Software Requirements Specification

Version 1.0

<<Annotated Version>>

June 17, 2020

School Management System

Kawthar klayshe

Rose alsulieman

Batool alali

Submitted in partial fulfillment

Of the requirements of

CS 310 Software Engineering

# Table of Contents

[Table of Contents i](#_Toc77487619)

[List of Figures ii](#_Toc77487620)

[1.0. Introduction](#_Toc77487621)

[1.1. Purpose](#_Toc77487622)

[1.2. Scope of Project](#_Toc77487623)

[1.3. Glossary](#_Toc77487624)

[1.4. References](#_Toc77487625)

[1.5. Overview of Document](#_Toc77487626)

[2.0. Overall Description](#_Toc77487627)

[2.1 System Environment](#_Toc77487628)

[2.2.1 Manager Use Case](#_Toc77487630)

[Use case: update Time Table](#_Toc77487631)

[Use case: update Teacher](#_Toc77487631)

[Use case: add an absent record for a teacher](#_Toc77487631)

[2.2.2 Teacher Use Case](#_Toc77487632)

[Use caseupdate marks record](#_Toc77487633)

[Use casesend an alarm](#_Toc77487633)

[2.2.3 Student Use Case](#_Toc77487634)

[Use case: View marks](#_Toc77487635)

[Use case:View time table](#_Toc77487635)

[2.2.4 Parents Use Cases](#_Toc77487636)

[Use case:connect with school](#_Toc77487635)

[2.3 User Characteristics](#_Toc77487648)

[2.4 Non-Functional Requirements](#_Toc77487649)

[3.0. Requirements Specification](#_Toc77487650)

[3.1 External Interface Requirements](#_Toc77487651)

[3.2 Functional Requirements](#_Toc77487652)

[3.2.1 Update person](#_Toc77487653)

[3.2.2 Add teacher](#_Toc77487654)

[3.2.3 Add student](#_Toc77487655)

[3.2.4 Update time table for teacher](#_Toc77487656)

[3.2.5 Update absent record for teacher](#_Toc77487657)

[3.2.6 Update mark record](#_Toc77487658)

[3.2.7 Send alarm](#_Toc77487659)

[3.2.8 Connect with school](#_Toc77487660)

[3.2.9 View mark record](#_Toc77487661)

[3.2.10 View time table](#_Toc77487661)

[3.3 Detailed Non-Functional Requirements](#_Toc77487665)

[3.3.1 Security](#_Toc77487667)

# 

# 1.0. Introduction

## 1.1. Purpose

The purpose of this document is to present a detailed description of the school Management 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 and how the system will react to external stimuli. This document is intended for both the stakeholders and the developers of the system and will be proposed to the Regional Historical Society for its approval..

## 1.2. Scope of Project

This software system will be a fast and accurate system for performing tasks ,Here we find a description of the school management system and upon the request of the customer, it was developed to fully automate the work and give the computer a big role .This system facilitates the work between the manager and employees and communication with the parents

Communication between the four users is done through this application.

## 1.3. Glossary

|  |  |
| --- | --- |
| **Term** | **Definition** |
| Manager | A person who manages the system and gives permission to users. |
| 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. |
| Database | Collection of all the information monitored by this system. |
| Stakeholder | Any person with an interest in the project who is not a developer. |
| User | Manager , Teacher , Students and parents. |
| Historical Society Database | The existing membership database (also HS database). |
| Member | A member of the Historical Society listed in the HS database. |
| Use case | A description of steps or procedures between the user and the software system that leads the user to something useful. |

## 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**

Student

Teacher

Parent

Manager

HS DB

School Management System

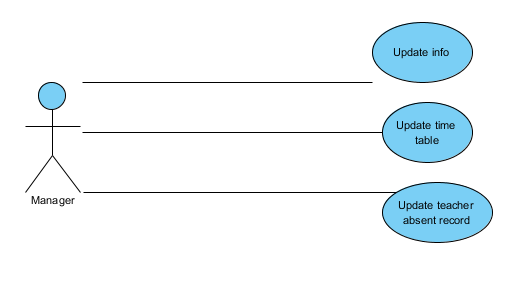
There are four actors in this system: the manager, students, parents, and teachers. All actors, including the manager, who is the system administrator, can access the system by email. There is a link with HS to verify logins

## 2.2 Functional Requirements Specification

This section outlines the use cases for each of the actors: Manager, Teacher, Student and Parent .Each one of them can perform several different roles during the use of the system.

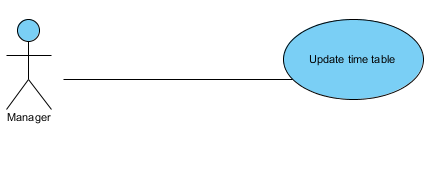
2.2.1 Manager Use Case

The manager has multiple use cases

****

**Use case:** update Time Table

**Diagram:**

****

**Brief Description:**

The Manager updates the time table for a teacher.

**Initial-step-by-step-description:**

1. The manager selects to update time table for a teacher

2. The system presents the time table for each teacher

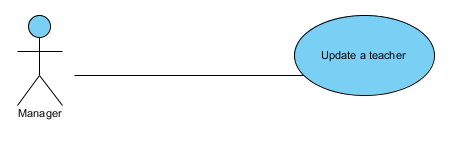
3. The manager fills in the data card with the required data.

4. Finally, the manager press “add” button to save changes

**XREF**3.2.4Update Time Table for Teacher

**Use case**: Update a teacher

**Diagram:**

****

**Brief Description:**

The manager enters a new teacher or updates an existing one**.**

**Initial-step-by-step-description:**

1. The manager selects to add/update a teacher

2. The system presents a choice of adding or updating

3. The manager chooses to add/update.

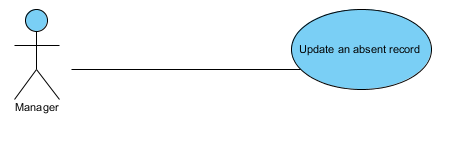
4. If the manager is updating a teacher, the system presents a list of teachers name to choose from and modify his information, else the system presents a blank grid.

5. The manager fills in the information and save form.

**XREF:** 3.2.1 Update Person, 3.2.2 Add Teacher

**Use case:** update an absent record for a teacher

**Diagram:**

****

**Brief Description:**

The manager updates the absent record for a teacher

**Initial-step-by-step-description:**

1.The manager selects to update an absent record

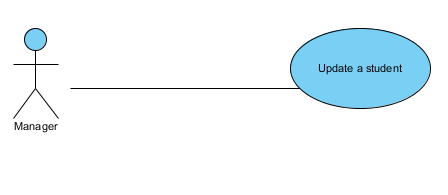
2.The system presents a list of absent records for teachers

3. The manager selects one of the records to update it.

**XREF:** 3.2.5 Update an absent record

**Use case:** update a student

**Diagram:**

****

**Brief description:**

The manager enters a new student or updates an existing one**.**

**Initial-step-by-step-description:**

1. The manager selects to add/update a student

2. The system presents a choice of adding or updating

3. The manager chooses to add/update..

4. If the manager is updating a teacher, the system presents a list of teachers name to choose from and modify his information, else the system presents a blank grid.

5. The manager fills in the information and save form.

**XREF:** 3.2.1 Update Person, 3.2.3 Add student

2.2.2 Teacher Use Cases

The teacher also has multiple use cases.



**Use case:** update marks record

**Diagram:**

****

**Brief Description:**

The teacher enters marks for each student

**Initial-step-by-step-description:**

1. The teacher selects to update marks record

2. The system present a list of mark records

3. The teacher chooses a record to modify it

**XREF:** 3.2.6 Update mark record

**Use case:** send an alarm

**Diagram:**

****

**Brief description:**

The Teacher sends an alarm to parent

**Initial-step-by-step-description:**

1 .The Teacher selects to send an alarm

2. The system calls email system

3. The teacher enters the parent e-mail address in the recipient line

4. The teacher fills out the warning message and sends it

**XREF:** 3.2.7 Send alarm

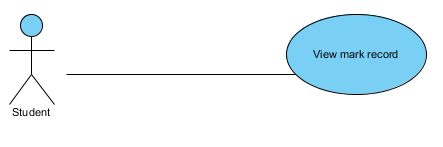
2.2.3 Student Use Case

The student has at least two use cases



**Use case:** View marks

**Diagram:**

****

**Brief description:**

The student views his mark record

**Initial-step-by-step-description:**

1 .The student selects to view his mark record by id

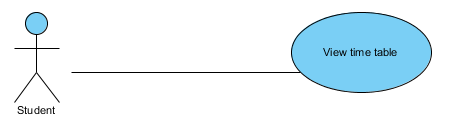
2. The system shows his mark record.

3. The student presses “download” button if he wants to download his mark record

**XREF:** 3.2.9 View Mark Record

**Use case:** View time table

**Diagram:**

****

**Brief description:**

The student views his time table

**Initial-step-by-step-description:**

1 .The student selects to view time table

2. The system shows a list of time table

3. The student enters to his time table

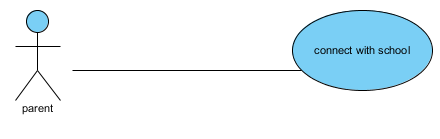
4. The student press “download” button if he wants to download his time table

**XREF:** 3.2.10 View Time Table

2.2.4Parent Use Case

**Use case:** connect with school

**Diagram:**

****

**Brief description:**

Parents connect with school to follow their children in school

**Initial-step-by-step-description:**

1. The parent chooses to connect with school

2. The system calls the email system.

3. Parent enters the teacher e-mail and the message

4. Finally, he sends the message

**XREF:** 3.2.8 Connect With School

## 2.3 User Characteristics:

The manager is expected to be able to fully manage and control the system, so he can assign permissions to another users.

Teacher, Student and Parent are expected to be able to use e-mail with attachments.

## 2.4 Non-Functional Requirements:

The System must be easy to use, characterized by fast processing and high reability in data preservation, and has a high security system.

**3.0. Requirements specifications**

**3.1 External Interface Requirements**

The link is the link to the HS database because the system needs to verify the membership of the student or employee of the school in order to log in to the system.

As the system requires the student or employee number with the email address every time he wants to enter the system

**3.2 Functional Requirements**

**3.2.1 update Time Table**

|  |  |
| --- | --- |
| **Use Case Name** | update Time Table |
| **XRef** | Section 2.2.1, update Time Table |
| **Trigger** | The principal modifies the teacher’s timeline |
| **Precondition** | The manager logs his entry into the system |
| **Basic Path** | 1. The principal chooses to update the schedule of the teacher  2. The system provides a search for the time record by number  3. The principal enters the teacher number  4. The system presents the teacher’s schedule, which is in the form of a schedule  5. The teacher time sheet is opened and filled  6. The manager chooses the Confirm Update Up date button |
| **Alternative Paths** | None |
| **Postcondition** | The specified schedule for the teacher is updated. |
| **Exception Paths** | The manager may abandon the process at any time. |
| **Other** | The timetable includes the following: (today, class, division, time of classes start) |

**3.2.2 Update person**

|  |  |
| --- | --- |
| **Use Case Name** | Update a person |
| **XRef** | Section 1 2.2Update a teacher  Section update a student2.2.1 |
| **Trigger** | The manager chooses to update a user's information (teacher, student ...) |
| **Precondition** | Administrator login required to the system |
| **Basic Path** | 1. The manager chooses to update a user's information (teacher, student ..)  2. The system provides a list alphabetically according to the selected category.  3. The manager chooses someone to adjust his data  4. The primary listing is opened and modified.  5. The manager chooses the Confirm Update Update button. |
| **Alternative Paths** | None |
| **Postcondition** | The modifications are saved in the database |
| **Exception Paths** | The manager may abandon the process at any time. |
| **Other** | None. |

**3.2.3 update an absent record for a teacher**

|  |  |
| --- | --- |
| **Use Case Name** | update an absent record for a teacher |
| **XRef** | Section 2.2.1  update an absent record for a teacher |
| **Trigger** | The principal opens and updates teacher records for teachers |
| **Precondition** | Log in to the system |
| **Basic Path** | 1. The principal chooses to update the teacher’s absenteeism record.  2. The system requires entering the unique teacher number to facilitate the search process  3. The principal enters the teacher’s number  4. The system displays the absence record for the teacher in the form of a schedule  5. The teacher's absence record is opened and modified.  6. The manager selects the Update Confirmation button and saves the changes. |
| **Alternative Paths** | In step 4, the record may be blank and added to or errors in it may be modified.  The form is ready to be filled out or modified. |
| **Postcondition** | Teacher absence log is updated. |
| **Exception Paths** | The manager may abandon the process at any time. |
| **Other** | Includes record of absence (absence date, absence status, notes) |

3.2.4 update marks record

|  |  |
| --- | --- |
| **Use Case Name** | update marks record |
| **XRef** | Section 2.2.2  update marks record |
| **Trigger** | The teacher opens and updates student records |
| **Precondition** | Log in to the system |
| **Basic Path** | 1. The teacher chooses to update the student grade record.  2. The system requires that the student's ID number be entered  3. The teacher enters the student number  4. The system provides the student's record, which is in the form of a schedule  5 The student's score book will be opened and filled out.  6. The teacher chooses the Confirm Update button and saves the changes. |
| **Alternative Paths** | None. |
| **Postcondition** | Student score book is updated. |
| **Exception Paths** | The teacher may abandon the process at any time. |
| **Other** | The scores record includes the following: (subject, study score, quarterly exam score, final result) |

**3.2.5 send an alarm**

|  |  |
| --- | --- |
| **Use Case Name** | send an alarm |
| **XRef** | Section 2.2.2  send an alarm |
| **Trigger** | The teacher sends an ultimatum to the parent |
| **Precondition** | Teacher login to the system |
| **Basic Path** | The teacher selects the contact list to direct to the email system and to send the alarm |
| **Alternative Paths** | None |
| **Postcondition** | The message is sent to the guardian. |
| **Exception Paths** | The teacher may abandon the process at any time. |
| **Other** | The teacher can enter the attachments in his message (file of signs ... file of absence ... etc) |

**3.2.6: View marks**

|  |  |
| --- | --- |
| **Use Case Name** | View marks |
| **XRef** | Section 2.2.3  View marks |
| **Trigger** | The student opens and displays the record of marks |
| **Precondition** | Student login to the system |
| **Basic Path** | 1. The student enters the list of "Student Services"  2. The menu contains several options from which to choose the "Grade Record View".  3. The system requires entering its number to display its record.  4 The student opens his score record  5 The student clicks the “Upload” button to obtain the transcript file  6. The system provides the student score record as a PDF |
| **Alternative Paths** | None |
| **Postcondition** | The scores file is downloaded to the student's machine. |
| **Exception Paths** | The student may abandon the process at any time. |
| **Other** | None |

**3.2.7: View time table**

|  |  |
| --- | --- |
| **Use Case Name** | View time table |
| **XRef** | Section 2.2.3  View time table |
| **Trigger** | The student opens and displays his schedule |
| **Precondition** | Log in to the system |
| **Basic Path** | 1. The student enters the list of "student services".  2. The student selects from the list the option "View Schedule"  3. The system requires entering the class and the division  4. The system presents the student's schedule in the database.  5 .Click the "Download" button to get his schedule  6. The system provides the time record as a PDF |
| **Alternative Paths** | None |
| **Postcondition** | The student's schedule is downloaded to his device. |
| **Exception Paths** | The student may abandon the process at any time. |
| **Other** | None |

**3.2.8: connect with school**

|  |  |
| --- | --- |
| **Use Case Name** | connect with school |
| **XRef** | Section 2.2.4  connect with school |
| **Trigger** | Parents contact the school administration |
| **Precondition** | Log in to the software system |
| **Basic Path** | The student’s guardian chooses the “Communication” menu to be directed to the email system |
| **Alternative Paths** | None |
| **Postcondition** | The message is sent to the specified teacher. |
| **Exception Paths** | The student’s parent may abandon the process at any time. |
| **Other** | None |

**3.2.9: add a teacher**

|  |  |
| --- | --- |
| **Use Case Name** | add a teacher |
| **XRef** | Section 2.2.1 update a teacher |
| **Trigger** | The manager selects to add a new teacher to the database |
| **Precondition** | The manager has to log in to the system and has accessed to the system manager screen. |
| **Basic Path** | 1-The system presents a blank grid to enter the teacher information.  2.The manager enters the information and submits the form  3.The system checks that the fields are not blank and updates the database. |
| **Alternative Paths** | If in step 2, either field is blank, the manager is instructed to add an entry. |
| **Postcondition** | The teacherhas been added |
| **Exception Paths** | The manager may abandon the operation at any time. |
| **Other** | The teacher information includes many fields to fill it such as: name, id ,email address ,phone,date…..etc |

**3.2.10: add a student**

|  |  |
| --- | --- |
| **Use Case Name** | add a student |
| **XRef** | Section 2.2.1 update a student |
| **Trigger** | The manager selects to add a new student to the database |
| **Precondition** | The manager has to log in to the system and has accessed to the system manager screen. |
| **Basic Path** | 1-The system presents a blank grid to enter the student information.  2.The manager enters the information and submits the form  3.The system checks that the fields are not blank and updates the database. |
| **Alternative Paths** | If in step 2, either field is blank, the manager is instructed to add an entry. |
| **Postcondition** | The teacherhas been added |
| **Exception Paths** | The manager may abandon the operation at any time. |
| **Other** | The student information includes many fields to fill it such as: name ,id ,email address ,phone,date….etc |

**3.3 Detailed Non-Functional Requirements**

**3.3.1 Security**

The system will have its own security to prevent unauthorized access to writing or modifying the private database for both students and staff.

No one can access the system if it is not registered by the administrator in a database.