Software Requirements Specification

Version 1.0

June 20/6/2020

Electronic Games hall System

Ghanem darboule

Zeina Al shamaa

Ahmad al ahmad

Ali al kaseem

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 1](#_Toc77487621)

[1.1. Purpose 1](#_Toc77487622)

[1.2. Scope of Project 1](#_Toc77487623)

[1.3. Glossary 2](#_Toc77487624)

[1.4. References 2](#_Toc77487625)

[1.5. Overview of Document 2](#_Toc77487626)

[2.0. Overall Description 4](#_Toc77487627)

[2.1 System Environment 4](#_Toc77487628)

[2.2 Functional Requirements Specification 5](#_Toc77487629)

[2.2.1 Reader Use Case 5](#_Toc77487630)

[Use case: Search Article 5](#_Toc77487631)

[2.2.2 Author Use Case 6](#_Toc77487632)

[Use case: Submit Article 6](#_Toc77487633)

[2.2.3 Reviewer Use Case 7](#_Toc77487634)

[Use case: Submit Review 7](#_Toc77487635)

[2.2.4 Editor Use Cases 8](#_Toc77487636)

[Use case: Update Author 8](#_Toc77487637)

[Use case: Update Reviewer 9](#_Toc77487638)

[Use case: Update Article 9](#_Toc77487639)

[Use case: Receive Article 10](#_Toc77487640)

[Use case: Assign Reviewer 11](#_Toc77487641)

[Use case: Receive Review 11](#_Toc77487642)

[Use case: Check Status 12](#_Toc77487643)

[Use case: Send Response 12](#_Toc77487644)

[Use case: Send Copyright 13](#_Toc77487645)

[Use case: Remove Article 14](#_Toc77487646)

[Use case: Publish Article 14](#_Toc77487647)

[2.3 User Characteristics 15](#_Toc77487648)

[2.4 Non-Functional Requirements 15](#_Toc77487649)

[3.0. Requirements Specification 17](#_Toc77487650)

[3.1 External Interface Requirements 17](#_Toc77487651)

[3.2 Functional Requirements 17](#_Toc77487652)

[3.2.1 Search Article 17](#_Toc77487653)

[3.2.2 Communicate 18](#_Toc77487654)

[3.2.3 Add Author 18](#_Toc77487655)

[3.2.4 Add Reviewer 19](#_Toc77487656)

[3.2.5 Update Person 19](#_Toc77487657)

[3.2.6 Update Article Status 20](#_Toc77487658)

[3.2.7 Enter Communication 20](#_Toc77487659)

[3.2.8 Assign Reviewer 21](#_Toc77487660)

[3.2.9 Check Status 21](#_Toc77487661)

[3.2.10 Send Communication 22](#_Toc77487662)

[3.2.11 Publish Article 22](#_Toc77487663)

[3.2.12 Remove Article 23](#_Toc77487664)

[3.3 Detailed Non-Functional Requirements 23](#_Toc77487665)

[3.3.1 Logical Structure of the Data 23](#_Toc77487666)

[3.3.2 Security 25](#_Toc77487667)

[Index 26](#_Toc77487668)

# List of Figures

[Figure 1 - System Environment 4](#_Toc77487669)

[Figure 2 - Article Submission Process 6](#_Toc77487670)

[Figure 3 - Editor Use Cases 8](#_Toc77487671)

[Figure 4 - Logical Structure of the Article Manager Data 23](#_Toc77487672)

# 1.0. Introduction

## 1.1. Purpose

The purpose of this document is to provide a detailed description of the electronic games hall system. It will explain the purpose and features of the system, interfaces

The system, what the system will do, and the constraints it must operate under

How the system will react to external stimuli. This document is for both

Stakeholders and system developers.

## 1.2 Scope of Project

­This system helps to operate at the comfort of the lounge manager so that he is able to know what is happening in the hall in addition to controlling it remotely and abandoning the workforce (employees) so that it gives up the workers wages at the end of each month

can communicate with the manager and have booked a specific game type , Or cancel a reservation, or switch the type of game already booked, and also makes it easy for players to play comfortably or change their playing time, as it also provides linking this system to a database containing the names of all the players who come to the hall, because there are facilities for them.

## 1.3. Glossary

|  |  |
| --- | --- |
| **Term** | **Definition** |
| Player | Anyone visiting hall. |
| Manager | Person who control the System. |
| Database | Collection of all the information monitored by this system. |
| Game card | The card which the player can play in the hall. |

## 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

Figure 1 - System Environment

Maneger

DB

Register a customer

Electronic hall System

Playar

This system consists of two main sides: the manager and the customer. Therefore, the customer through his call reserves an appointment or cancellation as mentioned previously .. In return, the manager makes sure if there are vacant devices and save the data of all players in a database for this system .

Book a game

## 2.2 Functional Requirements Specification

This section outlines the use cases for each of the active in the . 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 Maneger Use Case

#### Use case: Maneger

**Diagram:**

Maneger

Reservation game

Reserving a specific game that is contacted with the manager to determine a specific game

### 2.2.2 Maneger Use Case

**Brief Description**

### The manager is directly contacted

### The customer tells the manager what type of game he wants

### The manager is given the appointed time that the customer wants by coming to the hall

### The manager makes sure that the game is available or all devices are reserved at this time

### If available, the customer's name will be recorded in the hall's database and arrival information (time and date(

### 2.2.3 Playar Use Case

#### Use case: Playar

**Diagram:**

Playar

Play on game

**Brief Description**

The player played the game

**Initial Step-By-Step Description**

1 .After the player arrives in the hall, his information is confirmed (his name, the type of game chosen, and the running time of the game)

2 .In the event of a malfunction, the device will stop immediately and the manager will be informed about it

3 .The time during the malfunction of the device will be deducted from the time of its game

4 .If he needs food and drinks on him by pressing a button telling the manager that he wants something and then there are specific people to deliver to him what he wants

#### Use case: Extend Play

Playar

Extend Play

**Diagram:**

**Brief Description**

Extend Time Play If Wanted The Playar

**Initial Step-By-Step Description**

1 .After the customer finishes playing, the machine stops automatically

2 .If the customer wants to extend time with the push of a button, the playing time will be extended for the specified period

3 .Or choose a new game

4 .This should be taken into consideration if the game is not booked by anyone

5 .If it is reserved, the system informs the customer of this

In the event that it is not reserved, the direct player will play in it

#### Use case: game extension

**Diagram:**

Maneger

Extend Play

Play database

**Brief Description**

The manager and what to do during the game extension

Playar

**Initial Step-By-Step Description**

1 .The manager will know what the customer is doing (extending play time, changing the game ..)

2 .The manager will work in the event that the customer requests a new game by searching with him if available or not

3 .If available, he is allowed to play

4 .The manager is updating the customer database

How many games he played and when .

#### Use case: ended time

**Diagram:**

|  |  |
| --- | --- |
| |  | | --- | | DB |   Maneger  Play ended |

**Brief Description**

The player's playing time has ended

**Initial Step-By-Step Description**

1 .The system tells the customer that the time to play is over

2 .The manager updates the database information about this player (the types of games he played and the time for each of them(

3 .The system outputs a card, including the cost of the game and full details of its gameplay, this time and in previous times

4 .If the customer has exceeded 20% submitted to the terminal, the system will inform the manager of a discount with a specified rate

## 2.3 User Characteristics

## It is important that there be a book in the hall for each device that contains a detailed explanation of each game, because not all players are experienced in controlling and playing games, there are players that can be the first time they play the game, so they need to explain without the help of anyone ..

## 2.4 Non-Functional Requirements

Protection

Response time is less than 30 seconds

Connecting the system to a computer network.

Colored printing

Database availability

Connecting the system to a computer network

Providing all nutritional requirements.

# 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 player’s membership. The player 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 Electronic Games hall 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, Search Article  SDD, Section 7.1 |
| **Trigger** | Add player information to the database |
| **Precondition** | The information entered into the database is not repeated |
| **Basic Path** | 1. The player enters his personal information necessary for registration 2. After verification, the system adds a player's field to the database 3. The system shows the player the remaining steps to choose the available timing and games |
| **Alternative Paths** | 1.If he is already registered, he will give him an alert message |
| **Postcondition** | The player gets a membership card in the hall |
| **Exception Paths** | The player may log out |
| **Other** | Lists are created for the registered players in each section separately |

3.2.2 game Reservation

|  |  |
| --- | --- |
| **Use Case Name** | Game Reservation |
| **XRef** | Section 2.2.3, Submit Review  SDD, Section 7.2 |
| **Trigger** | Get the game chosen by the player |
| **Precondition** | Pay the subscription subscription for the game |
| **Basic Path** | 1.The player tells the manager about the game of his choice 2.Provides his information and pays the subscription |
| **Alternative Paths** | The player wants to try the game before registering |
| **Postcondition** | Talking player card information |
| **Exception Paths** | None |
| **Other** | None |

3.2.3 Play on game

|  |  |
| --- | --- |
| **Use Case Name** | Play on game |
| **XRef** | Section 2.2.4, Update Author  SDD, Section 7.3 |
| **Trigger** | Providing basic service to the player |
| **Precondition** | The player must have registered with the game then scored in the game |
| **Basic Path** | After registering for the game, it determines the timing and days for the player to come to the hall |
| **Alternative Paths** | The player does not come at his specified times The player changes his game during a period determined by the manager |
| **Postcondition** | Add information to the player's record by completing the game |
| **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.

Extend game

Num,name

DB

player

maneger

communication

writes

has

Figure 4 - Logical Structure of the Article Manager Data

The data descriptions of each of these data entities is as follows:

**Author Data Entity**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| Name | Text | Name of principle author |  |
| phone | Number | phoneNumber |  |
| id | Number | Player id |  |

# Index

Abstract, 6, 17, 27

add, 9, 11, 19, 20, 21

Add, 8, 9, 19

Article, 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

Article 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

Online Journal, 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