# CCWNC411\_Security\_LOG

# GITHUB

## SSH

### git@github.com:CCWNC-DEVELOPER/aspnetdb\_WebInterface.git

## HTTPS

### https://github.com/CCWNC-DEVELOPER/aspnetdb\_WebInterface.git

## Create New Repo:

### touch README.md

### git init

### git add README.md

### git commit -m "first commit"

### git remote add origin https://github.com/CCWNC-DEVELOPER/aspnetdb\_WebInterface.git

### git push -u origin master

## Push an existing repository from the command line

### git remote add origin https://github.com/CCWNC-DEVELOPER/aspnetdb\_WebInterface.git

### git push -u origin master

# STOPPED:

## Problems to work on:

### http://172.15.189.14:33666/LockedDown/Roles\_for\_Application.aspx

#### When I delete a role with existing users, it does not give me the feedback that there are users, all I see is that the role is not deleted

### Add a link to the CCWNC411 into the Master of The Security WA

## TO-DO

### Error in the PARM for Delete

#### Need a drop down for Users that are not in the application-role combination, that populates the text box

##### This will require a new SP to return a list of Usernames that are not in that has parms:

###### ApplicationName

###### RoleName

##### From View:

###### prj\_0031\_vw\_001\_ApplicationName\_RoleName\_UserName\_and\_IDs

#### http://localhost:62209/LockedDown/User\_CRUD.aspx

##### New User:

###### IsApproved =false

Should be true

#### http://localhost:62209/LockedDown/Roles\_for\_Application.aspx

##### I let me add a blank Role

#### Need a <h1> title for each page

#### http://localhost:62209/LockedDown/Roles\_for\_User\_Applications.aspx

##### need the Application and Role name in the GridView

##### need a delet button to delete it if so desired

### Users

#### Then create the ‘Update’ Sub for password

#### Then create the ‘Delete’ sub

#### Enable Delete

### Roles:

#### Create New Role

##### **aspnet\_Roles\_CreateRole**

###### Input

@ApplicationName nvarchar(256),

@RoleName nvarchar(256)

###### Output

Return

#### Update Role name

#### Delete Role

##### **aspnet\_Roles\_DeleteRole**

###### Input

@ApplicationName nvarchar(256),

@RoleName nvarchar(256),

@DeleteOnlyIfRoleIsEmpty bit

#### SELECT All Roles for a given Application

##### **aspnet\_Roles\_GetAllRoles**

###### Input

@ApplicationName nvarchar(256))

###### Output

RoleName

#### MISC

##### aspnet\_Roles\_RoleExists

###### @ApplicationName nvarchar(256),

###### @RoleName nvarchar(256)

##### aspnet\_Setup\_RemoveAllRoleMembers {**DO NOT USE THIS**}

###### @name sysname

### User – Roles

#### Add User to Role

##### **aspnet\_UsersInRoles\_AddUsersToRoles**

###### Input

@ApplicationName nvarchar(256),

@UserNames nvarchar(4000),

@RoleNames nvarchar(4000),

@CurrentTimeUtc datetime

#### SELECT Users in Role

##### aspnet\_UsersInRoles\_FindUsersInRole {NOT SURE about the need for this one}

###### @ApplicationName nvarchar(256),

###### @RoleName nvarchar(256),

###### @UserNameToMatch nvarchar(256)

##### **aspnet\_UsersInRoles\_GetUsersInRoles**

###### Input

@ApplicationName nvarchar(256),

@RoleName nvarchar(256)

###### Output

UserName

#### SELECT Roles for User

##### aspnet\_UsersInRoles\_GetRolesForUser

###### @ApplicationName nvarchar(256),

###### @UserName nvarchar(256)

#### Check If User is In Role

##### aspnet\_UsersInRoles\_IsUserInRole

###### @ApplicationName nvarchar(256),

###### @UserName nvarchar(256),

###### @RoleName nvarchar(256)

#### Delete User from role

##### **aspnet\_UsersInRoles\_RemoveUsersFromRoles**

###### Input

@ApplicationName nvarchar(256),

@UserNames nvarchar(4000),

@RoleNames nvarchar(4000)

# Index.aspx

## Need to document using the business object as a datasource

### And the configure DataSource wizard

## Applications

### To Do:

#### PasswordSalt: how to generate this?

#### Have it check to see if there is already an application with that name before proceeding

##### If there is a unique key violation, then show a message

#### Have it automatically put the lower case version into the second field for the application name

### Get Application List

### Add Application

### Delete application

## Users

### Try enabling the older working version of the redirect to see what it does

### New Strategy: 12/29/2013

#### Back-End

##### I did ‘ONE’ which was the simple version

##### Now do ‘TWO’ which is the strongly typed version and get it working as CodeBehind in project: MS\_CodeBehind, from: ObjectDataSource\_Example

##### Once that is working, bring it into CCWNC411\_Security, and hollow out the subs, but leave the signatures

###### Also redo the User\_Record using that template

###### Then bring in the front end without changing the names of the subs

##### Then

###### One-by-one populate the subs, and change any sub names as needed

##### NEXT

###### User\_CRUD

This will get the Application From the URL when redirected by the Drop-down

Then present a list of all the Users of that Application

Use a select parameter: ApplicationName

###### Use a parameterized Select statement for the ‘SelectMethod’ of the ObjectDataSource for the Grid

In this case the parameters will correspond to those of the SP: [aspnet\_Membership\_GetAllUsers]

@ApplicationName nvarchar(256),

@PageIndex int,

@PageSize int

So that when the drop-down chooses the application, then the grid populates with the members corresponding to that Application

### TO-DO: OLD

#### If “----SELECT===” then show message

#### Error when submitting:

##### Cannot insert the value NULL into column 'LastActivityDate', table 'aspnetdb.dbo.aspnet\_Users'; column does not allow nulls. INSERT fails.

##### Cannot insert the value NULL into column 'PasswordFormat', table 'aspnetdb.dbo.aspnet\_Membership'; column does not allow nulls. INSERT fails.

##### The statement has been terminated.

##### The statement has been terminated.

#### DetailsView

##### Has to have all the fields to insert/create user:

###### aspnet\_Membership\_CreateUser

##### Then will be able to:

###### Delete

aspnet\_Users\_DeleteUser

###### Update Password

aspnet\_Membership\_ResetPassword

#### Then create the ‘Create’ Sub

##### Must be able to use the ApplicationName to tie user to application

##### May want to do some overloads

###### Check out the overloads for the Membership.CreateUser sub

http://msdn.microsoft.com/en-us/library/system.web.security.membership.createuser%28v=vs.110%29.aspx

## User SP

### Select User

#### SP

##### prj\_0031\_WA003\_SP\_001\_SELECT\_User\_Given\_ApplicationName\_and\_UserName

###### aspnet\_Applications.ApplicationName,

###### aspnet\_Users.UserName,

###### aspnet\_Membership.Password,

###### aspnet\_Membership.PasswordSalt,

###### aspnet\_Membership.Email,

###### aspnet\_Membership.PasswordQuestion,

###### aspnet\_Membership.PasswordAnswer,

###### aspnet\_Membership.IsApproved,

###### aspnet\_Membership.CreateDate,

###### aspnet\_Membership.PasswordFormat,

###### aspnet\_Users.UserId,

###### --The ones below here are not in the Create SP

###### aspnet\_Membership.IsLockedOut,

###### aspnet\_Membership.LastLoginDate,

###### aspnet\_Membership.FailedPasswordAttemptCount, {**ADD** }

###### aspnet\_Users.LastActivityDate,

###### aspnet\_Applications.ApplicationId {**add**}

##### aspnet\_Membership\_GetPasswordWithFormat

##### aspnet\_Membership\_GetUserByUserId

###### Input:

@UserId uniqueidentifier,

@CurrentTimeUtc datetime,

@UpdateLastActivity bit = 0

###### Output:

From: aspnet\_Membership

Email

PasswordQuestion

Comment

IsApproved

CreateDate

LastLoginDate

IsLockedOut,

LastLockoutDate

LastPasswordChangedDate

From: aspnet\_Users

LastActivityDate

**UserName**

### Create User

#### aspnet\_Membership\_CreateUser\_Record

#### aspnet\_Membership\_**CreateUser**\_DL

##### SP

###### aspnet\_Membership\_CreateUser

@ApplicationName

@UserName

@Password

@PasswordSalt

@Email {add}

@PasswordQuestion {add}

@PasswordAnswer {add}

@IsApproved

@CurrentTimeUtc {NOT IN SELECT}

@**CreateDate**

@UniqueEmail {NOT IN SELECT}

@PasswordFormat

@UserId

### Delete User

#### DL Class

##### SP

###### aspnet\_Users\_DeleteUser

@ApplicationName

@UserName

@TablesToDeleteFrom

@NumTablesDeletedFrom

### Modify User

#### DL

##### Common fields for User Update

###### @UserName

###### @NewPassword

###### @ApplicationName

###### @IsApproved

###### @Comment

###### @Email

###### @UniqueEmail

###### @CurrentTimeUtc

###### @IsPasswordCorrect

###### @LastActivityDate

###### @LastLoginDate

###### @MaxInvalidPasswordAttempts

###### @NewPasswordAnswer

###### @NewPasswordQuestion

###### @PasswordAnswer

###### @PasswordAttemptWindow

###### @PasswordFormat

###### @PasswordSalt

###### 

###### @UpdateLastLoginActivityDate

##### SP for Methods

###### aspnet\_Membership\_UpdateUser

Updates the user's last activity date in the aspnet\_Users table and e-mail address, comment, is-approved status, and last login date in the aspnet\_Membership table.

@**ApplicationName**

@**UserName**

@Email

@Comment

@IsApproved

@LastLoginDate

@LastActivityDate

@UniqueEmail

@CurrentTimeUtc

###### aspnet\_Membership\_UpdateUserInfo

Updates account locking data for the specified user in the aspnet\_Users and aspnet\_Membership tables. Used in conjunction with provider methods that track bad password and bad password-answer attempts.

@**ApplicationName**

@**UserName**

@IsPasswordCorrect

@UpdateLastLoginActivityDate

@MaxInvalidPasswordAttempts

@PasswordAttemptWindow

@CurrentTimeUtc

@LastLoginDate

@LastActivityDate

###### **aspnet\_Membership\_ResetPassword**

Resets the specified user's password based on a password answer.

@**ApplicationName**

@**UserName**

@**NewPassword**

@MaxInvalidPasswordAttempts

@PasswordAttemptWindow

@PasswordSalt

@CurrentTimeUtc

@PasswordFormat

@PasswordAnswer

###### aspnet\_Membership\_ChangePasswordQuestionAndAnswer

Changes the specified user's password question and answer.

@**ApplicationName**

@**UserName**

@NewPasswordQuestion

@NewPasswordAnswer

#### GetUsers

##### UserName

##### Email

##### PasswordQuestion

##### Comment

##### IsApproved,

##### CreateDate,

##### LastLoginDate,

##### LastActivityDate,

##### **LastPasswordChangedDate**,

##### UserId

##### **IsLockedOut**

##### **LastLockoutDate**

## Roles SP

### Create Role

#### aspnet\_Roles\_CreateRole

### Delete Role

#### aspnet\_Roles\_DeleteRole

### Modify Role

## User-Role

### Add User to Role

### Delete User from Role