Data Entity

Data Item	Description
Name	Name of principle author
ID	ID number of Humanity Society member
Email Address	Internet address
Article	Article entity of
Num Review	Review entity
Specialty	Area of expertise

Article Data Entity

Data Item	Description
Name	Name of Article
Author	Author entity
Other Authors	Other authors is any; else null
Reviewer	Reviewer entity
Review	Review entity
Contents	Body of article
Category	Area of content
Accepted	Article has been accepted for publication
Copyright	Copyright form has been returned
Published	Sent to Online Journal

The Structure of the data to be stored in the Online Journal database on the server is as follows.

Published Article Entity

Data Item	Description
Name	Name of Article
Author	Name of one Author
Abstract	Abstract of article
Content	Body of article
Category	Area of content

3.5.1 Performance:

The applications regarding the transaction of articles and reviews between the users (author, editor and reviewer),verification of membership of a reviewer and searches made by the reader based on author names for which data like email addresses, user names, membership(ID) numbers are used more than once, a reasonable partition of the space in directory has to be allotted .

3.5.2 Reliability:

Failure in any of the applications for the user (author, reader, editor and reviewer) will be very less probable. The user can rely on the system (only under his given constraints) to handle his input and process the output effectively. Reliability factor even goes down if many instances of data stored in database are being transacted at the same time increasing database space.

3.5.3 Availability

The software will always be available for a user. Although, the quantitative specification of it's availability can only be derived from practical usage.

3.5.4 Security:

The privacy of the user's (author, reviewer) profile information has to be taken care of.

When the author (after a whole session of reviews and editing) is finally approved of his article, editor has to provide him a copyright for the article while publishing it on the Journal and secure his originality.

The server on which the Online Journal reside will have it's own security to prevent unauthorized *write/delete* access. There is no restriction on *read* access. The use of email by an Author or Reviewer is on the client systems and thus is external to the system.

3.5.5 Maintainability

While developing the software, we make sure that it can later be subjected to minor changes. The changes are restricted to correction of defects or minor enhancement in the functionality of software. This way, we keep up with the client's maintainability requirement.

3.5.6 Portability:

The final software is expected to work on systems independent of Operating System and phones having Internet facility.

3.6 Inverse Requirements

Software will not have following functionalities:

1. Website will not link to social network accounts like Facebook, LinkedIn, twitter etc.
2. Membership ID will be assigned as per the availability of that ID.
3. There will be no way for the author to know reviewers for his/her article.

3.7 Design Constraints

3.7.1 Software Language to Use

The languages that shall be used for coding Online Journal System are php, HTML5 and MYSQL and thus source code shall follow conventions of these languages.

3.7.2 Web based Product

The computers must be equipped with web browsers such as Internet explorer, Google Chrome, Firefox. Response time for loading the website should take no longer than two minutes.

3.8 Logical Database Requirements

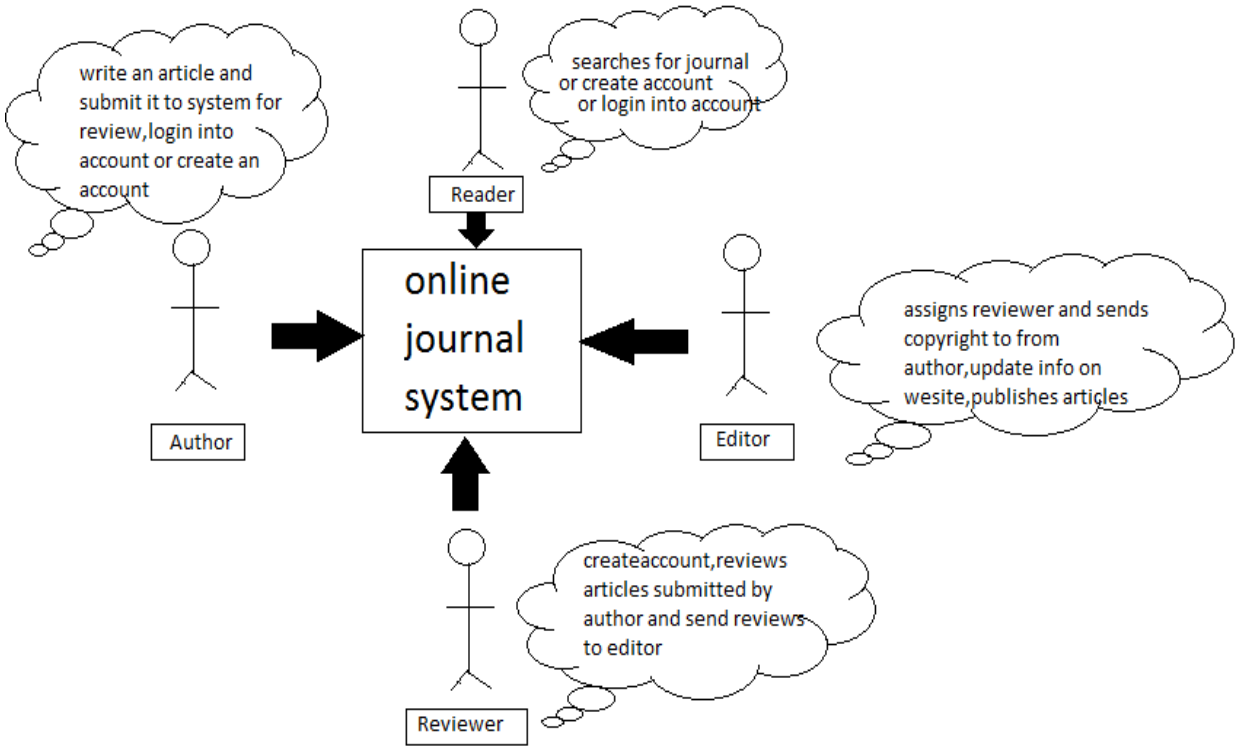
All data will be saved in the database: user accounts and profiles, article details (content, abstract, authors, date) etc. (except files which are stored on the disk.) The database allows concurrent access and will be kept consistent at all times, requiring a good database design.

3.9 Other Requirements

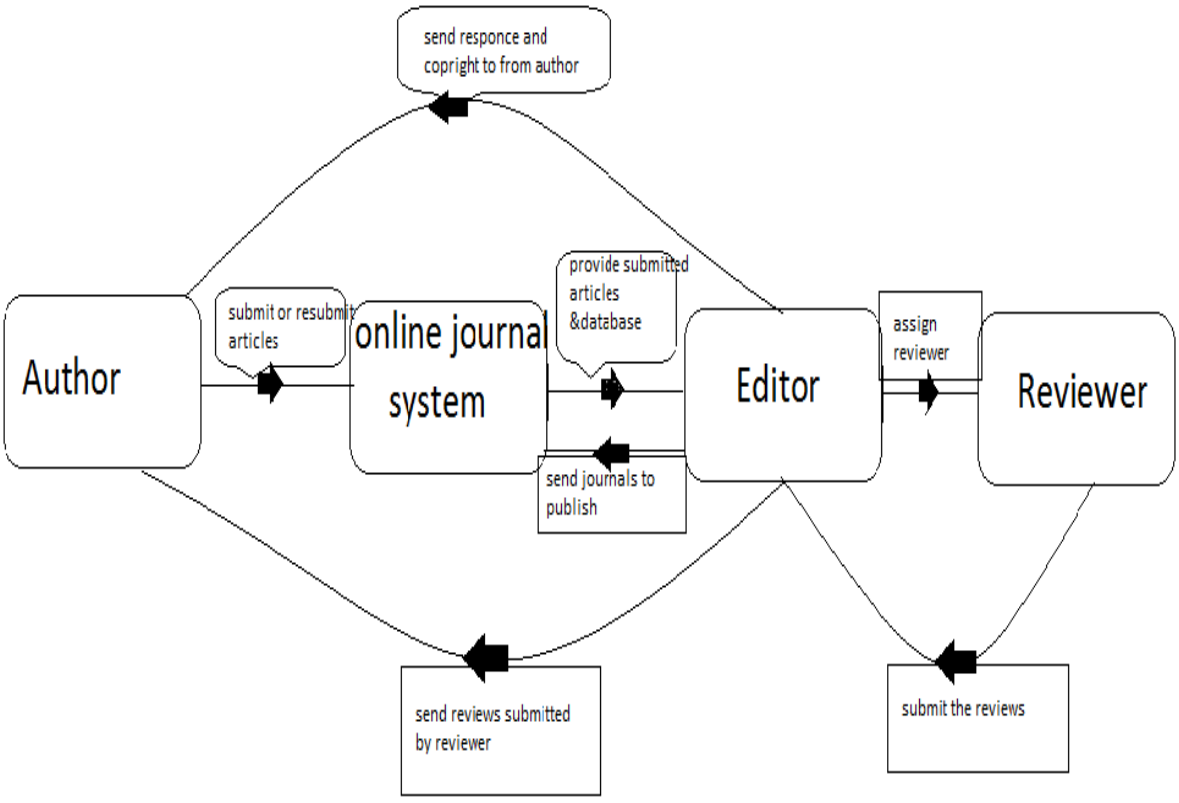
The only additional requirement will be to get a creative common license.

4. Analysis Models

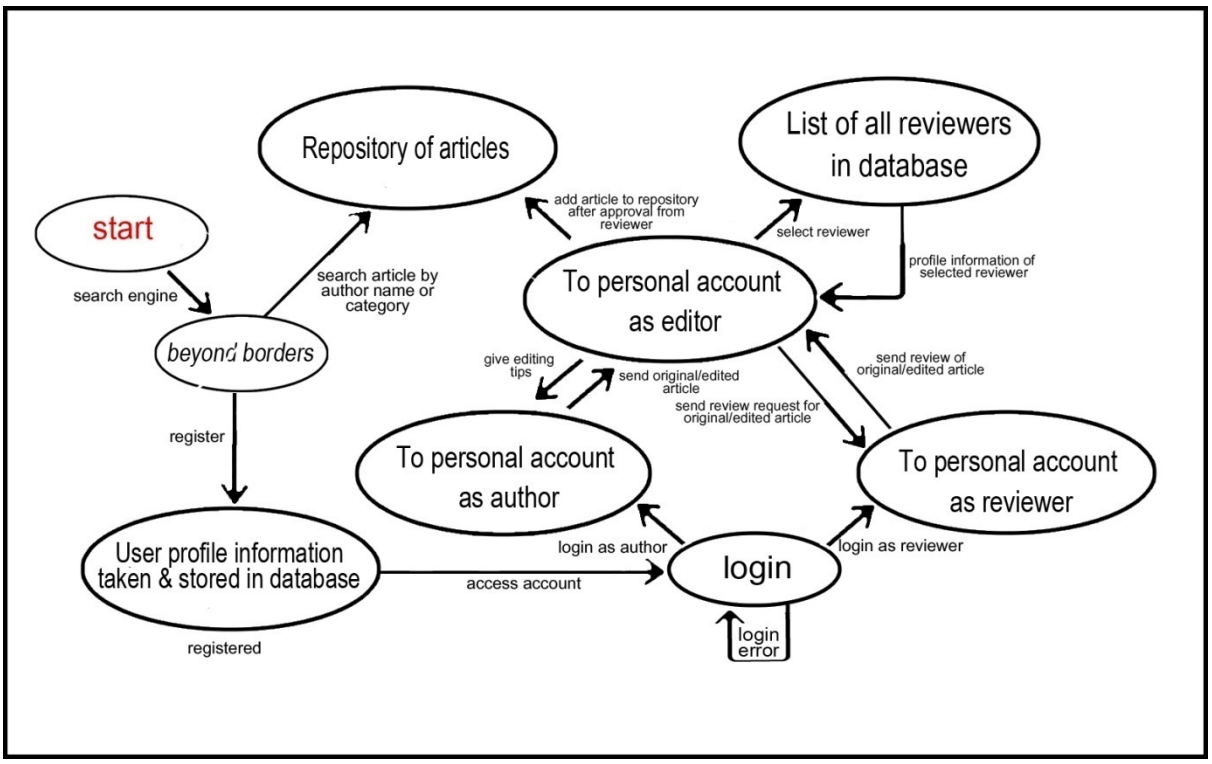
4.1 Sequence Diagrams



4.3 Data Flow Diagrams (DFD)



4.2 State-Transition Diagrams (STD)



5. Change Management Process

As a team, we will update and evaluate our SRS document every week as we make changes in our design and requirements. We will add new detailed information which will include: getting creative common license, and more specifications and requirements that we find along the way in the designing and implementation of the product. Changes to this document may be made after approval from development team and client.