

Table of Contents

About the Document.....	3
Information	4
DOX EMR – External API Specifications	5
Version of the software.....	12
Basic Requirements to use and configure API.....	13
API TERMS – DOX EMR.....	19
Section 1: Account and Registration	19
Accepting the Terms	19
Entity Level Acceptance	19
Registration.....	19
Subsidiaries and Affiliates	19
Section 2: Using Our APIs	19
Your End Users.....	19
Compliance with Law, Third Party Rights, and Other DOX Terms of Service	20
Permitted Access	20
API Limitations	20
Open Source Software	20
Communication with DOX.....	20
Feedback.....	20
Non-Exclusivity.....	20
Section 3: API Clients	21
API Clients and Monitoring.....	21
Security	21
Ownership.....	21
User Privacy and API Clients	21
Section 4: Prohibitions and Confidentiality	21
API Prohibitions.....	21
Confidential Matters.....	22
Section 5: Content.....	22

Content Accessible Through our APIs	22
Submission of Content	23
Retrieval of content	23
Data Portability	23
Prohibitions on Content.....	23
Section 6: Brand Features; Attribution.....	24
Brand Features.....	24
Attribution.....	24
Publicity.....	24
Promotional and Marketing Use.....	24
Section 7: Privacy and Copyright Protection.....	24
DOX DMCA Policy.....	24
Section 8: Termination	25
Termination.....	25
Your Obligations Post-Termination.....	25
Surviving Provisions	25
Section 9: Liability for our APIs	25
WARRANTIES.....	25
LIMITATION OF LIABILITY	25
Indemnification	26
Section 10: General Provisions	26
Modification.....	26
U.S. Federal Agency Entities	26
General Legal Terms	26

About the Document

Name of the Document	DOX External API Specs.docx
Purpose of the Document	To give a better understanding for integrating DOX External API to the external clients
Document Owner	P V Narasimha Rao
Date of First Draft	10 th November 2017
Date of Final Draft Approved	4 th January 2018
Date of Second Draft Approved	26 th February 2018
Approved By	Bart Ripperger
Date	27 th February 2018
Remarks & Comments	

Information

Company Name : **DOX EMR**

Address : Bart Ripperger, DPM
20235 N Cave Creek Rd
104 - 265
Phoenix, Az 85024
C: 602 628-2880
B: 866 301-7700
F: 480 393-1841
Mail : bripperger@doxemr.com

Website <http://www.doxemr.com/>

Hyperlink : <https://www.doxemr.com/onc-health-it-costs-limitations-disclos> to view website that shows API information and terms.

Hyperlink for API information and terms documentation:
<https://podiatry.doxemr.net/DoxExtAPI/DOXAPI-Application-Access.pdf>

DOX EMR – External API Specifications

GetUserToken

- **URL**

<https://podiatry.doxemr.net/DoxExtAPI/v1/GetUserToken>

- **Usage**

It validates the user credentials issued by end point server, then generates a user token to end point server.

- **Method**

GET : Request type

- **URL Params (Required)**

PracticeID=[integer]

UserID=[integer]

Password=[alphanumeric]

- **Success Response**

UserToken=[alphanumeric] {Returns an unique user token which will expire after 60 minutes or user logoff}

Content = { }

- **Error Response**

Code= [401 Invalid Practice Details]

Content= {error: “Invalid Practice Details” }

Code= [501 Invalid User Details]

Content = {error: “Invalid User Details”}

GetUserRolewiseUIDetails

- **URL**

<https://podiatry.doxemr.net/DoxExtAPI/v1/GetUserRolewiseUIDetails>

- **Usage**

It validates the UserToken for a valid enduser call, returns the patient details to end point server.

- **Method**

GET : Request type

- **URL Params (Required)**

UserToken =[alphanumeric]

- **Success Response**

Json=[alphanumeric] {Returns an unique user token which will expire after 60 minutes or user logoff}

Content = {

<DOB></DOB> Date of Birth <FirstName> </FirstName>
First name of the user <LastName> /</LastName> Last name of
the user <MiddleName> </MiddleName> Middle name of the
user <Number></Number> Phone number of the user
<Response/> Response code

}

- **Error Response**

Code=[401 Invalid token details]

Content= { error : "Invalid token details" }

Code=[601 User is in Inactive state]

Content = { error : "User is in Inactive state" }

GetPatientAppointments

- **URL**

<https://podiatry.doxemr.net/DoxExtAPI/v1/GetPatientAppointments>

- **Usage**

It validates the UserToken for a valid enduser call, returns patient appointment details to end point server.

- **Method**

GET : Request type

- **URL Params (Required)**

UserToken=[alphanumeric]

LastName=[alphabetic]

FirstName=[alphabetic]

MiddleName=[alphabetic]

DOB=[date]

Phone=[numeric]

- **Success Response**

Json=[alphanumeric] {Returns the requested patient appointment details }

Content = {

<AccessView></AccessView>	Access given to view
encounter details	
<AppointmentId></ AppointmentId >	Appointment Id
<ApptDateTime></ApptDateTime>	Appointment date and time
<DownloadTime></DownloadTime>	Encounter details downloaded time
stamp	
<Location> </Location>	Appointment location name
<Response/>	Response code
<ViewedTime></ViewedTime>	Encounter details viewed
time stamp	
}	

- **Error Response**

Code=[401 Invalid token details]

Content= { error : "Invalid token details" }

Code=[402 Patient does not exists]

Content = { error : "Patient does not exists"}

Code=[403 More than one record exists with this details]

Content = { error : "More than one record exists with this details"}

Code=[601 User is in inactive state]

Content = { error : "User is in inactive state"}

GetPatientSummarizedData

- **URL**

<https://podiatry.doxemr.net/DoxExtAPI/v1/GetPatientSummarizedData>

- **Usage**

It validates the UserToken for a valid enduser call, returns patient summarized data to end point server.

- **Method**

GET : Request type

- **URL Params (Required)**

UserToken=[alphanumeric]

AppointmentId=[numeric]

- **Success Response**

Json=[alphanumeric] {Returns the patient summarized data as an xml in valid CCDA format }

Content = {

 <Response/>

Response code

 <strxml></ strxml >

Patient summarixed data xml

}

- **Error Response**

Code=[401 Invalid token details]

Content= { error : “Invalid token details” }

Code=[601 User is in inactive state]

Content = { error : “User is in inactive state”}

UpdateAppointmentDataViewDate

- **URL**

<https://podiatry.doxemr.net/DoxExtAPI/v1/UpdateAppointmentDataViewDate>

- **Usage**

It validates the UserToken for a valid enduser call, updates the appointment viewed date of that appointment.

- **Method**

SET: Request type

- **URL Params (Required)**

UserToken=[alphanumeric]

AppointmentId=[numeric]

- **Success Response**

Json=[alphanumeric] {Updates the system time stamp as appointment viewed date of that appointment }

Content = {

 <Response/>

Response code

}

- **Error Response**

Code=[401 Invalid token details]

Content= { error : "Invalid token details" }

Code=[601 User is in inactive state]

Content = { error : "User is in inactive state" }

UpdateAppointmentDataDownloadDate

- **URL**

<https://podiatry.doxemr.net/DoxExtAPI/v1/UpdateAppointmentDataDownloadDate>

- **Usage**

It validates the UserToken for a valid enduser call, updates the appointment download date of that appointment.

- **Method**

SET: Request type

- **URL Params (Required)**

UserToken=[alphanumeric]

AppointmentId=[numeric]

- **Success Response**

Json=[alphanumeric] {Updates the system time stamp as appointment download date of that appointment }

Content = {

 <Response/>

Response code

}

- **Error Response**

Code=[401 Invalid token details]

Content= { error : "Invalid token details" }

Code=[601 User is in inactive state]

Content = { error : "User is in inactive state" }

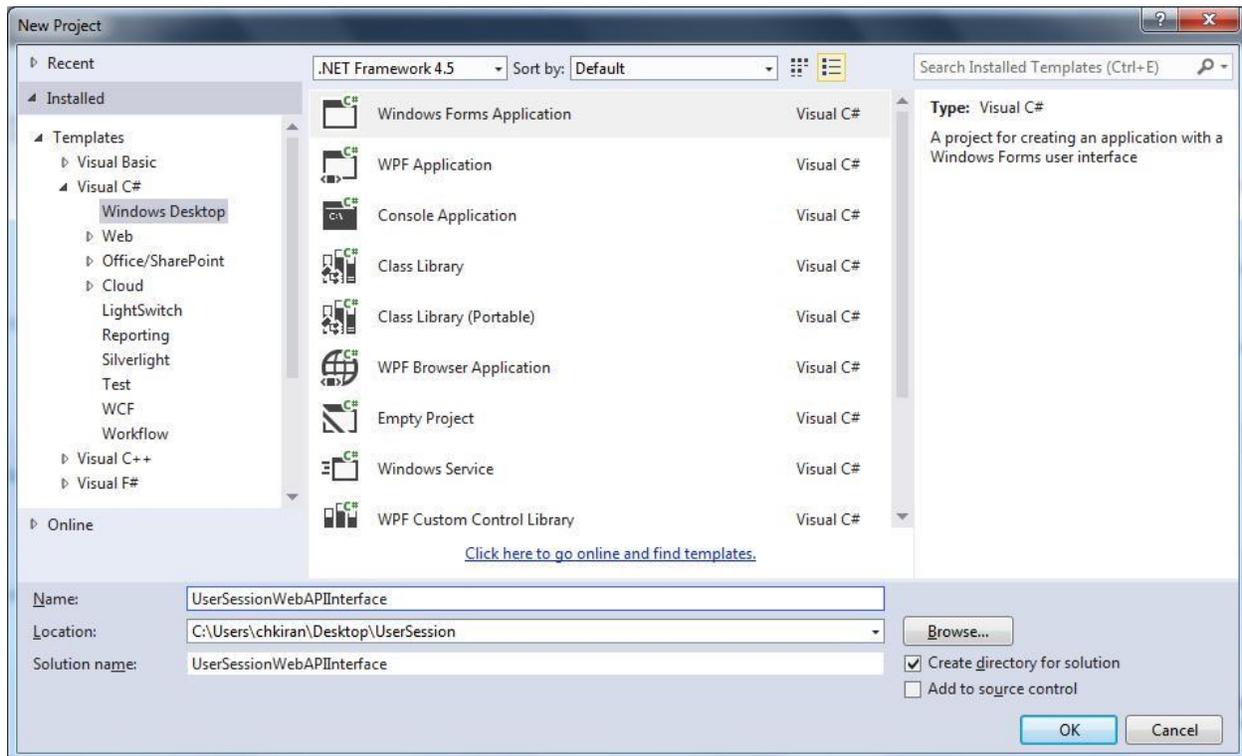
Basic Requirements to use and configure API

DOX used MVC 4.0 web API (RESTful web service) to develop this API. REST technology is generally preferred to the more robust Simple Object Access Protocol (SOAP) technology because REST leverages less band width, making it more suitable for internet usage. This API is code which allows other software programs to communicate with DOX PA database to retrieve the patient details along with CCDA files.

DOX API can be integrated to any programming languages which had the capability to support RESTful web services like MVC 4.0, Java, Ruby, PHP.

Below is a sample code written in MVC 4.0 to integrate the DOX API to use the methods exposed in the API.

Created new project for Integrating UserSessionAPI



GetUserToken:

```
UserSessionWebAPIInterface - Microsoft Visual Studio
FILE EDIT VIEW PROJECT BUILD DEBUG TEAM TOOLS VISUALSVN TEST ANALYZE WINDOW HELP
Debug Any CPU
Form1.Designer.cs Form1.cs Form1.cs [Design]
UserSessionWebAPIInterface UserSessionWebAPIInterface.Form1 btnInactivateUserSession_Click(object sender, EventArgs e)

private void btnVallUserToken_Click(object sender, EventArgs e)
{
    try
    {
        // Declarations for passing parameters to URL
        int PracticeID = 1250;
        string UserID = "Smith";
        string Password = "Adam";
        // Passing the parameters to URL
        string URI = "https://doxweb.doxemr.com/DoxExtAPI/v1/GetUserToken" + "/" + PracticeID + "," + UserID + "," + Password;

        // Create object on WebClient
        WebClient webClient = new WebClient();

        // Get the JSON Response from webClient.DownloadString
        var strTokenResponse = webClient.DownloadString(URI);
        DataTable dtTokenModel = ConvertJSONToDataTable(strTokenResponse);
        strTokenResponse = dtTokenModel.Rows[0]["Response"].ToString();

        //we can get the Token value and we can use the token for other methods
        string strToken = dtTokenModel.Rows[0]["Token"].ToString();

        //checking with the strTokenResponse validation
        if (strTokenResponse == "401")
        {
            MessageBox.Show("Invalid Practice Details", "Dox External API Data Receiver", MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
        }
        else if (strTokenResponse == "501")
        {
            MessageBox.Show("Invalid User Details", "Dox External API Data Receiver", MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
        }
    }
    catch (WebException ex)
    {
        MessageBox.Show(ex.Message);
    }
}
100 % Ln 482 Col 13 Ch 13 INS
```

GetPatientAppointments:

```
UserSessionWebAPIInterface - Microsoft Visual Studio
FILE EDIT VIEW PROJECT BUILD DEBUG TEAM TOOLS VISUALSVN TEST ANALYZE WINDOW HELP
Debug Any CPU
Form1.Designer.cs Form1.cs Form1.cs [Design]
UserSessionWebAPIInterface UserSessionWebAPIInterface.Form1 btnGetAppointments_Click(object sender, EventArgs e)

private void btnGetAppointments_Click(object sender, EventArgs e)
{
    try
    {
        // Declarations for passing parameters to URL
        string LastName = "Brotman";
        string FirstName = "Rachel";
        string MiddleName = "Dae";
        string DOB = "05/06/1985";
        string PhoneNo = "9854569555";
        string strToken = "D17E839C-3779-46DC-924E-50A2164D9F31";
        string Params = "";

        //the format must be in yyyyMMdd
        DOB = dtpPatDOB.Value.ToString("yyyyMMdd");

        // Building of Params to send to Rest Service Method.
        Params = strToken.Trim() + ',' + LastName.Trim() + ',' + FirstName.Trim() + ',' + MiddleName.Trim() + ',' + DOB.Trim() + ',' + PhoneNo.Trim();

        //Code to accept all the Security Certificates
        ServicePointManager.ServerCertificateValidationCallback += new System.Net.Security.RemoteCertificateValidationCallback(ValidateSecurityCertificate);

        // Create object on WebClient
        WebClient webClient = new WebClient();

        // Pass the parameters to URL
        String URI = "http://demoapps.icreatech.net/DoxExtAPI/v1" + "/" + "GetPatientAppointments" + "/" + Params;

        // Get the JSON Response from webClient.DownloadString
        var ApptModel = webClient.DownloadString(URI);

        DataTable dtApptModel = JsonResult(ApptModel);
        string strResponseStatus = dtApptModel.Rows[0]["Response"].ToString();
    }
}
100 % Item(s) Saved Ln 236 Col 17 Ch 17 INS
```

```
string strResponseStatus = dtApptModel.Rows[0]["Response"].ToString();

//checking with the strResponseStatus validation
if (strResponseStatus == "401")
{
    MessageBox.Show("Invalid Token Details", "Dox External API Data Receiver", MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
}
else if (strResponseStatus == "402")
{
    MessageBox.Show("Patient Does not exists", "Dox External API Data Receiver", MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
}
else if (strResponseStatus == "403")
{
    MessageBox.Show("More than one record exists with this details", "Dox External API Data Receiver", MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
}
else if (strResponseStatus == "501")
{
    MessageBox.Show("Please Enter Patient Mandatory fields (LastName,FirstName,DOB,Phone Number)", "Dox External API Data Receiver", MessageBoxButtons.OK, Message
}
else if (strResponseStatus == "502")
{
    MessageBox.Show("Please enter Patient DOB in yyyy#Mdd", "Dox External API Data Receiver", MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
}
else if (strResponseStatus == "601")
{
    MessageBox.Show("User Session Expired!!", "Dox External API Data Receiver", MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
}
}
catch (WebException ex)
{
    throw ex;
}
}
```

GetPatientSummarizedData:

```
private void btnPatientSummarizedData_Click(object sender, EventArgs e)
{
    try
    {
        // Declarations for passing parameters to URL
        string strToken = "798BD24F2-8861-495A-9598-6C618D1B1EF4";
        int ApptID = 5345;

        //Code ends to accept all the Security Certificates
        ServicePointManager.ServerCertificateValidationCallback += new System.Net.Security.RemoteCertificateValidationCallback(ValidateSecurityCertificate);

        // Create object on WebClient
        WebClient webClient = new WebClient();
        string strCCDURI = "";
        string strPatientDataType = "";

        if (strPatientDataType == "GetPatientSummarizedData")
        {
            // Pass the parameters to URL
            strCCDURI = "http://demoapps.icreatech.net/DoxExtAPI/V1" + "/" + "GetPatientSummarizedData" + "/" + strToken.Trim() + ',' + ApptID;

            // Get the JSON Response from webClient.DownloadString
            string strCCDResponse = webClient.DownloadString(strCCDURI);

            DataTable dtCCDResponse = new DataTable();
            dtCCDResponse = ConvertJSONToDataTable(strCCDResponse);

            //checking with the dtCCDResponse validation
            if (dtCCDResponse.Rows[0]["Response"].ToString() == "401")
            {
                MessageBox.Show("Invalid Token Details", "Dox External API Data Receiver", MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
                return;
            }
            else if (dtCCDResponse.Rows[0]["Response"].ToString() == "601")
            {
            }
        }
    }
}
```

```
UserSessionWebAPIInterface - Microsoft Visual Studio
FILE EDIT VIEW PROJECT BUILD DEBUG TEAM TOOLS VISUALSVN TEST ANALYZE WINDOW HELP
Debug Any CPU
Form1.Designer.cs Form1.cs Form1.cs [Design]
UserSessionWebAPIInterface UserSessionWebAPIInterface.Form1 btnPatientSummerizedData_Click(object sender, EventArgs e)

// Pass the parameters to URL
strCCDURI = "http://demoapps.icreatech.net/DoxExtAPI/V1/" + "/" + "GetPatientSummerizedData" + "/" + strToken.Trim() + ',' + ApptID;

// Get the J5on Response from webClient.DownloadString
string strCCDResponse = webClient.DownloadString(strCCDURI);

DataTable dtCCDResponse = new DataTable();
dtCCDResponse = ConvertJSONToDataTable(strCCDResponse);

//checking with the dtCCDResponse validation
if (dtCCDResponse.Rows[0]["Response"].ToString() == "401")
{
    MessageBox.Show("Invalid Token Details", "Dox External API Data Receiver", MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
    return;
}
else if (dtCCDResponse.Rows[0]["Response"].ToString() == "601")
{
    MessageBox.Show("User is in InActive State", "Dox External API Data Receiver", MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
    return;
}

byte[] newBytes = null;
string strConvertXML = dtCCDResponse.Rows[0]["CCDData"].ToString().Replace("\n", string.Empty);
newBytes = Convert.FromBase64String(strConvertXML);
// Get the J5on Response
strConvertXML = System.Text.Encoding.ASCII.GetString(newBytes);
}
catch (Exception ex)
{
    MessageBox.Show(ex.Message);
}
}

Ready Ln 326 Col 80 Ch 80 INS
```

GetPatientData:

```
UserSessionWebAPIInterface - Microsoft Visual Studio
FILE EDIT VIEW PROJECT BUILD DEBUG TEAM TOOLS VISUALSVN TEST ANALYZE WINDOW HELP
Debug Any CPU
Form1.Designer.cs Form1.cs Form1.cs [Design]
UserSessionWebAPIInterface UserSessionWebAPIInterface.Form1 btnPatientSummerizedData_Click(object sender, EventArgs e)

private void btnGetPatientData_Click(object sender, EventArgs e)
{
    // Declarations for passing parameters to URL
    string strCCDURI = "";
    string strPatientDataType = "";
    string strToken = "197CA482-37AC-48E6-998F-8D55323AFB0A";
    int ApptID = 5345;
    if (strPatientDataType == "GetPatientData")
    {
        string checkedparams = "0,1,0,0,0,0,0,0,0,0";

        // Pass the parameters to URL
        strCCDURI = "http://demoapps.icreatech.net/DoxExtAPI/V1/" + "/" + "GetPatientData" + "/" + strToken.Trim() + ',' + ApptID + "," + checkedparams;
    }

    // Create object on WebClient
    WebClient webClient = new WebClient();

    // Get the J5on Response from webClient.DownloadString
    string strCCDResponse = webClient.DownloadString(strCCDURI);

    DataTable dtCCDResponse = new DataTable();
    dtCCDResponse = ConvertJSONToDataTable(strCCDResponse);

    //checking with the dtCCDResponse validation
    if (dtCCDResponse.Rows[0]["Response"].ToString() == "401")
    {
        MessageBox.Show("Invalid Token Details", "Dox External API Data Receiver", MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
        return;
    }
    else if (dtCCDResponse.Rows[0]["Response"].ToString() == "601")
    {
        MessageBox.Show("User is in InActive State", "Dox External API Data Receiver", MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
        return;
    }
}

Ready Ln 326 Col 80 Ch 80 INS
```

UpdateAppointmentDataViewDate:

```
UserSessionWebAPIInterface - Microsoft Visual Studio
FILE EDIT VIEW PROJECT BUILD DEBUG TEAM TOOLS VISUALSVN TEST ANALYZE WINDOW HELP
Debug Any CPU
Form1.Designer.cs Form1.cs Form1.cs [Design]
UserSessionWebAPIInterface UserSessionWebAPIInterface.Form1 btnUpdateAppointmentDataViewDate_Click(object sender, EventArgs e)
private void btnUpdateAppointmentDataViewDate_Click(object sender, EventArgs e)
{
    try
    {
        // Declarations for passing parameters to URL
        string strToken = "95193732-0FEE-4772-B037-E6C4192D9ED4";
        int ApptID = 5345;
        //Code ends to accept all the Security Certificates
        ServicePointManager.ServerCertificateValidationCallback += new System.Net.Security.RemoteCertificateValidationCallback(ValidateSecurityCertificate);
        // Create object on WebClient
        WebClient webClient = new WebClient();
        // Pass the parameters to URL
        string ApptDataViewDateURI = "http://demoapps.icreatech.net/DoxExtAPI/V1" + "/" + "UpdateAppointmentDataViewDate" + "/" + strToken.Trim() + "," + ApptID;
        // Get the JSon Response from webClient.DownloadString
        var ResponseApptDataViewDateTime = webClient.DownloadString(ApptDataViewDateURI);
        DataTable dtResponseApptDataViewDateTime = new DataTable();
        dtResponseApptDataViewDateTime = ConvertJSONTODataTable(ResponseApptDataViewDateTime);
        if (dtResponseApptDataViewDateTime.Rows[0]["Response"].ToString() == "401")
        {
            MessageBox.Show("Invalid Token Details", "Dox External API Data Receiver", MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
            return;
        }
        else if (dtResponseApptDataViewDateTime.Rows[0]["Response"].ToString() == "601")
        {
            MessageBox.Show(" User is in InActive State", "Dox External API Data Receiver", MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
            return;
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message);
    }
}
100% Ln 392 Col 1 Ch 1 INS
```

UpdateAppointmentDataDownloadDate:

```
UserSessionWebAPIInterface - Microsoft Visual Studio
FILE EDIT VIEW PROJECT BUILD DEBUG TEAM TOOLS VISUALSVN TEST ANALYZE WINDOW HELP
Debug Any CPU
Form1.Designer.cs Form1.cs Form1.cs [Design]
UserSessionWebAPIInterface UserSessionWebAPIInterface.Form1 btnUpdateAppointmentDataDownloadDate_Click(object sender, EventArgs e)
private void btnUpdateAppointmentDataDownloadDate_Click(object sender, EventArgs e)
{
    try
    {
        // Declarations for passing parameters to URL
        int ApptID = 5345;
        string strToken = "95193732-0FEE-4772-B037-E6C4192D9ED4";
        //Code ends to accept all the Security Certificates
        ServicePointManager.ServerCertificateValidationCallback += new System.Net.Security.RemoteCertificateValidationCallback(ValidateSecurityCertificate);
        // Create object on WebClient
        WebClient webClient = new WebClient();
        //Authorization
        // Pass the parameters to URL
        string ApptDataViewDateURI = "http://demoapps.icreatech.net/DoxExtAPI/V1" + "/" + "UpdateAppointmentDataDownloadDate" + "/" + strToken.Trim() + "," + ApptID;
        // Get the JSon Response from webClient.DownloadString
        var ResponseApptDataViewDateTime = webClient.DownloadString(ApptDataViewDateURI);
        DataTable dtResponseApptDataViewDateTime = new DataTable();
        dtResponseApptDataViewDateTime = ConvertJSONTODataTable(ResponseApptDataViewDateTime);
        if (dtResponseApptDataViewDateTime.Rows[0]["Response"].ToString() == "401")
        {
            MessageBox.Show("Invalid Token Details", "Dox External API Data Receiver", MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
            return;
        }
        else if (dtResponseApptDataViewDateTime.Rows[0]["Response"].ToString() == "601")
        {
            MessageBox.Show(" User is in InActive State", "Dox External API Data Receiver", MessageBoxButtons.OK, MessageBoxIcon.Exclamation);
            return;
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message);
    }
}
100% Ln 433 Col 1 Ch 1 INS
```

InActivateUserSession:

UserSessionWebAPIInterface - Microsoft Visual Studio

FILE EDIT VIEW PROJECT BUILD DEBUG TEAM TOOLS VISUALSVN TEST ANALYZE WINDOW HELP

Start Debug Any CPU

Form1.Designer.cs Form1.cs Form1.cs [Design]

UserSessionWebAPIInterface UserSessionWebAPIInterface.Form1 btnUpdateAppointmentDataDownloadDate_Click(object sender, EventArgs e)

```
private void btnInactivateUserSession_Click(object sender, EventArgs e)
{
    string strToken = "95193732-0FEE-4772-8037-E6C4192D9ED4";

    //Code to accept all the Security Certificates
    ServicePointManager.ServerCertificateValidationCallback += new System.Net.Security.RemoteCertificateValidationCallback(ValidateSecurityCertificate);
    //Code ends to accept all the Security Certificates

    // Create object on WebClient
    WebClient webClient = new WebClient();

    // Pass the parameters to URL
    String URI = "http://demoapps.icreatech.net/DoExtAPI/V1" + "/" + "InActivateUserSession" + "/" + strToken.Trim();

    // Get the JSON Response from webClient.DownloadString
    string response = webClient.DownloadString(URI);

    this.Close();
}
}
```

100 % Ln 433 Col 1 Ch 1 INS

API TERMS – DOX EMR

By accessing or using our APIs, you are agreeing to the terms below. If there is a conflict between these terms and additional terms applicable to a given API, the additional terms will control for that conflict. Collectively, we refer to the terms below, any additional terms, terms within the accompanying API documentation, and any applicable policies and guidelines as the "Terms." You agree to comply with the Terms and that the Terms control your relationship with us. So please read all the Terms carefully

Under the Terms, "**DOX**" means DOX EMR, with offices: DOX EMR, 20235 N Cave Creek Rd, # 104 – 265, Phoenix, Az 85024, unless set forth otherwise in additional terms applicable for a given API. We may refer to "DOX" as "we", "our", or "us" in the Terms.

Section 1: Account and Registration

Accepting the Terms

You may not use the APIs and may not accept the Terms if (a) you are not of legal age to form a binding contract with DOX, or (b) you are a person barred from using or receiving the APIs under the applicable laws of the United States or other countries including the country in which you are resident or from which you use the APIs.

Entity Level Acceptance

If you are using the APIs on behalf of an entity, you represent and warrant that you have authority to bind that entity to the Terms and by accepting the Terms, you are doing so on behalf of that entity (and all references to "you" in the Terms refer to that entity).

Registration

In order to access certain APIs you may be required to provide certain information (such as identification or contact details) as part of the registration process for the APIs, or as part of your continued use of the APIs. Any registration information you give to DOX will always be accurate and up to date and you'll inform us promptly of any updates.

Subsidiaries and Affiliates

DOX has subsidiaries and affiliated legal entities around the world. These companies may provide the APIs to you on behalf of DOX and the Terms will also govern your relationship with these companies.

Section 2: Using Our APIs

Your End Users

You will require your end users to comply with (and not knowingly enable them to violate) applicable law, regulation, and the Terms.

Compliance with Law, Third Party Rights, and Other DOX Terms of Service

You will comply with all applicable law, regulation, and third party rights (including without limitation laws regarding the import or export of data or software, privacy, and local laws). You will not use the APIs to encourage or promote illegal activity or violation of third party rights. You will not violate any other terms of service with DOX (or its affiliates).

Permitted Access

You will only access (or attempt to access) an API by the means described in the documentation of that API. If DOX assigns you developer credentials (e.g. client IDs), you must use them with the applicable APIs. You will not misrepresent or mask either your identity or your API Client's identity when using the APIs or developer accounts.

API Limitations

DOX sets and enforces limits on your use of the APIs (e.g. limiting the number of API requests that you may make or the number of users you may serve), in our sole discretion. You agree to, and will not attempt to circumvent, such limitations documented with each API. If you would like to use any API beyond these limits, you must obtain DOX's express consent (and DOX may decline such request or condition acceptance on your agreement to additional terms and/or charges for that use). To seek such approval, contact the relevant DOX API team for information (e.g. by using the DOX developer's console).

Open Source Software

Some of the software required by or included in our APIs may be offered under an open source license. Open source software licenses constitute separate written agreements. For certain APIs, open source software is listed in the documentation. To the limited extent the open source software license expressly supersedes the Terms, the open source license instead sets forth your agreement with DOX for the applicable open source software.

Communication with DOX

We may send you certain communications in connection with your use of the APIs. Please review the applicable API documentation for information about opting out of certain types of communication.

Feedback

If you provide feedback or suggestions about our APIs, then we (and those we allow) may use such information without obligation to you.

Non-Exclusivity

The Terms are non-exclusive. You acknowledge that DOX may develop products or services that may compete with the API Clients or any other products or services.

Section 3: API Clients

API Clients and Monitoring

The APIs are designed to help you enhance DOX client applications. YOU AGREE THAT DOX MAY MONITOR USE OF THE APIS TO ENSURE QUALITY, IMPROVE DOX PRODUCTS AND SERVICES, AND VERIFY YOUR COMPLIANCE WITH THE TERMS. This monitoring may include DOX accessing and using your API Client, for example to identify security issues that could affect DOX or its users. You will not interfere with this monitoring. DOX may use any technical means to overcome such interference. DOX may suspend access to the APIs by you or your API Client without notice if we reasonably believe that you are in violation of the Terms.

Security

You will use commercially reasonable efforts to protect user information collected by your API Client, including personally identifiable information ("PII"), from unauthorized access or use and will promptly report to your users any unauthorized access or use of such information to the extent required by applicable law.

Ownership

DOX does not acquire ownership in API Clients, and by using our APIs, you do not acquire ownership of any rights in our APIs or the content that is accessed through our APIs.

User Privacy and API Clients

You will comply with all applicable privacy laws and regulations including those applying to PII. You will provide and adhere to a privacy policy for your API Client that clearly and accurately describes to users of your API Client what user information you collect and how you use and share such information (including for advertising) with DOX and third parties.

Section 4: Prohibitions and Confidentiality

API Prohibitions

When using the APIs, you may not (or allow those acting on your behalf to):

1. Sublicense an API for use by a third party. Consequently, you will not create an API Client that functions substantially the same as the APIs and offer it for use by third parties.
2. Perform an action with the intent of introducing to DOX products and services any viruses, worms, defects, Trojan horses, malware, or any items of a destructive nature.
3. Defame, abuse, harass, stalk, or threaten others.
4. Interfere with or disrupt the APIs or the servers or networks providing the APIs.

5. Promote or facilitate unlawful online gambling or disruptive commercial messages or advertisements.
6. Reverse engineer or attempt to extract the source code from any API or any related software, except to the extent that this restriction is expressly prohibited by applicable law.
7. Use the APIs for any activities where the use or failure of the APIs could lead to death, personal injury, or environmental damage (such as the operation of nuclear facilities, air traffic control, or life support systems).
8. Use the APIs to process or store any data that is subject to the International Traffic in Arms Regulations maintained by the U.S. Department of State.
9. Remove, obscure, or alter any DOX terms of service or any links to or notices of those terms.

Unless otherwise specified in writing by DOX, DOX does not intend use of the APIs to create obligations under the Health Insurance Portability and Accountability Act, as amended ("HIPAA"), and makes no representations that the APIs satisfy HIPAA requirements. If you are (or become) a "covered entity" or "business associate" as defined in HIPAA, you will not use the APIs for any purpose or in any manner involving transmitting protected health information to DOX unless you have received prior written consent to such use from DOX.

Confidential Matters

1. Developer credentials (such as passwords, keys, and client IDs) are intended to be used by you and identify your API Client. You will keep your credentials confidential and make reasonable efforts to prevent and discourage other API Clients from using your credentials. Developer credentials may not be embedded in open source projects.
2. Our communications to you and our APIs may contain DOX confidential information. DOX confidential information includes any materials, communications, and information that are marked confidential or that would normally be considered confidential under the circumstances. If you receive any such information, then you will not disclose it to any third party without DOX's prior written consent. DOX confidential information does not include information that you independently developed, that was rightfully given to you by a third party without confidentiality obligation, or that becomes public through no fault of your own. You may disclose DOX confidential information when compelled to do so by law if you provide us reasonable prior notice, unless a court orders that we not receive notice.

Section 5: Content

Content Accessible Through our APIs

Our APIs contain some third party content (such as text, images, videos, audio, or software). This content is the sole responsibility of the person that makes it available. We may sometimes review

content to determine whether it is illegal or violates our policies or the Terms, and we may remove or refuse to display content. Finally, content accessible through our APIs may be subject to intellectual property rights, and, if so, you may not use it unless you are licensed to do so by the owner of that content or are otherwise permitted by law. Your access to the content provided by the API may be restricted, limited, or filtered in accordance with applicable law, regulation, and policy.

Submission of Content

Some of our APIs allow the submission of content. DOX does not acquire any ownership of any intellectual property rights in the content that you submit to our APIs through your API Client, except as expressly provided in the Terms. For the sole purpose of enabling DOX to provide, secure, and improve the APIs (and the related service(s)) and only in accordance with the applicable DOX privacy policies, you give DOX a perpetual, irrevocable, worldwide, sub licensable, royalty-free, and non-exclusive license to Use content submitted, posted, or displayed to or from the APIs through your API Client. "Use" means use, host, store, modify, communicate, and publish. Before you submit content to our APIs through your API Client, you will ensure that you have the necessary rights (including the necessary rights from your end users) to grant us the license.

Retrieval of content

When a user's non-public content is obtained through the APIs, you may not expose that content to other users or to third parties without explicit opt-in consent from that user.

Data Portability

DOX supports data portability. For as long as you use or store any user data that you obtained through the APIs, you agree to enable your users to export their equivalent data to other services or applications of their choice in a way that's substantially as fast and easy as exporting such data from DOX products and services, subject to applicable laws, and you agree that you will not make that data available to third parties who do not also abide by this obligation.

Prohibitions on Content

Unless expressly permitted by the content owner or by applicable law, you will not, and will not permit your end users or others acting on your behalf to, do the following with content returned from the APIs:

1. Scrape, build databases, or otherwise create permanent copies of such content, or keep cached copies longer than permitted by the cache header;
2. Copy, translate, modify, create a derivative work of, sell, lease, lend, convey, distribute, publicly display, or sublicense to any third party;
3. Misrepresent the source or ownership; or
4. Remove, obscure, or alter any copyright, trademark, or other proprietary rights notices; or falsify or delete any author attributions, legal notices, or other labels of the origin or source of material.

Section 6: Brand Features; Attribution

Brand Features

"Brand Features" is defined as the trade names, trademarks, service marks, logos, domain names, and other distinctive brand features of each party. Except where expressly stated, the Terms do not grant either party any right, title, or interest in or to the other party's Brand Features. All use by you of DOX's Brand Features (including any goodwill associated therewith) will inure to the benefit of DOX.

Attribution

You agree to display any attribution(s) required by DOX as described in the documentation for the API. DOX hereby grants to you a nontransferable, non sublicense able, nonexclusive license while the Terms are in effect to display DOX's Brand Features for the purpose of promoting or advertising that you use the APIs. You must only use the DOX Brand Features in accordance with the Terms and for the purpose of fulfilling your obligations under this Section. You understand and agree that DOX has the sole discretion to determine whether your attribution(s) and use of DOX's Brand Features are in accordance with the above requirements and guidelines.

Publicity

You will not make any statement regarding your use of an API which suggests partnership with, sponsorship by, or endorsement by DOX without DOX's prior written approval.

Promotional and Marketing Use

In the course of promoting, marketing, or demonstrating the APIs you are using and the associated DOX products, DOX may produce and distribute incidental depictions, including screenshots, video, or other content from your API Client, and may use your company or product name. You grant us all necessary rights for the above purposes.

Section 7: Privacy and Copyright Protection

DOX DMCA Policy

We provide information to help copyright holders manage their intellectual property online, but we can't determine whether something is being used legally or not without their input. We respond to notices of alleged copyright infringement and terminate accounts of repeat infringers according to the process set out in the U.S. Digital Millennium Copyright Act. If you think somebody is violating your copyrights and want to notify us, you can find information about submitting notices and DOX's policy about responding to notices.

Section 8: Termination

Termination

You may stop using our APIs at any time with prior notice as described in your user agreement with DOX EMR. Further, upon termination, cease your use of the applicable APIs. DOX reserves the right to terminate the Terms with you or discontinue the APIs or any portion or feature or your access thereto for any reason and at any time without liability or other obligation to you.

Your Obligations Post-Termination

Upon any termination of the Terms or discontinuation of your access to an API, you will immediately stop using the API, cease all use of the DOX Brand Features, and delete any cached or stored content that was permitted by the cache header under Section 5. DOX may independently communicate with any account owner whose account(s) are associated with your API Client and developer credentials to provide notice of the termination of your right to use an API.

Surviving Provisions

When the Terms come to an end, those terms that by their nature are intended to continue indefinitely will continue to apply, including but not limited to: Sections 4(b), 5, 8, 9, and 10.

Section 9: Liability for our APIs

WARRANTIES

EXCEPT AS EXPRESSLY SET OUT IN THE TERMS, NEITHER DOX NOR ITS SUPPLIERS OR DISTRIBUTORS MAKE ANY SPECIFIC PROMISES ABOUT THE APIS. FOR EXAMPLE, WE DON'T MAKE ANY COMMITMENTS ABOUT THE CONTENT ACCESSED THROUGH THE APIS, THE SPECIFIC FUNCTIONS OF THE APIS, OR THEIR RELIABILITY, AVAILABILITY, OR ABILITY TO MEET YOUR NEEDS. WE PROVIDE THE APIS "AS IS".

SOME JURISDICTIONS PROVIDE FOR CERTAIN WARRANTIES, LIKE THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT. EXCEPT AS EXPRESSLY PROVIDED FOR IN THE TERMS, TO THE EXTENT PERMITTED BY LAW, WE EXCLUDE ALL WARRANTIES, GUARANTEES, CONDITIONS, REPRESENTATIONS, AND UNDERTAKINGS.

LIMITATION OF LIABILITY

WHEN PERMITTED BY LAW, DOX, AND DOX'S SUPPLIERS AND DISTRIBUTORS, WILL NOT BE RESPONSIBLE FOR LOST PROFITS, REVENUES, OR DATA; FINANCIAL LOSSES; OR INDIRECT, SPECIAL, CONSEQUENTIAL, EXEMPLARY, OR PUNITIVE DAMAGES.

TO THE EXTENT PERMITTED BY LAW, THE TOTAL LIABILITY OF DOX, AND ITS SUPPLIERS AND DISTRIBUTORS, FOR ANY CLAIM UNDER THE TERMS, INCLUDING FOR ANY IMPLIED WARRANTIES, IS LIMITED TO THE AMOUNT YOU PAID US TO USE THE APPLICABLE APIS (OR, IF WE CHOOSE, TO SUPPLYING YOU THE APIS AGAIN) DURING THE SIX MONTHS PRIOR TO THE EVENT GIVING RISE TO THE LIABILITY.

IN ALL CASES, DOX, AND ITS SUPPLIERS AND DISTRIBUTORS, WILL NOT BE LIABLE FOR ANY EXPENSE, LOSS, OR DAMAGE THAT IS NOT REASONABLY FORESEEABLE.

Indemnification

Unless prohibited by applicable law, if you are a business, you will defend and indemnify DOX, and its affiliates, directors, officers, employees, and users, against all liabilities, damages, losses, costs, fees (including legal fees), and expenses relating to any allegation or third-party legal proceeding to the extent arising from:

1. your misuse or your end user's misuse of the APIs;
2. your violation or your end user's violation of the Terms; or
3. any content or data routed into or used with the APIs by you, those acting on your behalf, or your end users.

Section 10: General Provisions

Modification

We may modify the Terms or any portion to, for example, reflect changes to the law or changes to our APIs. You should look at the Terms regularly. We'll post notice of modifications to the Terms within the documentation of each applicable API, to this website, and/or in the DOX developers console. Changes will not apply retroactively and will become effective no sooner than 30 days after they are posted. But changes addressing new functions for an API or changes made for legal reasons will be effective immediately. If you do not agree to the modified Terms for an API, you should discontinue your use of that API. Your continued use of the API constitutes your acceptance of the modified Terms.

U.S. Federal Agency Entities

The APIs were developed solely at private expense and are commercial computer software and related documentation within the meaning of the applicable U.S. Federal Acquisition Regulation and agency supplements thereto.

General Legal Terms

We each agree to contract in the English language. If we provide a translation of the Terms, we do so for your convenience only and the English Terms will solely govern our relationship. The Terms do not create any third party beneficiary rights or any agency, partnership, or joint venture. Nothing in the Terms will limit either party's ability to seek injunctive relief. We are not liable for failure or delay in performance to the extent caused by circumstances beyond our reasonable control. If you do not comply with the Terms, and DOX does not take action right away, this does not mean that DOX is giving up any rights that it may have (such as taking action in the future). If it turns out that a particular term is not enforceable, this will not affect any other terms. The Terms are the entire agreement between you and DOX relating to its subject and supersede any prior or contemporaneous agreements on that subject. For information about how to contact DOX, please visit our website

Except as set forth below: (i) the laws of California, U.S.A., excluding California's conflict of laws rules, will apply to any disputes arising out of or related to the Terms or the APIs and (ii) ALL CLAIMS ARISING OUT OF OR RELATING TO THE TERMS OR THE APIS WILL BE LITIGATED EXCLUSIVELY IN THE FEDERAL OR STATE COURTS OF SANTA CLARA COUNTY, CALIFORNIA, USA, AND YOU AND DOX CONSENT TO PERSONAL JURISDICTION IN THOSE COURTS.

If you are accepting the Terms on behalf of a United States federal government entity, then the following applies instead of the paragraph above: the laws of the United States of America, excluding its conflict of laws rules, will apply to any disputes arising out of or related to the Terms or the APIs. Solely to the extent permitted by United States Federal law: (i) the laws of the State of California (excluding California's conflict of laws rules) will apply in the absence of applicable federal law; and (ii) FOR ALL CLAIMS ARISING OUT OF OR RELATING TO THE TERMS OR THE APIS, THE PARTIES CONSENT TO PERSONAL JURISDICTION IN, AND THE EXCLUSIVE VENUE OF, THE COURTS IN SANTA CLARA COUNTY, CALIFORNIA.

If you are accepting the Terms on behalf of a United States city, county, or state government entity, then the following applies instead of the paragraph above: the parties agree to remain silent regarding governing law and venue.