Software Requirements Specification

Version 1.0

21/5/2021

***University educational system***

Safaa al-kurdi

Eman al-ahmad

Enas tawil

Arwa mandow

Seba al-hossen

# 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 education details…………………………………………………………………………..4

Reservation For education ………………………………………………………………………4

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

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

New education 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 education 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 search of leacture 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 ***"University educational system"*** 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 software system will be an application for students to help them to get the information easily, This system will be designed to upload ,download lecture, education programs and exams marks, the system will meet students needs while

remaining easy to understand and use.

More specifically, this system is designed to allow for professors to manage and upload lectures , and communicate with a group of students to publish technological information articles and answer their questions. The software will facilitate communication between students and professors. The system also contains a relational

database containing a list of professors, students, and Articles.

## 1.3. Glossary

|  |  |
| --- | --- |
| **Term** | **Definition** |
| Student | An authorized user is any person who has permission to view site , search ,download ,get marks. |
| Staff | everyone who works at a management the site and a 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

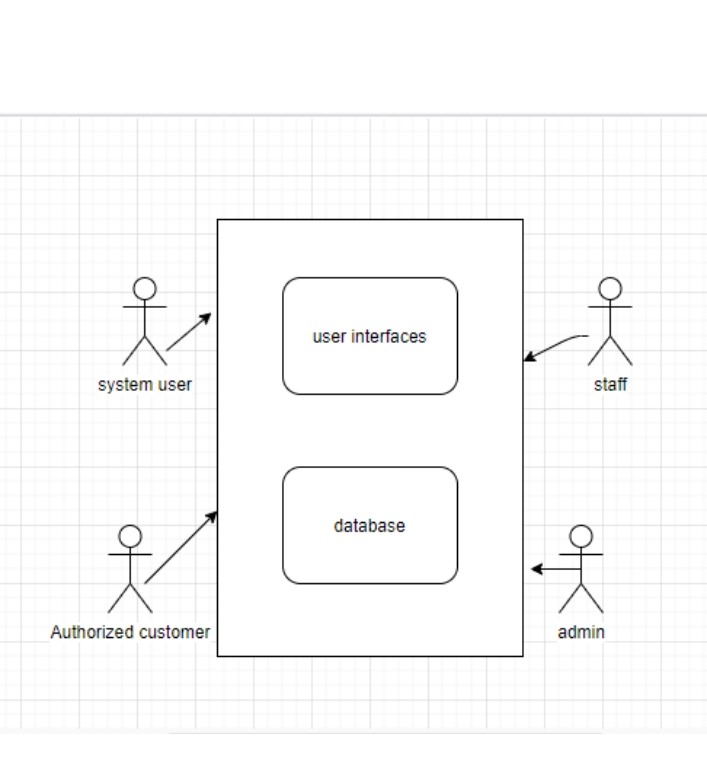
Both sections of the document describe the same software product in its entirety, but are intended for different audiences and thus use different language

# 2.0. Overall Description

## 2.1 System Environment

We have three actors in this system

The user can search for lectures and articles by interacting with the interface



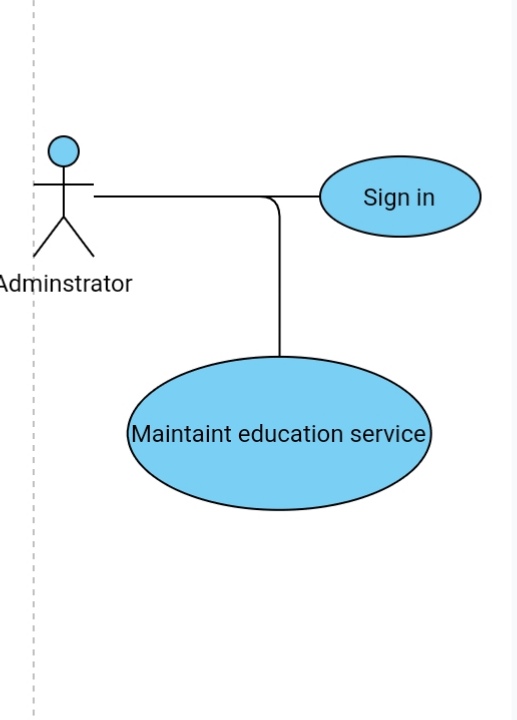
**System Environment Figure 1-**

## 2.2 Functional Requirements Specification

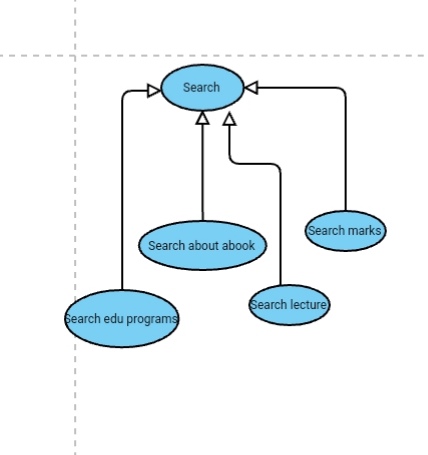
This section shows us usage cases for people who interact with the system.

### ***2.2.1***

adminstrator use case:



Use case: Business rules



**Figure use** case previouse

Brief Description:

Roles and Permissions Matrices are grids that define all of the possible user roles, system operations, and the specific permissions on those operations by role.

Use case: receives ,sends articles and lecture.

Brief Description:

receives, sends articles and lecture for review, and makes final judgments for publications.

**Initial Step-By-Step Description**

1. The user arrives at the website on the Interne

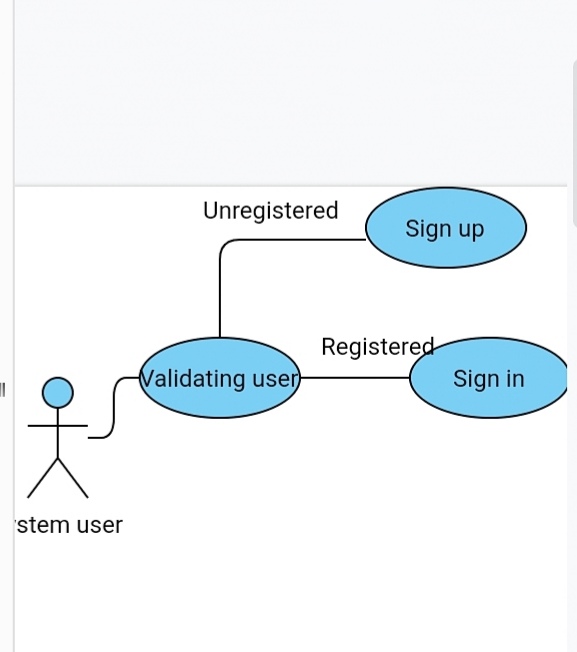
2. Lectures and articles are uploaded to the site

3. The lectures and articles submitted on the site are reviewed and revised

Upload it to the site

**2.2.2**

Student use case:



Use case: sign-in (login), sign-out (log out).

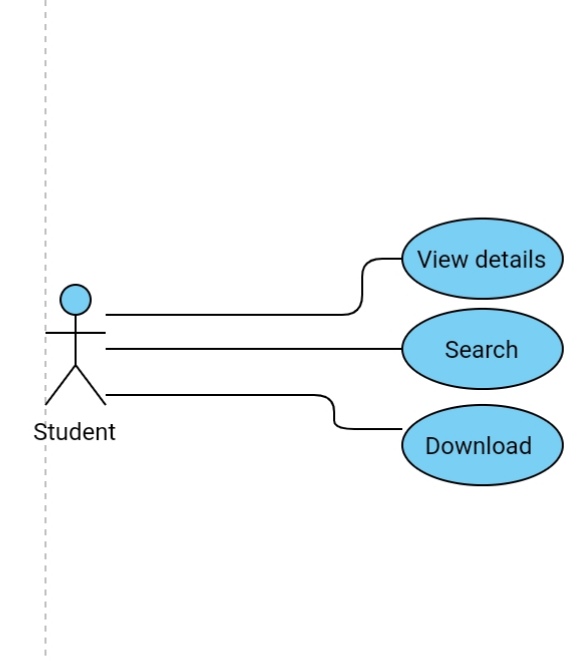
Brief Description:

Students Log in or log out Application to get what they need.

**Initial Step-By-Step Description**

1. Student access to the University educational system website on the Internet
2. He logs into the application
3. He chooses the lecture or article he wants
4. He logs out of the app when he gets what he wants

Use case: download lecture.



Brief Description:

The student loads the lecture that given by the professor.

**Initial Step-By-Step Description**

After logging into the application, the student chooses the lecture by searching for its name

The system displays a list of lectures for the selected course

The student chooses the lecture he wants

The student downloads the lecture from the application

## 2.3 User Characteristics

The application user is expected to be familiar with the Internet, be able to use the search engine, and be familiar with the use of e-mail

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

***2.4 Non-Functional Requirements***

***Reliability:*** This quality attribute specifies how likely the system or its element would run without a failure for a given period of time under predefined conditions.

***Maintainability:*** Maintainability defines the time required for a solution or its component to be fixed, changed to increase performance or other qualities, or adapted to a changing environment.

***Security:*** all data inside the system or its part will be protected against malware attacks or unauthorized access.

**3.0. Requirements Specification**

**3.1 External interface requirements**

**The only link to an external system is the association with the Historical Society Database (HS) and used by the student’s membership. The student believes that he or she is a potential member of the community as an effective reviewer and has imposed a condition of membership to the auditor. HS database fields of interest to the Member Name University educational system and Membership Number (ID) and player name (optional field for the HS database).**

**The Use Case Set References sends the Reviewer ID to the HS database, and a Boolean value indicating the membership status is used. The updating status of references requires a list of member names, membership numbers when adding new references. It returns a boolean value for membership status when updating references.**

**3.2 Job requirements**

**The logical structure of the data is presented in Section 3.3.1.**

**3.2.1 Register a customer**

|  |  |
| --- | --- |
| **Use Case Name** | **Register a customer** |
| **XRef** | **Section 2.2.1, Register a customer**  **SDD, Section 7.1** |
| **Trigger** | **Add student information to the database** |
| **Precondition** | **Information entered into database is not repeated** |
| **Basic Path** | 1. **The student will log into the application** 2. **He enters his personal data necessary to be added to the database** 3. **He uploads the lecture he wants** |
| **Alternative Paths** | **If the student has downloaded the lecture more than once, the system will inform him** |
| **Postcondition** | **The student gets an electronic account in the application** |
| **Exception Paths** | **The student may log out** |
| **Other** | **A list of the names of the users who downloaded the lectures is created** |

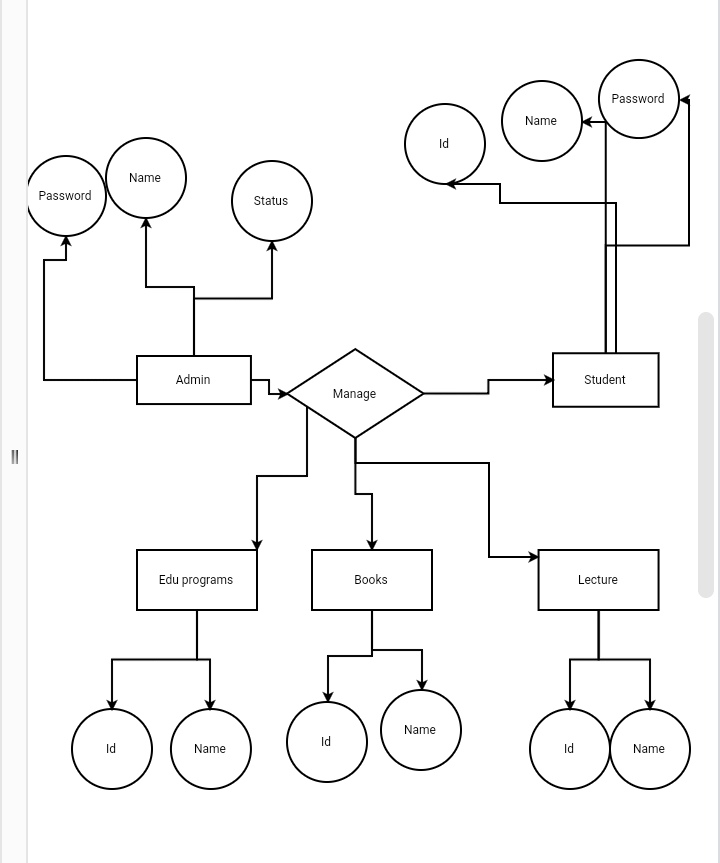
* + 1. **Download the lecture**

|  |  |
| --- | --- |
| **Use Case Name** | * + 1. **Download the lecture** |
| **XRef** | **Section 2.2.3, Submit Review**  **SDD, Section 7.2** |
| **Trigger** | **Get the lecture chosen by the student** |
| **Precondition** | **The student has an online account on the application** |
| **Basic Path** | 1. **The student chooses the lecture he wants** 2. **He downloads the lecture from the site** |
| **Alternative Paths** | **1. If the student chooses to search by category, the system creates and presents a list of all categories in the database**  **2. The student chooses the category.**  **3. The system creates and submits a list of all the lectures in that category in the database.**  **4. If the student chooses to search by keywords, the system presents a dialog box to enter the keyword or phrase.**  **5. The student enters a keyword or phrase.**  **6. The system searches the abstracts for all the lectures that contain that keyword or phrase, and creates and submits a list of all these lectures in the database.** |
| **Postcondition** | **The selected lectures is downloaded to the client machine.** |
| **Exception Paths** | **None** |
| **Other** | **None** |

## 3.3 Detailed Non-Functional Requirements

### 3.3.1 Logical Structure of the Data

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



**Figure3- Logical structure of the data**

**Student Data Entity**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| Name | Text | Name of principle author |  |
| Email Address | Text | Internet address |  |
| Lecture | Pointer | Lecture entity | May be several |

**Reviewer Data Entity**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| Name | Text | Name of principle author |  |
| ID | Integer | ID number of Historical Society member | Used as key in Historical Society Database |
| Email Address | Text | Internet address |  |
| Lecture | Pointer | Lecture entity of | May be several |
| Num Review | Integer | Review entity | Number of not returned reviews |
| History | Text | Comments on past performance |  |
| Specialty | Category | Area of expertise | May be several |

**Review Data Entity**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| Lecture | Pointer | Lecture entity |  |
| Reviewer | Pointer | Reviewer entity | Single reviewer |
| Date Sent | Date | Date sent to reviewer |  |
| Returned | Date | Date returned; null if not returned |  |
|  |  | Returned |  |
| Contents | Text | Text of review |  |

**Lecture Data Entity**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| Name | Text | Name of Article |  |
| Student | Pointer | Student entity | Name of principle author |
| Other Student | Text | Other student is any; else null | Not a pointer to an Student entity |
| Reviewer | Pointer | Reviewer entity | Will be several |
| Review | Pointer | Review entity | Set up when reviewer is set up |
| Contents | Text | Body of Lecture | Contains Abstract as first paragraph. |
| Category | Text | Area of content | May be several |
| Accepted | Boolean | Lecture has been accepted for publication | Needs Copyright form returned |
| Copyright | Boolean | Copyright form has been returned | Not relevant unless Accepted is True. |
| Published | Boolean | Sent to Online Journal | Not relevant unless Accepted is True. Lecture is no longer active and does not appear in status checks. |

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

**Published** **Lecture Entity**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| Name | Text | Name of Article |  |
| Author | Text | Name of one student | May be several |
| Abstract | Text | Abstract of article | Used for keyword search |
| Content | Text | Body of article |  |
| Category | Text | Area of content | May be several |

### 3.3.2 Security

**The server on which the University educational system resides will have its own security to prevent unauthorized write/delete access. There is no restriction on read access. The use**

**of email by an Student or Reviewer is on the client systems and thus is external to the system.**

**The PC on which the Article Manager resides will have its own security. Only the Editor will have physical access to the machine and the program on it. There is no special protection built into this system other than to provide the editor with write access to the University educational system to publish an lecture.**

# Index

|  |  |
| --- | --- |
| Abstract, 6, 17, 27  add, 9, 11, 19, 20, 21  Add, 8, 9, 19  Licture, 1, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28  Lecture Manager, 5, 8, 9, 10, 11, 12, 13, 14, 15, 16, 19, 20, 21, 22, 23, 24, 25, 28  Author, 1, 4, 5, 6, 7, 8, 9, 13, 14, 16, 17, 19, 20, 22, 23, 25, 26, 27  Category, 5, 14, 17, 18, 20, 21, 23, 26, 27  Database, 2, 9, 11, 14, 15, 16, 17, 18, 19, 20, 21, 22, 24, 25, 26, 27  Editor, 1, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 19, 20, 21, 22, 23, 24, 25, 28  Field, 17, 19, 20 | Form, 1, 6, 9, 10, 11, 12, 14, 19, 20, 21, 23, 24, 27  Grid, 9, 11, 12, 19, 20, 21  Historical Society, 1, 5, 9, 11, 16, 17, 19, 20, 26  **University educational system**, 4, 5, 6, 7, 15, 16, 17, 18, 24, 27, 28  Reader, 4, 5, 6, 16, 17, 18  Review, 1, 7, 11, 12, 18, 21, 23, 26, 27  Reviewer, 1, 4, 5, 6, 7, 9, 11, 16, 17, 19, 20, 21, 22, 23, 26, 27  Security, 27, 28  Status, 11, 12, 13, 14, 17, 21, 22, 23, 27  update, 9, 11, 20, 21  Update, 8, 9, 10, 11, 12, 13, 14, 15, 17, 19, 20, 21, 22  User, 7, 16, 18  Web Publishing System, 1, 4, 5, 17 |