Software Requirements Specification

Version 1.0

17.6.2020

TOURS AND TRAVEL MANAGEMENT SYSTEM

Arwa Alissa

Enass khwam

Dania sheikh-al-souk

# Table of Contents

[Table of Contents i](file:///C:\Users\User\Desktop\Version%201.docx#_Toc77487619)

[List of Figures ii](file:///C:\Users\User\Desktop\Version%201.docx#_Toc77487620)

[1.0. Introduction 1](file:///C:\Users\User\Desktop\Version%201.docx#_Toc77487621)

[1.1. Purpose 1](file:///C:\Users\User\Desktop\Version%201.docx#_Toc77487622)

[1.2. Scope of Project 1](file:///C:\Users\User\Desktop\Version%201.docx#_Toc77487623)

[1.3. Glossary 1](file:///C:\Users\User\Desktop\Version%201.docx#_Toc77487624)

[1.4. References 1](file:///C:\Users\User\Desktop\Version%201.docx#_Toc77487625)

[1.5. Overview of Document 2](file:///C:\Users\User\Desktop\Version%201.docx#_Toc77487626)

[2.0. Overall Description 3](file:///C:\Users\User\Desktop\Version%201.docx#_Toc77487627)

[2.1 System Environment 3](file:///C:\Users\User\Desktop\Version%201.docx#_Toc77487628)

[2.2 Functional Requirements Specification 3](file:///C:\Users\User\Desktop\Version%201.docx#_Toc77487629)

[2.2.1 System user Use Case](file:///C:\Users\User\Desktop\Version%201.docx#_Toc77487630) 3

[Use case: Validating user 3](file:///C:\Users\User\Desktop\#_Toc77487631)

[2.2.2](file:///C:\Users\User\Desktop\Version%201.docx#_Toc77487648) Authorized Customer……………………………………………………….........4

Use case: View tour details…………………………………………………………………………..4

Reservation For tour………………………………………………………………………4

Maintain Reservation……………………………………………………………………...4

Make payment…………………………………………………………………………….4

New tour information……………………………………………………………………..4

2.2.3 Staff Use case………………………………………………………………………………5

Use case: Receive Reservepackges details………………………………………………………….5

Receive Customer………………………………………………………………………..5

Provide Services…………………………………………………………………………5

2.2.4 Adminstrator Use case………………………………………………………………………..6

Use case: sign In…………………………………………………………………………………..6

Maintain tour service……………………………………………………………………6

2.3 User characteistics………………………………………………………………………………….7

[2.4 Non-Functional Requirements……………………………………………………………………..8](file:///C:\Users\User\Desktop\Version%201.docx#_Toc77487649)

[3.0. Requirements Specification 9](file:///C:\Users\User\Desktop\Version%201.docx#_Toc77487650)

[3.1 External Interface Requirements 9](file:///C:\Users\User\Desktop\Version%201.docx#_Toc77487651)

[3.2 Functional Requirements 10](file:///C:\Users\User\Desktop\Version%201.docx#_Toc77487652)

[3.2.1 Select The tour 10](file:///C:\Users\User\Desktop\Version%201.docx#_Toc77487653)

[3.3 Detailed Non-Functional Requirements 11](file:///C:\Users\User\Desktop\Version%201.docx#_Toc77487665)

[3.3.1 Logical Structure of the Data 11](file:///C:\Users\User\Desktop\Version%201.docx#_Toc77487666)

[3.3.2 Security 11](file:///C:\Users\User\Desktop\Version%201.docx#_Toc77487667)

# List of Figures

[Figure 1 - System Environment 3](#_Toc77487669)

[Figure 2 – Use Cases 3](#_Toc77487671)

[Figure 3 - Logical Structure of the Data 5](#_Toc77487672)

# 1.0. Introduction

## 1.1. Purpose

The purpose of this document is to present a detailed description of the MANAGEMENT TOURS AND TRAVEL It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate . This document is intended for both the stakeholders and the developers of the system.

## 1.2. Scope of Project

This program saves time for customers to choose the right trips for it.

## 1.3. Glossary

|  |  |
| --- | --- |
| **Term** | **Definition** |
| Authorized Customer | An authorized user is any person who has permission to view tour , maintain ,make payment and new tour information. |
| Staff | everyone who works at a management tours and travel and provide Services. |
| Database | Collection of all the information monitored by this system. |
| Adminstrator | a person who manages a computer network or system |
| System user | is similar in concept to a server **user** but resides in the **system** database as an OUser record. Like a database **user**, a **system user** is assigned roles that are comprised of resources and permissions |

## 

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

The third chapter, Requirements Specification 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 in its entirety, but are intended for different audiences and thus use different language

# 2.0. Overall Description

## 2.1 System Environment

Author

Reader

Editor

HS DB

Online Journal

Article Manager

Web Publishing System

Reviewer

Figure 1 - System Environment

The Web Publishing System has four active actors and one cooperating system.

The Author, Reader, or Reviewer accesses the Online Journal through the Internet. Any Author or Reviewer communication with the system is through email. The Editor accesses the entire system directly. There is a link to the (existing) Historical Society.

<< The division of the Web Publishing System into two component parts, the Online Journal and the Article Manager, is an example of using domain classes to make an explanation clearer. >>

## 2.2 Functional Requirements Specification

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

**Diagram:**

Reader

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**

Before this use case can be initiated, the Reader has already accessed the Online Journal Website.

1. The Reader chooses to search by author name, category, or keyword.
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.

**Xref:** Section 3.2.1, Search Article

Rewrite

Review

Active Article

Submit

Publish

Figure 2 - Article Submission Process

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

In case of multiple authors, this term refers to the *principal author*, with whom all communication is made.

#### Use case: Submit Article

Author

Submit Article

**Diagram:**

**Brief Description**

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

**Initial Step-By-Step Description**

Before this use case can be initiated, the Author has already connected to the Online Journal Website.

1. The Author chooses the *Email Editor* button.
2. The System uses the *sendto* 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.
4. The System generates and sends an email acknowledgement.

**Xref:** Section 3.2.2, Communicate

### 2.2.3 Reviewer Use Case

#### Use case: Submit Review

**Diagram:**

Reviewer

Submit Review

**Brief Description**

The reviewer submits a review of an article.

**Initial Step-By-Step Description**

Before this use case can be initiated, the Reviewer has already connected to the Online Journal Website.

1. The Reviewer chooses the *Email Editor* button.
2. The System uses the *sendto* 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.

**Xref:** Section 3.2.2, Communicate

### 2.2.4 Editor Use Cases

The Editor has the following sets of use cases:

Figure 3 - Editor Use Cases

Update Info

Editor

Handle Art

Ck Status

Send Rec

Publish Art

**Update Information use cases**

#### Use case: Update Author

Editor

Update Author

**Diagram:**

**Brief Description**

The Editor enters a new Author or updates information about a current Author.

**Initial Step-By-Step Description**

Before this use case can be initiated, the Editor has already accessed the main page of the Article Manager.

1. The Editor selects to *Add/Update Author*.
2. The system presents a choice of adding or updating.
3. The Editor chooses to add or to update.
4. If the Editor is updating an Author, the system presents a list of authors to choose from and presents a grid filling in with the information; else the system presents a blank grid.
5. The Editor fills in the information and submits the form.
6. The system verifies the information and returns the Editor to the Article Manager main page.

**Xref:** Section 3.2.3, Add Author; Section 3.2.5 Update Person

#### Use case: Update Reviewer

**Diagram:**

Editor

Update Reviewer

Hist Soc DB

**Brief Description**

The Editor enters a new Reviewer or updates information about a current Reviewer.

**Initial Step-By-Step Description**

Before this use case can be initiated, the Editor has already accessed the main page of the Article Manager.

1. The Editor selects to *Add/Update Reviewer*.
2. The system presents a choice of adding or updating.
3. The Editor chooses to add or to update.
4. The system links to the Historical Society Database.
5. If the Editor is updating a Reviewer, the system and presents a grid with the information about the Reviewer; else the system presents list of members for the editor to select a Reviewer and presents a grid for the person selected.
6. The Editor fills in the information and submits the form.
7. The system verifies the information and returns the Editor to the Article Manager main page.

**Xref:** Section 3.2.4, Add Reviewer; Section 3.2.5, Update Person

#### Use case: Update Article

**Diagram:**

Editor

Update Article

**Brief Description**

The Editor enters information about an existing article.

**Initial Step-By-Step Description**

Before this use case can be initiated, the Editor has already accessed the main page of the Article Manager.

1. The Editor selects to *Update Article*.
2. The system presents s list of active articles.
3. The system presents the information about the chosen article.
4. The Editor updates and submits the form.
5. The system verifies the information and returns the Editor to the Article Manager main page.

**Xref:** Section 3.2.6, Update Article Status

**Handle Article use cases**

#### Use case: Receive Article

Editor

Receive Article

**Diagram:**

**Brief Description**

The Editor enters a new or revised article into the system.

**Initial Step-By-Step Description**

Before this use case can be initiated, the Editor has already accessed the main page of the Article Manager and has a file containing the article available.

1. The Editor selects to *Receive Article*.
2. The system presents a choice of entering a new article or updating an existing article.
3. The Editor chooses to add or to update.
4. If the Editor is updating an article, the system presents a list of articles to choose from and presents a grid for filling with the information; else the system presents a blank grid.
5. The Editor fills in the information and submits the form.
6. The system verifies the information and returns the Editor to the Article Manager main page.

**Xref:** Section 3.2.7, Enter Communication

#### Use case: Assign Reviewer

This use case extends the *Update Article* use case.

**Diagram:**

Editor

Assign Reviewer

Hist Soc DB

**Brief Description**

The Editor assigns one or more reviewers to an article.

**Initial Step-By-Step Description**

Before this use case can be initiated, the Editor has already accessed the article using the *Update Article* use case.

1. The Editor selects to *Assign Reviewer*.
2. The system presents a list of Reviewers with their status (see data description is section 3.3 below).
3. The Editor selects a Reviewer.
4. The system verifies that the person is still an active member using the Historical Society Database.
5. The Editor repeats steps 3 and 4 until sufficient reviewers are assigned.
6. The system emails the Reviewers, attaching the article and requesting that they do the review.
7. The system returns the Editor to the *Update Article* use case.

**Xref:** Section 3.2.8, Assign Reviewer

#### Use case: Receive Review

Editor

Receive Review

This use case extends the *Update Article* use case.

**Diagram:**

**Brief Description**

The Editor enters a review into the system.

**Initial Step-By-Step Description**

Before this use case can be initiated, the Editor has already accessed the article using the *Update Article* use case.

1. The Editor selects to *Receive Review*.
2. The system presents a grid for filling with the information.
3. The Editor fills in the information and submits the form.
4. The system verifies the information and returns the Editor to the Article Manager main page.

**Xref:** Section 3.2.7, Enter Communication

**Check Status use case:**

#### Use case: Check Status

**Diagram:**

Editor

Check Status

**Brief Description**

The Editor checks the status of all active articles.

**Initial Step-By-Step Description**

Before this use case can be initiated, the Editor has already accessed the main page of the Article Manager.

1. The Editor selects to *Check Status*.
2. The system returns a scrollable list of all active articles with their status (see data description in section 3.3 below).
3. The system returns the Editor to the Article Manager main page.

**Xref:** Section 3.2.9, Check Status

**Send Recommendation use cases:**

#### Use case: Send Response

This use case extends the *Update Article* use case.

**Diagram:**

Editor

Send Response

**Brief Description**

The Editor sends a response to an Author.

**Initial Step-By-Step Description**

Before this use case can be initiated, the Editor has already accessed the article using the *Update Article* use case.

1. The Editor selects to *Send Response*.
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.
4. The system returns the Editor to the Article Manager main page.

**Xref:** Section 3.210, Send Communication

#### Use case: Send Copyright

This use case extends the *Update Article* use case.

**Diagram:**

**Brief Description**

Editor

Send Copyright

The Editor sends a copyright form to an Author.

**Initial Step-By-Step Description**

Before this use case can be initiated, the Editor has already accessed the article using the *Update Article* use case.

1. The Editor selects to *Send Copyright*.
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.
4. The system returns the Editor to the Article Manager main page.

**Xref:** Section 3.2.10, Send Communication

#### Use case: Remove Article

This use case extends the *Update Article* use case.

Editor

Remove Article

**Diagram:**

**Brief Description**

The Editor removes an article from the active category.

**Initial Step-By-Step Description**

Before this use case can be initiated, the Editor has already accessed the article using the *Update Article* use case.

1. The Editor selects to remove an article from the active database.
2. The system provides a list of articles with the status of each.
3. The Editor selects an article for removal.
4. The system removes the article from the active article database and returns the Editor to the Article Manager main page.

**Xref:** Section 3.2.12, Remove Article

**Publish Article use case:**

#### Use case: Publish Article

This use case extends the *Update Article* use case.

Editor

Publish Article

**Diagram:**

**Brief Description**

The Editor transfers an accepted article to the Online Journal.

**Initial Step-By-Step Description**

Before this use case can be initiated, the Editor has already accessed the article using the *Update Article* use case.

1. The Editor selects to *Publish Article*.
2. The system transfers the article to the Online Journal and updates the search information there.
3. The system removes the article from the active article database and returns the Editor to the Article Manager home page.

**Xref:** Section 3.2.11, Publish Article

<< Since three of the actors only have one use case each, the summary diagram only involves the Editor. Adapt the rules to the needs of the document rather than adapt the document to fit the rules. >>

## 2.3 User Characteristics

The Reader is expected to be Internet literate and be able to use a search engine. The main screen of the Online Journal Website will have the search function and a link to “Author/Reviewer Information.”

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 Windows literate and to be able to use button, pull-down menus, and similar tools.

The detailed look of these pages is discussed in section 3.2 below.

## 2.4 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 Historical Society. The software developed here assumes the use of a tool such as Tomcat for connection between the Web pages and the database. 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 is already installed on this computer and is a Windows operating system.