**Software Requirements**

**Specification**

**for**

**Calendar/Reservation Interface System**

**Version 5.4.4 approved**

**Prepared by Pedro Angeles, George Chan, Ryan Dunning, Hummam Sagga**

**CS3337 Software Engineering**

**10/02/17**

# Table of Contents

Table of Contents...................................................................................................................<pg 2>

Revision History....................................................................................................................<pg 3>

    1.         Introduction................................................................................................................<pg 4>

                 1.1.         Purpose............................................................................................................<pg 4>

                 1.2.         Intended Audience and Reading Suggestions.................................................<pg 4>

                 1.3.         Product Scope.................................................................................................<pg 4>

                 1.4.         Definitions, Acronyms, and Abbreviations ...................................................<pg 4>

                 1.5.         References......................................................................................................<pg 5>

    2.         Overall Description....................................................................................................<pg 6>

                 2.1.         Product Perspective........................................................................................<pg 6>

                 2.2.         Product Functions...........................................................................................<pg 6>

                 2.3.         User Classes and Characteristics....................................................................<pg 6>

                 2.4.         Operating Environment...................................................................................<pg 7>

                 2.5.         Design and Implementation Constraints.........................................................<pg 7>

                 2.6.         User Documentation.......................................................................................<pg 7>

                 2.7.         Assumptions and Dependencies.....................................................................<pg 7>

                 2.8.         Apportioning of Requirements.......................................................................<pg 7>

    3.         External Interface Requirements...............................................................................<pg 8>

                 3.1.         User Interfaces................................................................................................<pg 8>

                 3.2.         Hardware Interfaces........................................................................................<pg 8>

                 3.3.         Software Interfaces.........................................................................................<pg 9>

                 3.4.         Communications Interfaces.............................................................................<pg 9>

    4.         Requirements Specification.......................................................................................<pg 10>

                 4.1.         Functional Requirements................................................................................<pg 10>

                 4.2.         External Interface Requirements....................................................................<pg 12>

                 4.3.         Logical Database Requirements.....................................................................<pg 12>

                 4.4.         Design Constraints.........................................................................................<pg 12>

    5.         Other Nonfunctional Requirements...........................................................................<pg 13>

                 5.1.         Performance Requirements............................................................................<pg 13>

                 5.2.         Safety Requirements......................................................................................<pg 13>

                 5.3.         Security Requirements...................................................................................<pg 13>

                 5.4.         Software Quality Attributes...........................................................................<pg 13>

                 5.5.         Business Rules...............................................................................................<pg 13>

    6.         Other Requirements...................................................................................................<pg 14>

Appendix A: Glossary...........................................................................................................<pg 15>

Appendix B: Analysis Models..............................................................................................<pg 16>

Appendix C: To Be Determined List....................................................................................<pg 17>

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| First Draft | 10/2/17 | afirst draft of document | 1.1.0 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# 1.   Introduction

## 1.1    Purpose

The purpose of this document is to list all the requirements and capabilities of the Calendar and Reservation Interface System or CRIS. CRIS is currently running on a pre- alpha version on cs3 as it is not yet complete.

## 1.2    Intended Audience and Reading Suggestions

This SRS is designed for testers, developers and users. If you are a User only looking to find out how to use CRIS, check Section 2.6 0f this SRS. This section will be updated as CRIS continues to be updated. For Testers, please jump to section 4 of this SRS to view the requirements of this Software. If you are a developer, it would be beneficial to read from the beginning to learn what CRIS can currently do.

## 1.3    Product Scope

This software product is the Calendar Reservation Interface System aka CRIS.

CRIS is designed to fit the needs of a person everyday calendar needs, as well as the needs of peoples who hold a managerial position within an organization, business or institution. This software will allow a user to create a calendar, and with-in this calendar, create and manage a space with tables through the process of table reservation. This process will allow the user to select the details of the space (i.e. create tables), confirm or deny reservations on the space as well-set restrictions for specific users. CRIS will also allow others to follow *public* calendars (i.e. calendars that belong to a user that are public to all other users). This portion of the program is designed to allow other uses to reserve tables that are available in other people's calendars. Users who are making reservations can also edit or cancel these reservations.

The software is designed to be used by many people from many different professions. Since the nature of the software is that of reserving tables, this software can be applied to restaurants, ballrooms, computer labs, concert halls, out-door/in-door swap meets, quite literally anywhere where *tables* can be placed (a table in this sense, does not have to be a physical representation of a table, but can represent a space which people can reserve).

## 1.4    Definitions, Acronyms, and Abbreviations

Table: a space which someone may reserve in an event

Event: a location in the real world in which a calendar with reservations is made for.

CRIS: Calendar/Reservation Interface System

Superadmin: the rank above admin

## 1.5    References

There are currently no references that were used in this document.

# 2.   Overall Description

## 2.1    Product Perspective

This software currently does not relate to any other products; however this software does use Languages such as Java SQL CSS HTML jQuery, jQuery UI …

Software that is similar is any software that is at its core a calendar program that has sharing capabilities (i.e. google calendar, Calendar application on mac, etc.). However, CRIS links directly into a reservation system built into the calendar.

## 2.2    Product Functions

**This software will allow the user to:**

* Create private calendars and events
  + This function is the core of the software. It will allow a user to create a calendar and to schedule an event (i.e. a location in which reservations can be made) on said calendar.
  + Many base components of events will be able to be edited including the name of the event, time that event will be active, location of the event, reservable spots at the event.
* Make private calendars and events public
  + Private calendars and events will be allowed to be made public by the calendars owners. The Calendar owner will be allowed to grant administrator access to other users. Users without administrator access to a calendar will be considered guests.

## 2.3    User Classes and Characteristics

As Stated in Section 2.2, there are three separate classes of users based upon privilege level in respect to the Public Calendars. The three classes are Owner, Administrator and Guest. Each one will have decreasing priorities in the same order.

* + Owners will be allowed to access all functionality of a Public calendar, including setting limitations on other users including but not limited to:
    - General functionality of a private calendar.
    - Giving other users Administrator access.
    - Setting reservation limits on classes of users based on party size as well as times.
  + Administrators will be allowed to:
    - View calendar and edit general settings of the calendar and event i.e. name of the event, access time of the event by other classes of users, size and amount of reservable tables.
  + Guests will be allowed to:
    - View the calendar as well as make reservations dependent on available space.

## 2.4    Operating Environment

This system will live on a remote server in which others may access it through a web browser. As of now, a version of it lives on cs3. Software that CRIS depends on exists in the JavaScript libraries belonging to jQuery and jQuery-UI, as well as the styling elements built into Bootstrap.

## 2.5    Design and Implementation Constraints

This system as of now does not have any constraints dealing with memory or hardware. This system does however need access to the internet and a browser that can interpret HTML CSS and JavaScript.

The software is also dependent on jQuery and the FullCalender JavaScript libraries.

## 2.6    User Documentation

This system does not have any user documentation that will be deployed with the software as of now

## 2.7    Assumptions and Dependencies

This system shall run on any machine that has Java installed on it.

## 2.8    Apportioning of Requirements

The GUI that allows the user to organize tables may be delayed for this system.

# 3.   External Interface Requirements

## 3.1    User Interfaces

This system will interact with the user mainly through a web browser. The first page that any user will see is the login page. The login page will consist of buttons top that will allow the user to register if the user has not done so and two text fields for the user’s username and password with a button on the bottom of the text fields to log in. Once the user clicks the register button at the top of the login page, they will be redirected to the registration page where the user will be asked to input their first name, last name, password, password confirmation, and optionally an email. There will be buttons on the top of the registration page to go back to the login page and on the bottom of the page to submit the information to the remote server for processing. Once the user clicks on either button, they will be redirected back to the login page, where they can log in with their information. Once they click log in, they will be redirected to their respective homepage, where the user will see buttons on the top to go back to their homepage and to log out. On the homepage, the user will see two tabs; the first tab will be the user’s private calendars, whereas the second tab will be the user’s public calendars. If the user clicks on their respective tabs, they will see either options to modify their private calendars or private calendars. For both tabs, there will be a sidebar that has options to modify calendars.

For the public tab, the options sidebar will contain a button to add a calendar, a drop-down menu to edit existing calendars, and a drop-down menu to delete existing calendars. For the private tab, the options sidebar will contain a button to follow a calendar and a drop-down menu to unfollow any calendars that the user is currently following. If the user chooses to add a calendar to their private tab, they will be redirected to a page to create a new calendar. If the user chooses to edit or delete a calendar in their private tab, their respective drop-down menus will list all their calendars that they can modify. If the user chooses to follow a calendar in their public tab, they will be redirected to a page that has a list of all available calendars that they can follow. If the user chooses to unfollow a calendar, they will see a list of all their currently following calendars.

Inside both tabs, a list of all the calendars the user is keeping track of will be presented to them in text format based on the names of the calendars. The user can click on any name to be redirected to the respective calendar page, where they will see all events that are happening on that calendar. Once the user clicks on an event, they will be redirected to an event registration page, where they can create an event if they are in their private tab or can register for an event if they are in the public tab. The page will ask them for the name, duration, location, and table specification if they are creating an event or it will ask the user to register for a specific duration and specify how many people are in the user’s party if they are registering for an event. There will be a button on the bottom that says save/register on the bottom of the page, and if the user clicks that, it will redirect them back to their homepage where they will see their updated list of calendars.

## 3.2    Hardware Interfaces

This system shall run on any device that supports Internet functionalities, such as an Ethernet cable or a network interface controller that may be found on any laptop or desktop.

## 3.3    Software Interfaces

This system shall run on a website with a server. This system shall require the user of the FullCalendar JavaScript libraries. This system shall require the use of jQuery. This system shall require the use of Bootstrap.

## 3.4    Communications Interfaces

This system shall interact with a web browser that will connect to a remote server. This system is only accessible through the Internet. This system should not have any communication security or encryption issues, data transfer rates, and synchronization mechanisms.

# 4.   Requirements Specification

## 4.1    Functional Requirements

1.0) This system shall have a login page.

1.1) This system shall allow the user to create an account via a tab on the login page.

1.2) This system shall allow the user to log in after entering their information.

1.3) This system shall redirect the user to their homepage after logging in.

2.0) This system shall have a registration page.

2.1) This system shall allow the user to create a username on the registration page.

2.2) This system shall allow the user to create a password on the registration page.

2.3) This system shall ask the user for their name on the registration page.

2.4) This system shall ask the user for their email address on the registration page.

2.5) This system shall allow the user to submit their information via a button on the registration page.

2.6) This system may include email verification via the user’s email on the registration page.

2.7) This system shall redirect the user back to the login page once they conclude registration on the registration page.

3.0) This system shall have a homepage for every user.

3.1) This system shall have a button at the top of the homepage to log out.

3.2) This system should have a button at the top of the page to go back to the homepage.

4.0) This system shall have a tab for the user’s private calendars on the homepage.

4.1) This system shall include an options sidebar on the left side of the private calendars tab.

4.1.1) This system shall have an option to add a calendar from a public calendar inside the options sidebar.

4.1.2) This system shall have an option to edit an existing calendar via a drop-down menu inside the options sidebar.

4.1.3) This system shall have an option to delete an existing calendar via a drop-down menu inside the options sidebar.

4.2) This system may allow the user to customize the amount of days to display on any calendar on the private calendars tab.

4.3) This system shall redirect the user to another page once they click the option to add a calendar.

4.3.1) This system shall ask the user the name of the event.

4.3.2) This system shall ask the user the location of the event.

4.3.3) This system shall ask the user when their event will take place.

4.3.4) This system shall allow the user to write a description of their event.

4.3.5) This system shall ask the user the expected number of people attending said event.

4.3.6) This system shall allow the user to designate tables.

4.4) This system shall display the names of the user’s calendars in a list format inside the private calendars tab.

5.0) This system shall have a tab for the user’s public calendars on the homepage.

5.1) This system shall include an options sidebar on the left side of the public calendars tab.

5.1.1) This system shall have an option to follow an existing calendar inside the options sidebar.

5.1.2) This system shall have an option to unfollow an existing calendar via a drop-down menu inside the options sidebar.

5.1.3) This system may have an option to edit a public calendar via a drop-down menu inside the options sidebar.

5.2) This system shall redirect the user to another page to follow a calendar once they click the follow option inside the sidebar.

5.2.1) This system shall include a search bar to search any existing calendars inside the page to follow a calendar.

5.3) This system shall allow the user to register for an event once they click on a calendar from the list of calendars that the user has followed.

5.3.1) This system may include a button to email a superadmin to reserve a spot.

5.3.2) This system shall redirect the user to the calendar view for the calendar chosen on the public calendars tab.

5.3.3) This system shall redirect the user to an event registration confirmation page once they click on an event.

5.3.3.1) The system shall list the name of the event on the event registration confirmation page.

5.3.3.2) The system shall list the location of the event on the event registration confirmation page.

5.3.3.3) The system shall list the duration of the event on the event registration confirmation page.

5.3.3.4) The system should ask the user to choose a specific time during the selected event to register for on the event registration confirmation page.

5.3.3.5) The system shall ask the user the number of people in the user’s party that will attend the event on the event registration confirmation page.

5.3.3.6) This system shall ask the user which table they want to reserve on the event registration confirmation page.

5.3.4) This system may notify the event organizer of all activity that occurs on their event via email.

5.3.4.1) This system may give the event organizer the option to disable notifications of all activity that occurs on their event via email.

5.3.5) This system should implement a waiting list for any event that exceeds capacity.

5.3.5.1) This system should notify anyone via email on the waiting list if their position in the list updates.

5.3.6) This system should check if the user that is registering for an event if there is a time conflict on their schedules from other calendars.

5.3.6.1) This system should warn the user of any potential time conflicts in their schedules from the same or other calendars.

5.4) This system shall display a different color on each calendar cell that corresponds to a case.

5.4.1) This system shall display one color on a calendar cell that corresponds to the case where an event is not happening on that corresponding day.

5.4.2) This system shall display one color on a calendar cell that corresponds to the case where an event is happening on that corresponding day.

5.4.3) This system shall display one color on a calendar cell that corresponds to the case where the user is already registered for an on that corresponding day.

5.4.4) This system shall display one color on a calendar cell that corresponds to the case where an event is not available on that corresponding day.

## 4.2    External Interface Requirements

* + 1. This system shall display an error message if the user enters an invalid username.
    2. This system shall check to see if the user’s username exists in the database.
    3. This system shall display an error message if the user enters an invalid password.
    4. This system shall check to see of the user’s password exists in the database.
    5. This system shall redirect the user to the registration page if the user decides to create an account.
    6. This system shall redirect the user to their homepage if the user logs in with their account.
    7. This system shall redirect the user to the event creation page if the user clicks on the option to add a calendar to their private calendars.
    8. This system shall redirect the user to the event registration page if the user clicks on an appropriate calendar cell in their list of calendars that they are currently following in the public calendars tab.

5.0.2) This system may handle any conflicts with an error message then returns to the user’s previous page that they were on after they close the prompt.

## 4.3    Logical Database Requirements

* + - 1. This system shall send user information, such as the user’s username and password, to a remote server for storage.
      2. This system shall send a ticket from the web browser to the remote server once the user attempts to log in with their user information.
      3. This system should use SQL to organize all users registered to the system.
      4. This system should use SQL to organize all events created by any user.

## 4.4    Design Constraints

6.0) This software shall require the use of the Internet and a browser that can interpret HTML, JavaScript, and CSS.

6.1) This system should need access to jQuery and the FullCalendar JavaScript libraries.

**5.   Other Nonfunctional Requirements**

## 5.1 Performance Requirements

99.9% of all processes through the server should take less than one (1) second.

## 5.2    Safety Requirements

This system shall not warrant any need for safety requirements.

## 5.3    Security Requirements

There shall be no security requirements in this system at this current version.

## 5.4    Software Quality Attributes

There shall be no software quality attributes for this system at this current version.

## 5.5    Business Rules

There shall be no business rules in this system at this current version.

**6.   Other Requirements**

There shall be no other requirements for this system as of now.

# Appendix A: Glossary

Table: a space which someone may reserve in an event

Event: a location in the real world in which a calendar with reservations is made for.

CRIS: Calendar/Reservation Interface System

Superadmin: the rank above admin

# Appendix B: Analysis Models

This system does not have any analysis models at the moment.

# Appendix C: To Be Determined List

This system may include a GUI for organizing the tables in the future.