**ITEC 315**

**Homework – Manage Artis Lab Reservation System Users**

**(revision 7: 4/16/19 1:13 pm)**

The Artis Lab Reservation System ALRS allows select faculty, staff, and students to reserve any of the six resources in the Lab:

* Carilion Suite
* Conference Room
* Break-out Room
* Stadium Pod
* East D-table
* West D-table

For each of these resources a user can have any of the following privileges:

* View (the user can view reservations but cannot make, delete, or modify reservations)
* Reserve (the user can view reservations and can make, delete, or modify their own reservations)
* Admin (the user can view, make, delete, or modify any reservations)

Users are also designated as either ***Student*** or ***FacStaff***. Most students will have ***View*** privs on the Carilion Suite and Conference Room but have ***Reserve*** privs on the other four resources. Most professors will have Reserve privs on all six resources. ***View*** is the minimum priv any user will have on any resource. The ability to customize the privs is required.

In addition, the system includes a list of SysAdmins who are allowed to add, delete, or modify users and privileges. Currently, all user and priv maintenance is done through SQL statements, there is no administrative interface to allow a system administrator to add, delete, or modify users, privs, and other SysAdmins. You are to design that interface using a Windows Form project. The goals of the system are to allow a system administrator to:

1. efficiently add new users and establish common and custom privs
2. efficiently manage existing users, their privs, type (Student or FacStaff), and status (Active or Sunset), including the ability to see how and a how much someone is using the lab
3. Quickly sunset old students and facstaff by switching status to Sunset for students and facstaff that have passed the estimated sunset date in the system.
4. Quickly sunset faculty and staff that are retiring, etc based on the estimated sunset date in the system.
5. Quickly review a list of users that have not made a reservation in the last X days in order to help decide if their account should be sunset.
6. Determine who the heavy users of the system are and what the relative load on lab resources is to help manage the lab.
7. List, Add, and Delete Sys Admins.

For purposes of this assignment you may consider the six resources to be fixed. That is to say, that your system may assume that the current six resources will not change in name or number. So when you are setting privs for the six resources for someone you have a choice:

* You can use a DataGridView that handles any number of resources in the privilege set, or
* You can establish controls for the six assumed privs and interate through the DataTable representing the priv set when reading it or writing it. We will talk about this in class.

Your solution must include a primary application window, a major child window to handle a user and user tasks, and a modal login dialog window that opens on top of your main application window when the application starts. Any other windows/dialogs that you have are up to you to design.

Like the prior homework, you will use a web service for data services. ***See hwk 3 to remind yourself of how to set this up in your project***. The web service will be located at:

<http://itec-web-dev04/EWS/ArtisLabInfoService.asmx>

First, add this service reference to your Project by using the PROJECT->ADD SERVICE REFERENCE choice on the menu strip. When adding it I suggest using the namespace **ARTISLABservice**.

Now, Add a reference to the web service to the program.cs file so that it is available from any form. To do this, open your program.cs file (look in the solution explorer window to find it) and CAREFULLY add the two highlighted lines of code below, the one right above the thread directive and the other just before the Application.Run method call:

public static ARTISLABservice.ArtisLabInfoServiceSoapClient ARTISLABclient;

[STAThread]

static void Main()

{

Application.EnableVisualStyles();

Application.SetCompatibleTextRenderingDefault(false);

ARTISLABclient = new ARTISLABservice.ArtisLabInfoServiceSoapClient("ArtisLabInfoServiceSoap");

Application.Run(new Form1()); //or whatever your mainform is named

}

The web service (ARTISLABservice) will expose these classes:

User

Resource

The web service will also expose these methods:

**SECURITY METHODS**

NOTE: You will use your RU username but a password I provide by email to access this system. Do not delete the email so that you can refer back to the password if you forget it.

* sysLogIn(string username, string password)
  + Used to allow system administrators to login to the service. Creates a new session, returns a string which normally represents a security token.
  + If the returned string is "ERROR:DENIED" or "ERROR:DBMS FAILURE" or any string that starts with "ERROR" then no session has been created.
* sysLogOut(string securityToken)
  + Ends the session with the supplied token
  + Returns a string which is either "SUCCESS " or an error message starting with ERROR

For now you can add and delete new sysadmins but not log on as them.

**READ METHODS**

* getPrivList()
  + Returns a DataTable with these columns
    - Priv
    - Description
* getResourceUsage(string securityToken)
  + Returns a DataTable with these columns
    - rName
    - UsesInLast90Days
    - UsesInLastYear
    - TotalUses
    - Description
* getUserList(string usertype, string status, string securityToken)
  + usertype must be All, Student, or FacStaff
  + status must be All, Active, or Sunset
  + Returns a DataTable with these columns
    - UserName
    - FullName
* getUserTypeList()
  + Returns a DataTable with these columns
    - UserType
* getUserStatusList()
  + Returns a DataTable with these columns
    - UserStatus
* searchUsers(string usertype, string status, string stringtofind,

int DaysInactiveCutOff, int UseIn90DaysCutOff, string SecurityToken)

* + usertype must be All, Student, or FacStaff
  + status must be All, Active, or Sunset
  + stringtofind will search username, fullname, and notes. Set to "" if not using
  + DaysInactiveCutOff will find users who have not had a reservation for that many dats

Set this to 0 if not searching on this.

* + UsesIn90DaysCutOff will find users who have more than that number of reservation in the

Last 90 days. Set this to 0 if not searching on this.

* + Returns a DataTable with these columns which it also searched
    - UserName
    - FullName
    - UserType
    - Status
    - DaysInactive (number of days with no reservation, 0 indicates future reservations)
    - UsesInLast90Days (reservations made in last 90 days)
    - UsesInLastYear (reservations made in last year plus future reservations)
    - AllUses (count of all reservations made by this user)
    - CreateDate
    - EstSunsetDate
    - SunsetDate
    - Notes
* getSysAdminList(string secruityToken)
  + Returns a DataTable with these columns
    - UserName
    - FullName
* getResourceList()
  + Returns a DataTable with one column, rName, representing a list of resources
* getResource(string rname, string secruityToken)
  + Returns an instance of a resource object
  + If no such resource exists the fields will be Null except for rname and status. Status will be ERROR.
  + Fields of the instance include
    - rName
    - Description
    - UsesInLast90Days
    - UsesInLastYear
    - Alluses
    - errmessage (SUCCESS or ERROR: followed by error details after constructor attempt)
* getUser(string username, string secruityToken)
  + Returns an instance of a user object
  + If no such user exists the fields will be Null except for username and status. Status will be ERROR.
  + Fields of the instance include
    - username
    - firstname
    - lastname
    - createdate
    - usertype
    - estsunsetdate (use grad date for students, leave blank for most facstaff)
    - sunsetdate
    - status (Active or Sunset)
    - notes
    - errmessage (SUCCESS or ERROR: followed by error details after constructor attempt)
    - A DataTable named PrivUseSet with these columns
      * rName
      * priv
      * UsesInLast90Days
      * UsesInLastYear
      * Alluses
    - A DataTable named PastReservations with these columns
      * ReservationID
      * rName
      * DateReserved
      * BeginTime
      * EndTime
      * Description
    - A DataTable named FutureReservations with the same columns as PastReservations.

**DATA MANIPULATION METHODS**

* addNewUser(string username, string usertype, string token)
  + Creates a new user in the database with the specified username and usertype and RETURNS that user object.
  + UserType must be Student or FacStaff
  + Once the new user is created you can use your same maintenance screen to edit the user.
* addSysAdmin(string username, string firstname, string lastname, string secruityToken)
  + A string will be returned. Either SUCCESS or ERROR: some message.
* deleteSysAdmin(string username, string secruityToken)
  + Deletes the designated system administrator
  + Will refuse to delete the logged in administrator
  + A string will be returned. Either SUCCESS or ERROR: some message.
* saveUser(usertosave User, string secruityToken)
  + Takes a user object and saves it.
  + A string will be returned. Either SUCCESS or ERROR: some message.
  + Once a user is created the username cannot be changed
* deleteUser(string username, string secruityToken)
  + Deletes the designated and all their reservations.
  + A string will be returned. Either SUCCESS or ERROR: some message.
* sunsetRetiredUsers(string usertype, int graceDays, string secruityToken)
  + usertype is All, Student, or FacStaff
  + Calls a method which will sunset any user that is now past their estimated sunset date by at least the number of days indicated by the graceDays input parameter.
  + A string will be returned. Either SUCCESS or ERROR: some message.

WHEN WORKING WITH THE WEBSERVICE PLEASE ONLY UPDATE OR DELETE YOURSELF OR USERS YOU HAVE ADDED. DO NOT DELETE OR CHANGE OTHER CLASS MEMBERS OR THEIR TEST DATA. ALL ACTIVITY IS BEING LOGGED. BE PROFESSIONAL. DO NOT SHARE YOUR PASSWORD WITH CLASSMATES

The web service provides you access to these database fields:

|  |  |  |  |
| --- | --- | --- | --- |
| Fieldname | C# Type | Oracle Type (Size) | Notes |
| username | string | Varchar2(25) |  |
| firstname | string | Varchar2(25) |  |
| lastname | string | Varchar2(25) |  |
| createDate | DateTime | Date |  |
| estSunsetDate | DateTime | Date |  |
| SunsetDate | DateTime | Date |  |
| usertype | string | Varchar2(10) | Student or FacStaff |
| status | string | Varchar2(10) | Active or Sunset |
| comments | string | Varchar2(60) |  |
| revnum | int | Number(8) | For mutation detection |
| rname | string | Varchar2(25) |  |
| description | string | Varchar2(100) |  |
| ReservationID | int | Number(8) |  |
| dateReserved | DateTime | Date |  |
| begTime | DateTime | Date |  |
| endTime | DateTime | Date |  |
| priv | string | Varchar2(10) |  |
| UsesInLast90Days | int | Number(8) |  |
| UsesInLastYear | int | Number(8) |  |
| TotalUses | int | Number(8) |  |