Table of Contents

1. Introduction 1

1.1 Purpose 1

1.2 Document Conventions 1

1.3 Intended Audience and Reading Suggestions 1

1.4 Product Scope 2

1.5 References 2

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 3

2.5 Design and Implementation Constraints 3

2.6 User Documentation 4

2.7 Assumptions and Dependencies 4

3. External Interface Requirements 5

3.1 User Interfaces 5

3.2 Hardware Interfaces 5

3.3 Software Interfaces 5

3.4 Communications Interfaces 6

4. System Features 6

4.1 System Feature 1 6

4.2 System Feature 2 (and so on) 7

5. Other Nonfunctional Requirements 10

5.1 Performance Requirements 10

5.2 Safety Requirements 10

5.3 Security Requirements 11

5.4 Software Quality Attributes 11

Software Requirements Specification

for

Community Center Status Update

Version 1.0 approved

MD.Saiful Islam and S.M.Shahed

Computer Science and Engineering Discipline, Khulna University

1.8.18

# 1.1Introduction

This section gives a scope description and overview of everything included in this SRS document. Also, the purpose for this document is described and a list of abbreviations and definitions is provided.

## Purpose

The purpose of this document is to give a detailed description of the requirements for the “Community Center Status Update” (CCSU) software. It will illustrate the purpose and complete declaration for the development of system. It will also explain system constraints, interface and interactions with other external applications.

## Document Conventions

We shall follow srs template introduced by IEEE. We will follow use ‘Times New Roman’ front. For a section the front size will be 18 and the front will be also bolded. The front size of the body will be 12.The statement of the great significance will be highlighted.

## Intended Audience and Reading Suggestions

The document is intended for different types of reader such as project manager, users, testers and documentation writers. The document will be included overall description of the system and their functionalities, user classes, product functions, operating environment, design, external interface and requirements, implementation, system features. The readers should read the documents step by step. The document also included the user interaction of that system.

## Product Scope

There are different types of community center in Khulna city. But it should not be properly used and monitoring. Suppose a outsider come to the city. He/she do not know the location of center around him/her. For this regard he/she should use our system and can easily find the location of community center around him/her. The system will show the facilities that provided to the guest and can also see the present status of the community center. The system will be user friendly and can reduce the efforts of the user. Our goal is to design a Community Center Status that will be help to the user.

## References

[1] IEEE Software Engineering Standards Committee, “IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications”, October 20, 1998.

[2]<https://www.tutorialspoint.com/software_testing_dictionary/software_requirement_specification.htm>

## 2.1 Overall Description

This section will give an overview of the whole system. The system will be explained in its context to show how the system interacts with other systems and introduce the basic functionality of it. It will also describe what type of stakeholders that will use the system and what functionality is available for each type. At last, the constraints and assumptions for the system will be presented.

## 2.2 Product Perspective

Community Center Status Update is a very useful and timesaving system for those who like to organize their program of Marriage ceremony, Birthday party, General gathering and another ceremony or party. Another perspective of this system is booking in online to anywhere .User can see the quality of the community center .User can see the size and capacity of this community center.

## 2.3 Product Features

Community Center Status Update system provides users the following functions/features:

* List of community centers under a city corporation.
* Users can request of booking listed community centers.
* User can see the facilities of the community center.
* User can see the present status of the community center.
* Review the community center by user.

## 2.4 User Class and Characteristics

Users of the system should be able to get information of community centers under a city corporation with the given date/time of booking from the database.  The system will support two types of user privileges, Customer, and Administrator. Customers will have access to customer functions, and the Administrator will have access to both customer and booking information functions. The customer should be able to do the following functions:

* Make a new reservation

• Select desire community center  
 • Apply for this community center  
 • Giving the information  
 • Flexible Date/time  
 • Confirmation

* Cancel an existing reservation
* View his itinerary

The Employee should have following management functionalities:

* CUSTOMER FUNCTIONS.

• Get all customers who have applied reservation on a day.  
 • View time schedule.  
 • Get all reservation whose start and end times are on time/delayed.  
 • Calculate total cost for a reservation.

* ADMINISTRATIVE

• Add/Delete a reservation  
• Add a new reservation  
• Update fare for reservation.  
• Update end/start times for reservation.

Each community center has a limited number of available reservation.

## 2.5 Operating Environment

Operating environment for the airline management system is as listed below.

* distributed database
* client/server system
* Operating system: Windows.
* database: sql+ database
* platform: asp.net

## 2.6 Design and Implementation Constraints

1. The global schema, fragmentation schema, and allocation schema.
2. SQL commands for above queries/applications
3. How the response for application 1 and 2 will be generated. Assuming these are global queries. Explain how various fragments will be combined to do so.
4. Implement the database at least using a centralized database management system.

## 2.7 Assumptions and Dependencies

Let us assume that this is a community center status update system and it is used in the following application:

* A request for booking/cancellation of a community center from any source to any destination, giv
* Calculate of reservation cost and calculating appropriate reward points for these reservation.

# External Interface Requirements

**3.1 User Interfaces**

* Front-end software: Vb.net version
* Back-end software: SQL+

**3.2 Hardware Interfaces**

* Windows.
* A browser which supports CGI, HTML & Javascript.

**3.3 Software Interfaces**

Following are the software used for community center status update system.

|  |  |
| --- | --- |
| **Software used** | **Description** |
| Operating system | We have chosen Windows operating system for its best support and user-friendliness. |
| Database | To save the reservation, customer records we have chosen SQL+ database. |
| ASP.Net | To implement the project we have chosen ASP.Net language for its more interactive support. |

**3.4 Communications Interfaces**

This project supports all types of web browsers. We are using simple electronic forms for the reservation forms, booking list etc.

# 4.System Features

### 4.1.1 Description and Priority

### The community center status update system maintains information on reservation, facilities, personal preferences, prices, and bookings. Of course, this project has a high priority because it is very difficult to arrange a program or ceremony in community center without prior reservations.

**4.1.2 Stimulus**

* Search for a community center to arranging various kind of programs
* Displays a detailed list of community center in city corporation “Reservation” or Booking on a particular day.
* Cancel an existing Reservation.

**4.1.3 Functional Requirements**

user

4.3.Use cases description

4.3.1.Registration

|  |  |
| --- | --- |
| **Use Case Name:** | Registration |
| **Brief Description:** | The web server is waiting on to connect. |
| **Priority** | Essential |
| **Trigger** | User select the link on the CCSU registration form |
| **Precondition** | User is connected to the internet and on the CCSU 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. |

4.3.2. Fill out the Survey

|  |  |
| --- | --- |
| **Use Case Name:** | Survey |
| **Brief Description:** | This operation permits user to fill out a survey. |
| **Priority** | Essential |
| **Trigger** | User choose to fill out a survey |
| **Precondition** | User are connected to the internet and on the Home Page. |
| **Basic Path** | 1. The Server presents the user with a form.  2. User fill in the form and click submit or press Enter  3. The Server checks to see if all required fields are not empty.  4. If the required fields are not empty, the  Server creates a new record  in then Survey Table of the User  Database.  5. If any of the required fields are empty,  the Server returns a  message and returns the User to the  Survey form.  6. The Server returns to the Admin Home Page |
| **Alternate Path** | N/A |
| **Post condition** | The survey record is created in the Survey Table of the Database. |
| **Exception Path** | If the connection is terminated before the form is submitted, the fields are all cleared and the Server is returned to the wait state. |

4.3.3 Select the desire community center

|  |  |
| --- | --- |
| **Use Case Name:** | Select the desire community center |
| **Brief Description:** | This operation permits to select the desire community center where user want to do his event. |
| **Priority** | Essential |
| **Trigger** | Any community center selected by user. |
| **Precondition** | He/She must be connected to the Internet and on the CCSU Home page |
| **Basic Path** | 1. He/she click on community center profile. 2. The Server returns the list of community center. 3. He /she select the any community center. 4. He/She click the submit button. 5.Then it returns to the reservation page. |
| **Alternate Path** | N/A |
| **Post condition** | A record is created in the Admin/User Table of the community center 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 CCSU Home Page, the record is created of the community center Database. |

4.3.4 Apply for Reservation

|  |  |
| --- | --- |
| **Use Case Name:** | Apply for Reservation |
| **Brief Description:** | This operation permits to apply for reservation. |
| **Priority** | Essential |
| **Trigger** | Anyone apply for reservation |
| **Precondition** | He/She must be connected to the Internet and on the CCSU Home page |
| **Basic Path** | 1. He/she click on add a new reservation. 2. The Server returns a form. 3. He fill in the form and click *submit* 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 Event Finder Database. 7. He may select Cancel. 8. If he select Cancel, the form is cleared and Admin are returned to the CCSU Admin Home page. |
| **Alternate Path** | N/A |
| **Post condition** | A record is created in the Admin/User Table of the community center 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 CCSU Home Page, the record is created of the community center Database. |

**4.3.5. Seeing Facility Details**

|  |  |
| --- | --- |
| **Use Case Name:** | Seeing facility Details |
| **Brief Description:** | This operation permits User to see the facility in details |
| **Priority** | Essential |
| **Trigger** | User choose in the Database. |
| **Precondition** | Admin/User must be connected to the Internet, logged in and on the CCSU Home page |
| **Basic Path** | 1. The Admin/User clicks on facilities profile option.  2. The Server returns facilities information. |
| **Alternate Path** | N/A |
| **Post condition** | The User/Admin seeing the facilities or he/she is returned to the CCSU 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. |

4.3.5. Create a new Event

|  |  |
| --- | --- |
| **Use Case Name:** | Create a new Event |
| **Brief Description:** | This operation permits admin to create a new event on the Home page |
| **Priority** | Essential |
| **Trigger** | Admin choose to create a new event on the Home page. |
| **Precondition** | Admin must be connected to the Internet , logged in and on the CCSU Home page |
| **Basic Path** | 1. Admin click on add a new entry. 2. The Server returns a form. 3. Admin fill in the form and click *submit* 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 Event Table in the Event Finder Database, and return Admin to the CCSU Admin Home Page. 7. Admin may select Cancel. 8. If Admin select Cancel, the form is cleared and Admin are returned to the CCSU Admin Home page. |
| **Alternate Path** | N/A |
| **Post condition** | A record is created in the Event Table of the community center 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 CCSU Admin Home Page, the record is created in the Event Table of the community center Database. |

4.3.6. Update an Event

|  |  |
| --- | --- |
| **Use Case Name:** | Update an Entry |
| **Brief Description:** | This operation permits Admin to update an existing entry in the Event Finder Database. |
| **Priority** | Essential |
| **Trigger** | Admin/user choose to update an existing entry in the Database. |
| **Precondition** | Admin must be connected to the Internet, logged in and on the CCSU Home page |
| **Basic Path** | 1. The Admin clicks on update an event. 2. The Server returns a form to choose an event by drop down menu.  3. The Server returns a form with the data for that Event in it and a message to update the data they wish and click submit or press Enter 4. The Server replaces the old data with the new data and returns the Admin to the CCSU Admin Home Page. |
| **Alternate Path** | N/A |
| **Post condition** | The record in the Event Table of the Database has been updated and the Admin is returned to the CCSU Admin 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. 2. If the connection is terminated after the form is submitted, but before Admin are returned to the CCSU Admin Home Page, the record is created in the Event Table of the community center Database. |

4.3.7. Search an Event

|  |  |
| --- | --- |
| **Use Case Name:** | Search an Event |
| **Brief Description:** | This operation permits the Admin/User to search for the information. |
| **Priority** | If time permits. |
| **Trigger** | The User/Admin chooses to search. |
| **Precondition** | User/Admin must be connected to the Internet, logged in and on the CCSU Home page |
| **Basic Path** | 1. A search bar will be seen by an Admin or User.  2. They search according their will.  3. Then they click on Search or press Enter.  4. The Server returns the information and back to the Home page. |
| **Alternate Path** | N/A |
| **Post condition** | The User/Admin receives the information or he is returned to the CCSU Home Page. |
| **Exception Path** | 1. If the connection is terminated before the information is returned, the Server is returned to the wait state. 2. If the connection is terminated after the information is returned, the Server is returned to the wait state |

**4.3.8. Delete an Event**

|  |  |
| --- | --- |
| **Use Case Name:** | Delete an Event |
| **Brief Description:** | This operation permits Admin to delete an event. |
| **Priority** | Essential |
| **Trigger** | Admin choose to delete an existing entry in the Database. |
| **Precondition** | Admin must be connected to the Internet, logged in and on the CCSU Home page |
| **Basic Path** | 1. The Admin clicks on delete an event. 2. The Server returns a form to choose an event by drop down menu.  3. The Server returns a form with the data for that Event in it and a message to delete the event they wish and click delete or press Enter. 4. The Server delete the old data and returns the Admin to the CCSU Admin Home Page. |
| **Alternate Path** | N/A |
| **Post condition** | The record in the Event Table of the Database has been deleted and the Admin is returned to the CCSU Admin Home Page. |
| **Exception Path** | If the connection is terminated before the form is submitted, the fields are all cleared and the Server is returned to the wait state. |

**4.3.9. Submit Feedback**

|  |  |
| --- | --- |
| **Use Case Name:** | Submit Feedback |
| **Brief Description:** | This operation permits User to submit feedback about the event |
| **Priority** | Optional |
| **Trigger** | User choose to submit a Feedback about the event |
| **Precondition** | User must be connected to the Internet , logged in and on the CCSU Home page |
| **Basic Path** | 1. User click on Feedback option. 2. The Server returns a box. 3. Admin fill in the box and click *submit* or press enter. 4. Admin may select Cancel. 5. If User select Cancel, the box is cleared and User is returned to the CCSU Home page. |
| **Alternate Path** | N/A |
| **Post condition** | A record is created in the Feedback Table of the community center Database. |
| **Exception Path** | 1. If the connection is terminated before the feedback is submitted, the fields are cleared and the Server is returned to the wait state. 2. If the connection is terminated after the feedback is submitted, but before User are returned to the CCSU Home Page, the record is created in the Feedback Table of the community center Database. |

**4.3.10. View Feedback**

|  |  |
| --- | --- |
| **Use Case Name:** | View Feedback |
| **Brief Description:** | This operation permits Admin to view the users feedback |
| **Priority** | Optional |
| **Trigger** | Admin choose to view the feedback |
| **Precondition** | Admin must be connected to the Internet , logged in and on the CCSU Admin Home page |
| **Basic Path** | Admin click on the view feedback option and see the feedback |
| **Alternate Path** | N/A |
| **Post condition** | The User receives the information or he is returned to the CCSU Home Page |
| **Exception Path** | If there is a connection failure, the Server returns to the wait state. |

**4.3.11. Add an Admin**

|  |  |
| --- | --- |
| **Use Case Name:** | Add an Admin |
| **Brief Description:** | This operation permits to add an admin |
| **Priority** | Essential |
| **Trigger** | Admin choose to add an admin |
| **Precondition** | Admin must be connected to the Internet , logged in and on the CCSU Home page |
| **Basic Path** | 1. Admin clicks on add an admin.  2. All data from the database is returned from database and beside every data there is an option/check box to add as admin.  3. By clicking the check box and finally click on add option or press enter.  4. Admin may select Cancel.  8. If Admin select Cancel, the box is cleared and he is returned to the CCSU Home page. |
| **Alternate Path** | N/A |
| **Post condition** | A record is updated. |
| **Exception Path** | If the connection is terminated before the form is submitted, the fields are all cleared and the Server is returned to the wait state. |

# 5. Other Nonfunctional Requirements

## 5.1 Performance Requirements

The basic objective of normalization is to reduce redundancy which means that information is to be stored only once. Storing information several times leads to wastage of storage space and increase in the total size of the data stored.

If a database is not properly designed it can give rise to modification anomalies. Modification anomalies arise when data is added to, changed or deleted from a database table. Similarly, in traditional databases as well as improperly designed relational databases, data redundancy can be a problem. These can be eliminated by normalizing a database.

Normalization is the process of breaking down a table into smaller tables. So that each table deals with a single theme. There are three different kinds of modifications of anomalies and formulated the first, second and third normal forms (3NF) is considered sufficient for most practical purposes. It should be considered only after a thorough analysis and complete understanding of its implications.

**5.2 Safety Requirements**

If there is extensive damage to a wide portion of the database due to catastrophic failure, such as a disk crash, the recovery method restores a past copy of the database that was backed up to archival storage (typically tape) and reconstructs a more current state by reapplying or redoing the operations of committed transactions from the backed up log, up to the time of failure.

**5.3 Security Requirements**

Security systems need database storage just like many other applications. However, the special requirements of the security market mean that vendors must choose their database partner carefully.

**5.4 Software Quality Attributes**

* **Availability:** The community center should be available on the specified date and specified time as many customers are doing advance reservations.
* **Correctness:** The community center should strict to their rules and regulation.
* **Maintainability:** The administrators and customers should maintain correct schedules of time.
* **Useability:** The facilities should satisfy a maximum number of customers needs.