CIS 3342 Term Project –  
Responsibilities

Subject: CIS 3342 Spring 2021 Section 2 Term Project – Responsibilities

Group Member 1: James Skomsky (JimS) email: tun49199@temple.edu

Group Member 2: Frank Petrillo email: (FrankP) frankpetrillo5@gmail.com

Topic Proposal 1: New app name is Connect2Cal ~~Connect2MD~~ WebApp Note: per team discussion one small change will be made to the original proposal. Namely the calendar data object originally just Health Care Appointments. The object will include a new type of field. This will provide scheduling for additional types of events for instance a calendar to help find/schedule a Vaccine appointment or find/schedule a flight on an AirLine. This will provide some simple business logic and in general increase the marketability for the WebApp.

Please see the Summary List at the bottom for a detailed list of specific items.

1. Login.ASPX: (JimS)
   1. Will Utilize SOAP WebMethods and StoredProcedures to read login credentials, serialize/deserialize Byte[] to the database to implement “Log In Remember Me” functionality. The existence of a cookie tells the code-behind to Redirect to either the ~~AdminClient~~ or UserClient page depending on account type.
2. ForgotPassword (JimS)
   1. Utilize SOAP WebMethods and StoredProcedures to get responses to Security Questions and allow resetting password if all matches.
3. CreateAccoute.ASPX FrankP (Frank Petrillo)
   1. Validation All user input will be validated server-side.
   2. This page presents a DropDownList of security questions to all user to provide unique responses to implement “Recover Lost Login” functionality.
   3. Utilize RESTfull WebAPI’s to update the database with Account information.
   4. implement SMTP email to send a link to update/finalize the account registration.
   5. Redirect to the Login.ASPX page.
4. ~~AdminClient.ASPX page~~
5. UserClient.ASPX page (JimS)
   1. This is the main bulk of the WebApp and will be shared among team members.
6. Reusable Component: This competent will be loaded and updated in the code-behind.
   1. Custom User Controls (ASCX) Modal Help/About using AJAX and Update Panels (JimS)
   2. Custom User Controls (ASCX) Calendar Add (FrankP)
7. Dynamic Display
   1. Calendar Sidebar (JimS)
8. Data Model Design/SQL Code JimS & FrankP All database items prefixed with “TP\_”
   1. Dbo.Account table User credentials, account type and demographic info will be stored here.
   2. Dbo.AccountDocument table User official documents will be stored here. Business logic will enforce simple basic rules on appointment making. For instance making an appointment in a calendar of type “Airline” will require the Account joins to the AccountDocument table where the AccountDocumentType. = “Covid Passport”). For the beta, sample docs will be provided.
   3. Dbo.Calendar table Each event/place of business will have a row in the Calendar table. Calendar’s may be assigned a type to enforce simple business requirements to make an appointment. An alert will be raised to the User otherwise.
   4. Dbo.Event table Users will have rows in the table for every appointment.
   5. Dbo.EventAttachment table Users can upload attachments to share via the appointment. . For the beta sample docs will be provided.
9. Navigation System (JimS) A standard temple <header><nav> will be provided for a common NavBar. Each page will add <span>, <asp:Button> and <ul> elements as necessary. All styling via Bootstrap will be provided to properly position menu items.
10. Security (JimS and FrankP) The Web application will use ES Rijndael's Symmetric Encryption Algorithm to store password credentials as VARBINARY in the database. Create account will encrypt the password, and the login will decrypt the field. For more info see Class Example http://cis-iis2.temple.edu/users/pascucci/cis3342/EncryptionExample1.aspx.
11. (JimS): Will update once we’ve had lecture on this topic.

Summary List

JimS/FrankP **Data Model**

Create SQL scripts (see calendar-db-install.sql in root folder of zip)

Create an ERD (see Calendar-ERD.pdf in root folder of zip)

JimS **LogIn/Security Questions/ResetPassword Pages (using SOAP WebService)**

LogIn method()

SecurityQuestions() method

UpdatePassword() method

JimS **Remember Me**

Encrypts password in cookie for "Remember Me" check box.

1. Log in with user credentials stored in cookie available

2. Manually LogIn otherwise

JimS **Security**

Password encryption (see Account.Encrypt method)

JimS **UserClient Page** (see UserClient.aspx) Implements SOAP WebMethods

JimS **Login Page** (see Login.aspx) Implements SOAP WebMethods

Jims **ForgotPassword Page** (see ForgotPassword.aspx) Implements SOAP WebMethods

JimS **Calendar** table with Core WebAPI Calendar ControllerActions GET/POST/PUT/DELETE

JimS **Event** db tableusing Core WebAPI Event ControllerActions GET/POST/PUT/DELETE

JimS **Dynamic Data Displays**

UserClient.aspx Side Bar of Calendar CheckBoxes controls and a Button for Calendar Options

JimS **Stored Procuedures**

TP\_Account\_Document\_Insert\_SP

TP\_Calendar\_Insert\_SP

TP\_Event\_Insert\_SP

TP\_GetAccountId\_SP]

TP\_TP\_Account\_Insert\_SP

TP\_Account\_Document\_Insert\_SP

TP\_Account\_Login\_SP

TP\_Account\_Security\_Questions\_SP

TP\_Account\_Update\_Password\_SP

TP\_Calendar\_Delete\_SP

TP\_Calendar\_Insert\_SP

TP\_Calendar\_Select\_SP

TP\_Calendar\_Update\_SP

TP\_Event\_Delete\_SP

TP\_Event\_Insert\_SP

TP\_Event\_Select\_By\_EventId\_SP

TP\_Event\_Select\_SP

TP\_Event\_Update\_SP

TP\_GetAccountId\_SP

TP\_SecurityQuestion\_SP

JimS **AJAX (Modal Help/About)** see ModalUserControl.ascx

JimS **Custom User Control** see ModalUserControl.ascx