Software Requirements Specification

for

<SMART MOVIE THEATER MANAGEMENT>

Prepared by

Sheikh Sohel Moon

Student ID : 160202

Mesbah Ur Rahman

Student ID : 160218

<Khulna University>

<01 August 2018>

Table of Contents

Table of Contents ii

Revision History ii

1. Introduction 1

1.1 Purpose 1

1.2 Document Conventions 1

1.3 Intended Audience and Reading Suggestions 1

1.4 Product Scope 1

1.5 References 1

2. Overall Description 2

2.1 Product Perspective 2

2.2 Product Functions 2

2.3 User Classes and Characteristics 2

2.4 Operating Environment 2

2.5 Design and Implementation Constraints 3

2.6 User Documentation 3

2.7 Assumptions and Dependencies 3

3. External Interface Requirements 3

3.1 User Interfaces 3

3.2 Hardware Interfaces 3

3.3 Software Interfaces 3

3.4 Communications Interfaces 4

4. System Features 4

4.1 System Feature 1 4

4.2 System Feature 2 (and so on) 4

5. Other Nonfunctional Requirements 5

5.1 Performance Requirements 5

5.2 Safety Requirements 5

5.3 Security Requirements 5

5.4 Software Quality Attributes 5

5.5 Business Rules 6

6. Other Requirements 7

Appendix A: Glossary 5

Appendix B: Analysis Models 5

Appendix C: To Be Determined List 6

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| SMART MOVIE TEHATER MANAGEMENT | August 31, 2018 |  | 1.0 |
| SMART MOVIE THEATER MANAGEMENT | October 23, 2018 | Some Final Changes Were Made For Submission | 1.1 |
| SMART MOVIE THEATER MANAGEMENT | October 28, 2018 | Some Final Changes Were Made For Submission | 1.2 |

# Introduction

## Purpose

**SMART MOVIE THEATER MANAGEMENT** system is a part of a **SMART CITY AUTOMATION**. It’s purpose is to give easy access and operation to all kinds of entertainment related to movies.

## Document Conventions

This report uses the following conventions.

1. DB as Database
2. ER as Entity Relationships
3. UI as User Interface

## Intended Audience and Reading Suggestions

*The project is a prototype of* ***SMART MOVIE THEATER MANAGEMENT SYSTEM*** *and**is designed for all the people living in this city.*

## Product Scope

The **SMART MOVIE THEATER MANAGEMENT SYSTEM is to ease the** Theater management easier. We will have a database server supporting major theaters around the city. Above all, we hope to provide a comfortable user experience to all.

## References

A beautiful example of a web based theater management system can be found on this link for getting a good overview of a very professional theater management system.

<https://www.artsalliancemedia.com/products/theatre-management-system>

# Overall Description

## Product Perspective

The **SMART MOVIE MANAGEMENT SYSTEM** is all about the management of a theater with the help of technology. For example, our system will gather the information for booking tickets, available tickets, users information, time of a show, booking time etc.

## Product Functions

The **SMART MOVIE MANAGEMENT SYSTEM** will have some functions within itself.

They are given below:

1. With this system the movie theater ticket sell will be managed.
2. This system will provide all kind of movie schedules for the entertainment seekers.
3. The theater locations will be given with the help of google map for easy tracking.
4. The schedules will be managed in a more efficient way and so on.

## User Classes and Characteristics

Users of the system should be able to buy tickets easily from our system. There are two types of privileges, users and administrators. Users will have access to user sections and administrators will have access to not only users section but also admin section.

## Operating Environment

*Operating environment for the theater management system is listed below.*

* *database*
* *client/server system*
* *Operating system: Multi-operating systems*
* *database: SQL + database*
* *platform: Multi-platform*

## Design and Implementation Constraints

1. *SQL commands for above queries/applications.*
2. *Implement the database at least using a centralized database management system.*
3. *Working with different schema.*

## User Documentation

A manual and online help will be provided by the software management group to provide good support to the users using the system.

## Assumptions and Dependencies

As we can see the **SMART CITY AUTOMATION** project is a big project to handle. So all the developers working here should be a helpful hand in every sphere. We should encourage each other and develop a mentality on working together. We have to doubly sure that any dependencies don’t make a problem in the development or cause a failure in the development of the system.

# External Interface Requirements

## User Interfaces

* *Front-end software: VB.Net(For A Desktop Based System), PHP(For A Web Based System)*
* *Back-end software: SQL*

## Hardware Interfaces

* *Windows.*
* *VB.Net*
* *SQL*

## Software Interfaces

1. *Operating system*

*We have chosen Windows operating system for it’s good support and user-friendliness.*

1. *Database*

*To save the records we have chosen SQL database.*

1. *VB.Net*

*For desktop UI and functionality implementation.*

## Communications Interfaces

This project supports all types of desktop. We are using VB.net based UI and functionality to get those operations going.

# System Features

*4.1.1* **Description**

1. With this system the movie theater ticket sell will be managed.
2. This system will provide all kind of movie schedules for the entertainment seekers.
3. This system will save us a good amount of time by giving us the freedom to get our tickets in some clicks.
4. The theater locations will be given with the help of Google map for easy tracking.
5. The schedules will be managed in a more efficient way.
6. *The information of the users will be well hidden and encrypted to provide security.*

*4.1.2* **Stimulus/Response Sequences**

*1. The user will be able to create account into the system.*

*2. The user will be able to login into the system. Which will give the user to easily manage his own purpose.*

*3. Booking of tickets will be easier than the analog way.*

*4. Canceling the ticket before the show will be possible within the system.*

*5. Show schedules will be updated for the users.*

*4.1.3* **Functional Requirements**

*1. UI based system.*

*2. SQL based databases for storing all important data.*

*3. A client section and a administrator section.*

*4. Multi-user support.*

# Other Nonfunctional Requirements

## Performance Requirements

The UI, database and all other components of the system should be properly designed to operate with ease. They should be quick and robust. Should use little resources to give good performance. Should be secured too and occupy little space. So that the users will be encouraged to use this system.

## Safety Requirements

If there is extensive damage to a wide portion of the database due to system failure, the recovery method restores a past copy of the database that was backed up to archived storage.

## Security Requirements

Security systems need database storage and many other third party applications. So, vendors should choose their third party partners wisely.

## Software Quality Attributes

*There Are Mainly four quality attributes of a good software:*

***Maintainability***

*Software should be written in such a way so that it can evolve to meet the changing needs of customers. This is a critical attribute because software change is an inevitable requirement of a changing business environment.*

***Dependability and security***

*Software dependability includes a range of characteristics including reliability,security and safety. Dependable software should not cause physical or economic damage in the event of system failure. Malicious users should not be  able to access or damage the system.*

***Efficiency***

*Software should not make wasteful use of system resources such as memory and processor cycles. Efficiency therefore includes responsiveness, processing time, memory utilization, etc.*

***Acceptability***

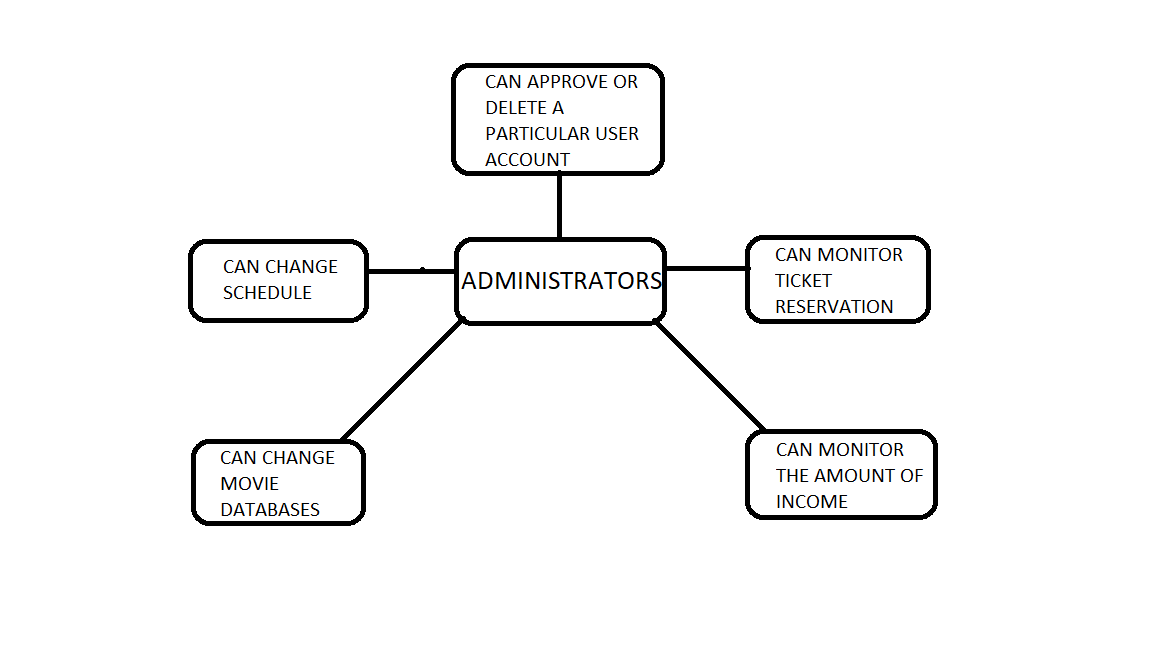
*Software must be acceptable to the type of users for which it is designed. This means that it must be understandable, usable and compatible with other systems that they use.*

## Business Rules

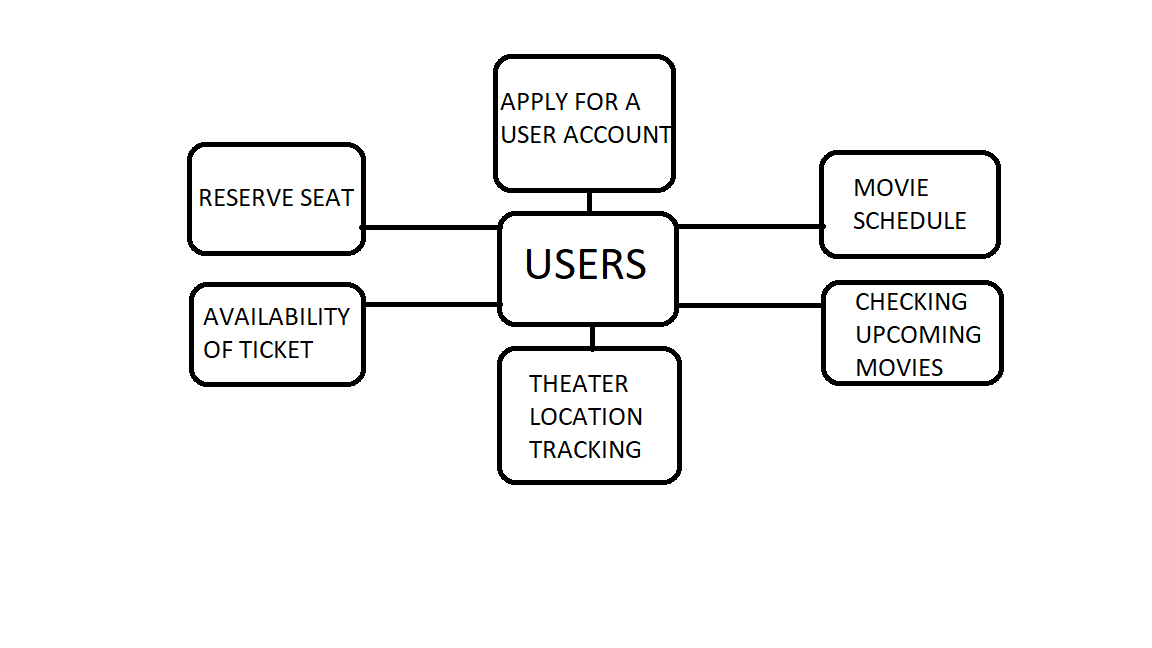
Business ethics and rules should be followed with caution. We should always update the system as needed. Because the world is developing day by day. Specially business is getting into a new dimension. So we should keep a eye on business goals and rules.

# 6.Other Requirements

FOR ADMINSTRATORS



FOR USERS



**6.1Use Cases**

|  |  |
| --- | --- |
| **Use Case Name:** | Registration |
| **Brief Description:** | The web server is waiting on to connect. |
| **Priority** | Essential |
| **Trigger** | User select the link on the SMTM registration form |
| **Precondition** | User is connected to the internet and on the SMTM home page. |
| **Basic Path** | The server presents Registration. |
| **Alternate Path** | N/A |
| **Post condition** | User is on the registration form. |
| **Exception Path** | If there is a connection failure, the Server returns to the wait state. |

6.2 Create the desired theater

|  |  |
| --- | --- |
| **Use Case Name:** | Create the desired theater |
| **Brief Description:** | This operation permits to create the desired theater. |
| **Priority** | Essential |
| **Trigger** | Any theater selected by admin. |
| **Precondition** | Admin must login to create. |
| **Basic Path** | 1. He/she click on add theater. 2. Then add a theater with necessary informations.. 3. He/She click the add theater button. |
| **Alternate Path** | N/A |
| **Post condition** | A record is created in the Theaters Table of the SMTM Database. |
| **Exception Path** | 1. If the connection is terminated before the form is submitted, the fields are cleared and the Server is returned to the wait state. 2. If the connection is terminated after the form is submitted, but before he is returned to the SMTM Admin Panel. the record is created to the SMTM Database. |

6.3 Apply for booking

|  |  |
| --- | --- |
| **Use Case Name:** | Apply for booking |
| **Brief Description:** | This operation permits to apply for booking. |
| **Priority** | Essential |
| **Trigger** | Anyone can book a ticket |
| **Precondition** | He/She must be connected to the Internet and on the SMTM User Login. |
| **Basic Path** | 1. He/she click on book a new ticket. 2. The Server returns a form. 3. He fill in the form and click *Book Ticket* or press enter. 4. The Server checks to see if any required field is empty. 5. If any required field is empty, the Server will send a message and return a new entry form page. 6. If no required field is empty, the Server will create a new record in the Bookings Database. 7. He may select Cancel. 8. If he select Back, the form is cleared and User are returned to the SMTM Admin Home page. |
| **Alternate Path** | N/A |
| **Post condition** | A record is created in the Reservations Table of the SMTM Database. |
| **Exception Path** | 1. If the connection is terminated before the form is submitted, the fields are cleared and the Server is returned to the wait state. 2. If the connection is terminated after the form is submitted, but before he is returned to the SMTM Home Page, |

**6.4 Seeing Movie Details**

|  |  |
| --- | --- |
| **Use Case Name:** | Seeing movie Details |
| **Brief Description:** | This operation permits User to see the movie information in details. |
| **Priority** | Essential |
| **Trigger** | User choose from the Database. |
| **Precondition** | Admin/User must be connected to the Internet, logged in and on the SMTM Home. |
| **Basic Path** | 1. The User clicks on book ticket option.  2. The Server returns movies information when he/she chooses a movie. |
| **Alternate Path** | N/A |
| **Post condition** | The User seeing the details or he/she is returned to the SMTM Home Page |
| **Exception Path** | 1. If the connection is terminated before the form is submitted, the fields are cleared and the Server is returned to the wait state. |

6.5 Create and update a schedule

|  |  |
| --- | --- |
| **Use Case Name:** | Create a new Event |
| **Brief Description:** | This operation permits admin to create a new schedule on the Home page |
| **Priority** | Essential |
| **Trigger** | Admin choose to create a new schedule on the Home page. |
| **Precondition** | Admin must be connected and logged in the SMTM Home. |
| **Basic Path** | 1. Admin click on add/update a new entry. 2. The Server returns a form. 3. Admin fill in the form and click *Add/update* or press enter. 4. The Server checks to see if any required field is empty. 5. If any required field is empty, the Server will send a message and return Admin to the new entry form page. 6. If no required field is empty, the Server will create a new record in the schedules in the movies table, and return Admin to the SMTM Admin Home Page. 7. Admin may select Admin panel. 8. If Admin select admin panel, the form is cleared and Admin is returned to the SMTM Admin Home page. |
| **Alternate Path** | N/A |
| **Post condition** | A record is created/updated in the movies table of the SMTM Database. |
| **Exception Path** | 1. If the connection is terminated before the form is submitted, the fields are cleared and the Server is returned to the wait state. 2. If the connection is terminated after the form is submitted, but before Admin are returned to the SMTM Admin Home Page, the record is created in the Movies Table. of the SMTM Database. |

**6.6 monitor income of theater**

|  |  |
| --- | --- |
| **Use Case Name:** | Monitor income of theater |
| **Brief Description:** | This operation permits Admin to check the income of a particular date |
| **Priority** | Essential |
| **Trigger** | Admin choose to see the income in the Database. |
| **Precondition** | Admin must be logged in and on the SMTM Home. |
| **Basic Path** | 1. The Admin clicks on manage theater.  2. The Server returns a check sales option. |
| **Alternate Path** | N/A |
| **Post condition** | The record is calculated from reservations Table of the Database. |
| **Exception Path** | If the connection is terminated before the click is given,the Server returns to the wait state. |