**Data Export to Web (web service) Specifications**

 Data export to web feature sends (posts) assessment export files (.csv) to a web location specified by the app user. This app feature requires the web service security in the form of basic authentication. Web service should be set to accept basic authentication. Web service implementation should also acknowledge successful transfer of the file in the form of json object and http response code of 200 (OK). Any other response code other than **200** is considered invalid by the app. NIH Toolbox app only sends generates exports in csv format, so web service should be able to accept csv file format.

The app only accepts web service implemented with the following rules:

* URL set with https protocol.
* Security: basic authentication
	+ The app validates the web service in the form of basic authentication. App users could provide url, user id and password from the app’s “Settings” screen and test the connection to the web service.
* Form action: POST
* Content-Type: multipart/form-data
* Http response:
	+ HTTP Response status code: **200 (OK)** for successful transfer. Any http response code other than 200 is considered invalid by the app.
	+ Form Response JSON Object in the following format:

 {

 “error”: error code, (**required**)

 “message”: message (**optional**)

 }

 error\_code of 0 indicates successful transfer of the file.

 Sample of web service implementation in .NET (c#):

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* BASIC AUTHENTICATION \*/

 if (ctx.Request.HttpMethod == "OPTIONS")

 {

 ctx.Response.ContentType = "text";

 ctx.Response.ContentEncoding = Encoding.UTF8;

 ctx.Response.AppendHeader("Content-Length", "0");

 ctx.Response.AppendHeader("Connection", "keep-alive");

 ctx.Response.AppendHeader("Access-Control-Allow-Origin", "\*");

 ctx.Response.AppendHeader("Access-Control-Allow-Headers", "Authorization, Origin, content-type, accept");

 ctx.Response.AppendHeader("Access-Control-Allow-Credentials", "true");

 ctx.Response.AppendHeader("Access-Control-Allow-Methods", "OPTIONS, GET, POST");

 return;

 }

 if (ctx.Request.Headers["Authorization"] == null)

 {

 ctx.Response.StatusCode = 401;

 ctx.Response.AppendHeader("WWW-Authenticate", "Basic");

 return;

 }

 //validate authentication

 if (!Authenticate.ValidateRequest(ctx.Request.Headers["Authorization"].ToString()))

 {

 ctx.Response.StatusCode = 401;

 return;

 }

 //save file

 if (ctx.Request.HttpMethod.ToUpper() == "POST")

 {

 try

 {

 if (ctx.Request.Files.Count > 0)

 {

 HttpPostedFile myFile = ctx.Request.Files[0];

 if (myFile != null && myFile.ContentLength > 0)

 {

 Stream fileContents = myFile.InputStream;

 string fileName = Path.GetFileName(myFile.FileName);

 string path = Path.Combine(ctx.Server.MapPath("~/UploadedData"), fileName);

 myFile.SaveAs(path);

 ctx.Response.ContentType = "application/json";

 ctx.Response.Write("{\"error\":\"0\"}");

 return;

 }

 }

 else

 {

 ctx.Response.StatusCode = 1;

 return;

 }

 }

 catch (Exception e)

 {

 ctx.Response.StatusCode = 1;

 return;

 }

 }