# APPENDIX B

# GC-ONLINE DOCUMENTATION

## Overview

GCOnline is a proprietary web-based software application developed by Egan Jackson Lohman for the specific purpose of quantifying and analyzing lipid compounds, either as free fatty acid (FFA), monoacylglyceride (MAG), diacylglyceride (DAG), triacylglyceride (TAG) or fatty acid methyl ester (FAME), after separation and detection via GC-FID or GC-MS. The following provides general usage, functionality and documentation for the application.

## Software Beta

### Technology

GCOnline was written using the following coding languages and paradigms:

1. ASP.NET 4.0 Framework
2. C# Object Oriented Coding language
3. Model, View, Controller (MVC) 4.0 coding paradigm
4. Entity Framework 5.0 Model First coding paradigm
5. SQL Server 2010
6. HTML 4
7. jQuery

### Source Code

The source code for GCOnline has been made public and is hosted under Google Code Project. GCOnline is licensed under the MIT License. Code can be downloaded via Subversion Repository source control software. We recommend using TortoiseSVN Client.

1. Code URL: <http://code.google.com/p/gc-online/>
2. SVN: https://gc-online.googlecode.com/svn/trunk/
3. The MIT License (MIT)

Copyright (c) 2013 – Montana State University

Permission is hereby granted, free of charge, to any person obtaining a copy  
of this software and associated documentation files (the "Software"), to deal  
in the Software without restriction, including without limitation the rights  
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell  
copies of the Software, and to permit persons to whom the Software is  
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in  
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## Workflow

Once you have downloaded and compiled the source code using Microsoft’s Visual Studio 2010 Express or later, you will need to configure IIS to run the application as a website. Configuring IIS is outside the scope of this documentation, please refer to online materials or consult your IT administrator.

GCOnline requires data collected from a GC to be imported into the application as a Microsoft Excel workbook file (.xslx). The organization of these workbooks and subsequent worksheets is critical for the application to function properly.

Each workbook should contain only worksheets containing GC data. These data can be either sample data or calibration data, but no other worksheets should exist in the workbook. Each spreadsheet should correspond to one and only one Run of a Sequence. A workbook can contain as many Run worksheets as desired and they do not have to correspond to the actual runs per sequence from the GC. In other words, you can mix and match runs from any given sequence; however for organizational purposes this is not recommended. There is an example workbook included in the root directory of the source code repository.

Each worksheet, which corresponds to one Run, must have the following columns with the headers spelled and capitalized precisely.

1. SequenceName
2. RunName
3. ExperimentDate
4. Dilution
5. CDW
6. SampleVolume
7. CDWVolume
8. BMCmmol
9. CellCount
10. Time
11. Area
12. Height
13. Width
14. Area%
15. Symmetry

Columns 1-9 correspond to your specific sample information, all of these fields are required, but you can set them to 1 if you don’t require a specific calculation: The header column must have the exact text string of the column name (ex. SequenceName), the cell directly below the header row holds the actual value.

1. SequenceName – ex. 20130906\_Calibration, This is the name of the sequence, you can name this whatever you want, but it needs to be the same on each worksheet (type = string)
2. RunName – ex. Run\_1A (Name of run) (type = string)
3. ExperimentDate – ex. 20130906 (type = DateTime)
4. Dilution – ex. 3 (unit = mL, type = integer, double or float) This is the dilution, if any you used
5. CDW – ex. 1.23 (unit = mg/mL, type = integer, double or float) This is the overall cell dry weight of the CULTURE. This is not the mass you weighed out from which you extracted. This is multipled by the SampleVolume to derive the actual sample weight.
6. SampleVolume – ex. 45 (unit = mL, type = integer, double or float) This is the sample volume you actually extracted from. This is multiplied by CDW to determine the concentration (mg/mL) of extracted biomass.
7. CDWVolume – ex. 45 (unit = mL, type = integer, double or float) This is the overall culture volume you measured to determine your CDW.
8. BMCmmol ex 0.1 (unit = C mmol, type = integer, double or float) If you don’t require a Carbon ratio you do not need to specify this. Just enter 1.
9. CellCount – 1e7 (unit = cells/mL, type = integer, double or float) This is used to calculate lipid per 1000 cells. You can set this to 1 if not required.

Columns 10-15 correspond to data obtained from the GC. All of these fields are required:

1. Time – Retention time for individual peak
2. Area – Response area for individual peak
3. Height – Height of individual peak
4. Width – Width of individual peak
5. Area% – Percent area of individual peak
6. Symmetry – General symmetry rating

Again, each worksheet should contain columns for the six fields generated from the GC software (*e.g.* Agilent Chemstation). A header row should contain the name of the field, spelled and capitalized precisely as written above. Each row below the header row should contain the corresponding data for each peak in the chromatogram. There is no limit on how many peaks per run can be analyzed.

### Sequence Management

The home page of GCOnline is the sequence management section. Here you can upload new workbooks, delete existing sequences, or view existing sequences.

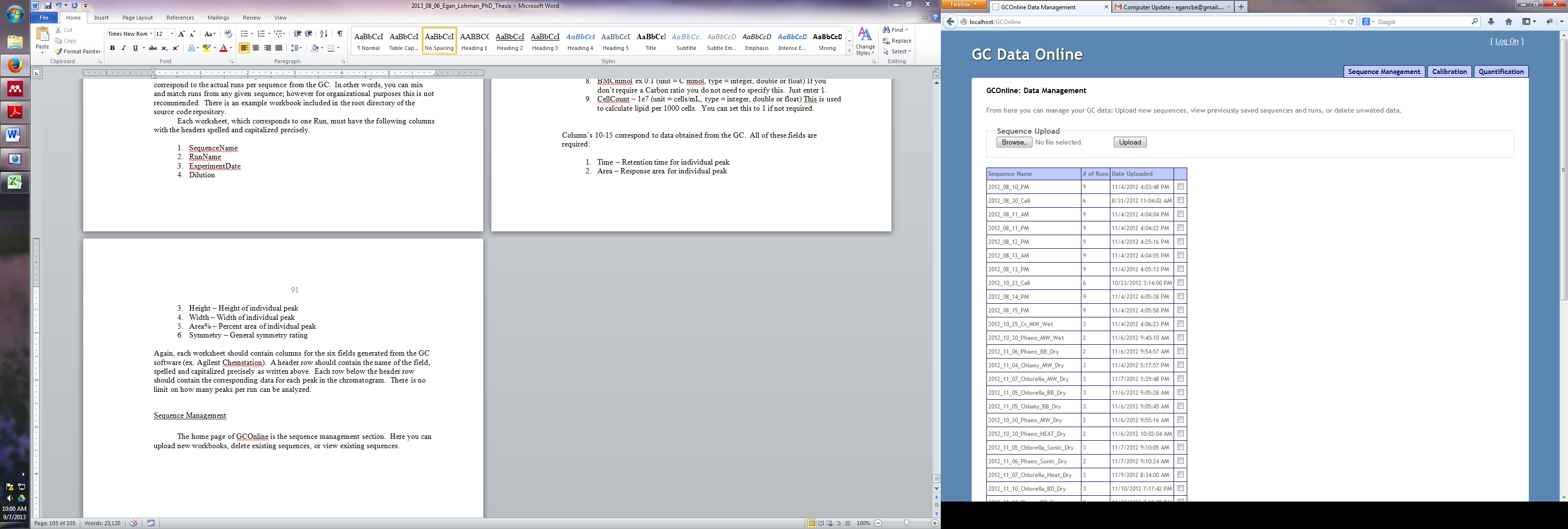


Figure B.1: GCOnline Sequence Management Screenshot.

Your first order of business should be to upload a workbook containing a sequence and subsequent runs for a calibration. Use the browse button to locate the workbook on your local or networked hard drive, and upload it to the database. Once the sequence has uploaded, it will appear in the list below the upload button. You can then check the box next to the sequence and use the navigation at the bottom to View Runs, Get Details on the Sequence, or Delete the Sequence.

Check the box next to your calibration sequence, and then click View Runs.

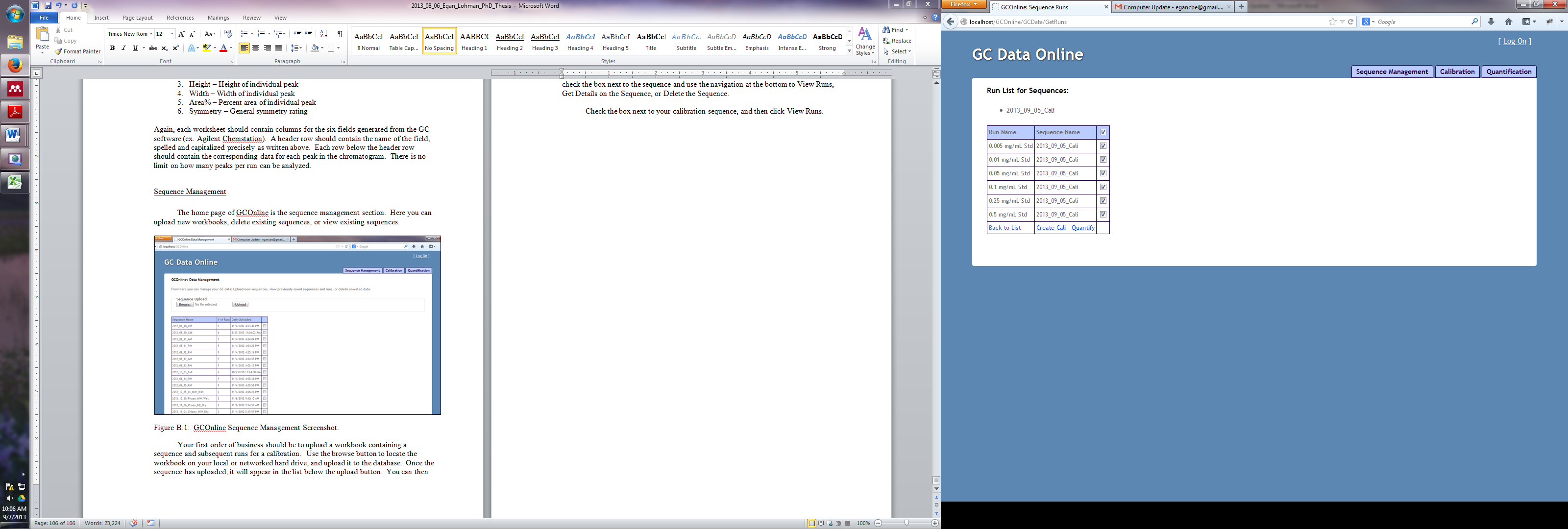


Figure B.2: GCOnline View Runs Screenshot.

From the View Runs screen, you can choose the runs you wish to analyze by checking the box corresponding to each run. The checkbox at the top selects or deselects all runs. You can then either click Create Cali, or Quantify. Create Cali will allow you to define which standard each peak in each run corresponds to, and then generates a calibration curve using Microsoft’s LINEST function. This is the same code used in Microsoft Excel’s LINEST function for linear regression fit.

Select all runs in the sequence and then click Create Cali. A new table will appear assigning each run to a standard concentration. The algorithm will try to identify these concentrations from the name of the Run, so it is best practice to include the concentration in the Run Name (see Figure B.2). If the algorithm makes a mistake, simply change the concentration value for each run in the Concentration (mg/mL) column.

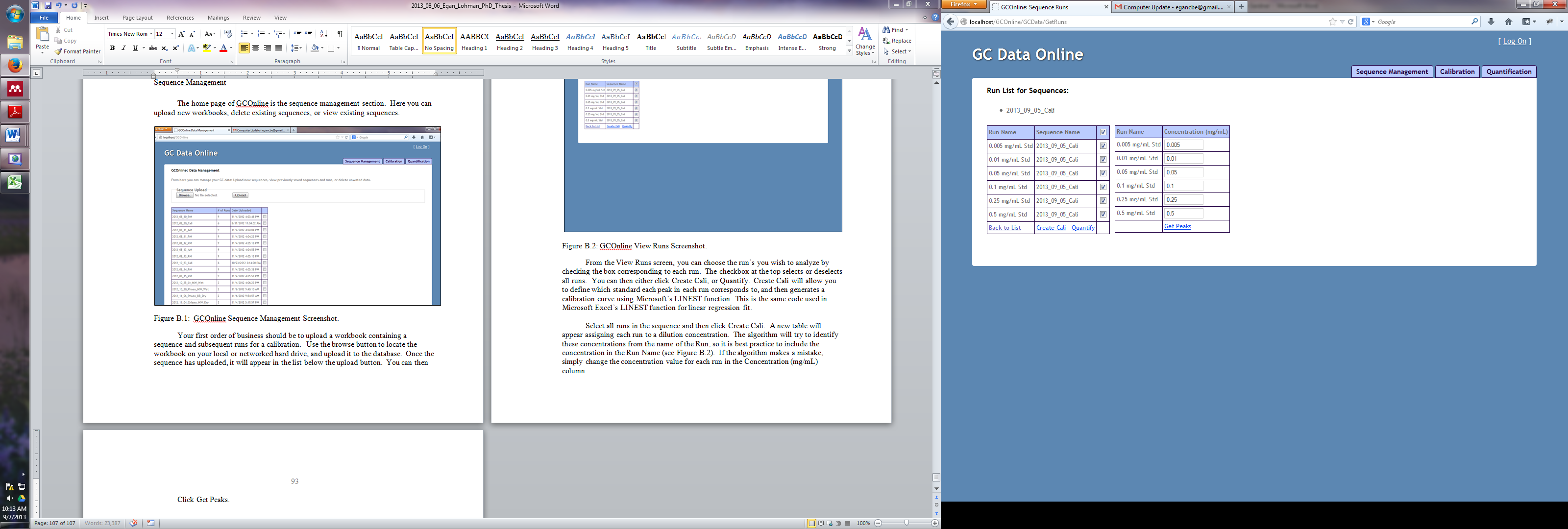


Figure B.3: GCOnline Create Calibration Screenshot.

Clicking Get Peaks, will generate individual charts for each calibration run. From here you can specify which peak correlates to each standard in your calibration set. By default the standards are hardcoded into the software and currently use a Biodiesel standard mix containing C10:0, C12:0, C14:0, C16:0, C18:0, and C20:0 FFAs; C12:0, C14:0, C16:0, and C18:0 MAGs; C12:0, C14:0, C16:0, and C18:0 DAGs; along with C11:0, C12:0, C14:0, C16:0, C17:0, C18:0, C20:0 and C22:0 TAGs (Sigma-Aldrich, St. Louis MO). If you need to use different standards you will have to modify the code (See Files, Classes, Methods and Functions below).

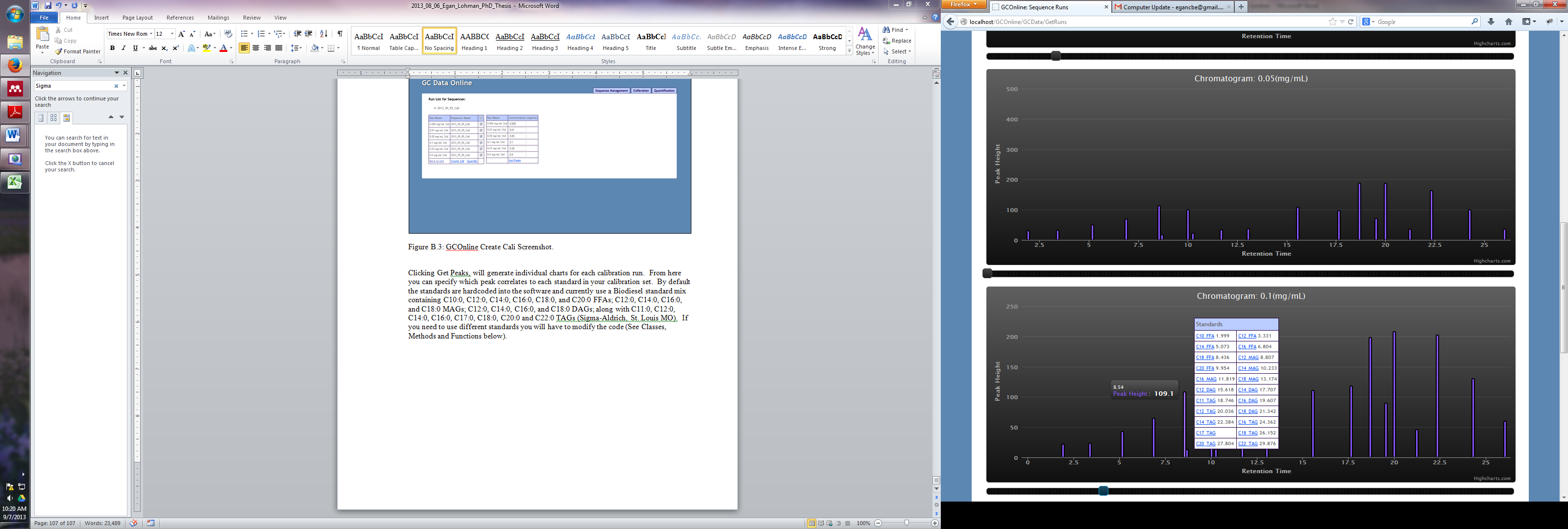


Figure B.4 GCOnline Chromatogram Chart Screenshot

Methodically assign each peak to its corresponding standard for each calibration chromatogram by clicking the peak in the graph and assigning a standard from the popup window. Once you have assigned all peaks/run, you will have a complete table at the top of the screen assigning each standard a value per concentration range you used for your calibration (*e.g.* 0.005 -0.5 mg/mL).

Click the Generate Cali Curve link under the calibration table. GCOnline will use the assignments and the LINEST function to generate a linear regression best fit to your data. This will appear as a table. NOTE: This may take a few minutes and it will appear that nothing is happening. Be patient, your regression fit will appear shortly.

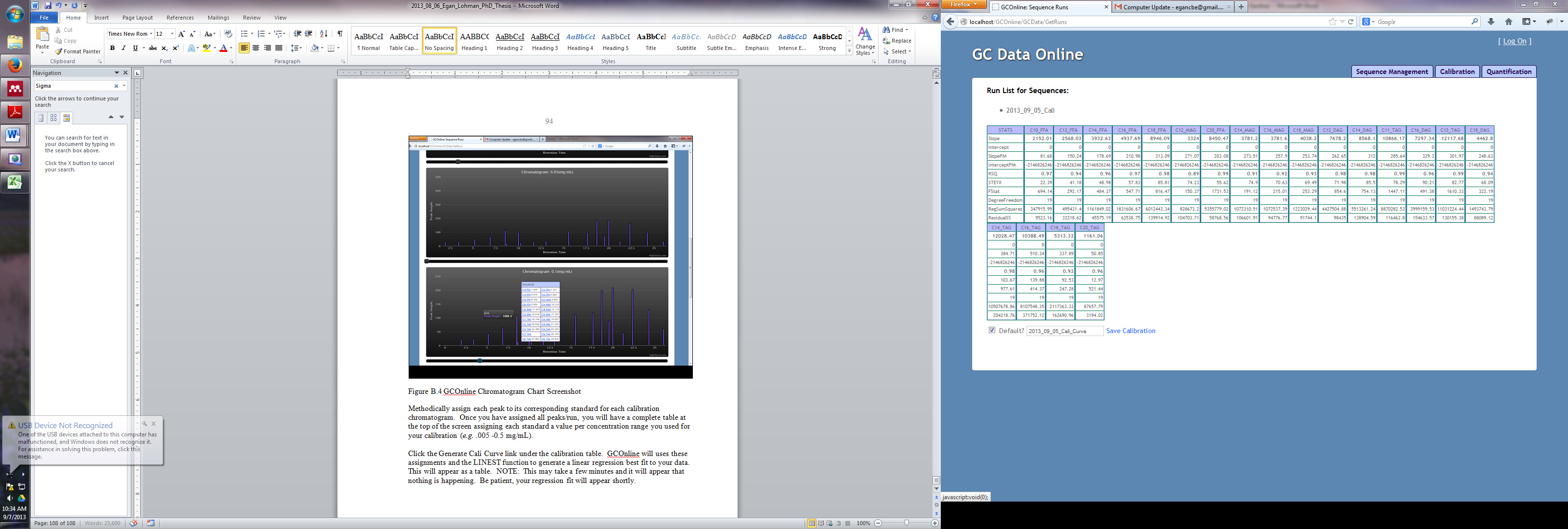


Figure B.5: GCOnline Generate Calibration Curve Screenshot

This table will contain the following for each standard:

1. Slope
2. Intercept
3. SlopePM
4. InterceptPM
5. RSQ
6. STEYX
7. FStat
8. DegreeFreedom
9. RegSumSquares
10. ResidualSS

These are the typical statistical values generated when running a linear regression model. If you are satisfied with the results, Enter a name for your calibration and click save. NOTE, you can also check the Default checkbox (selected by default) which assigns this calibration curve as the default calibration to use by the software. Click Save Calibration.

### Analyzing and Quantifying Samples

Once you have saved a calibration curve to the database, you can analyze a sample sequence. Follow the steps outlined above to upload a sample sequence workbook and then from the Sequence Manager click View Runs. As outlined above, you can then select the runs you wish to quantify by clicking the Quantify link under the run table.

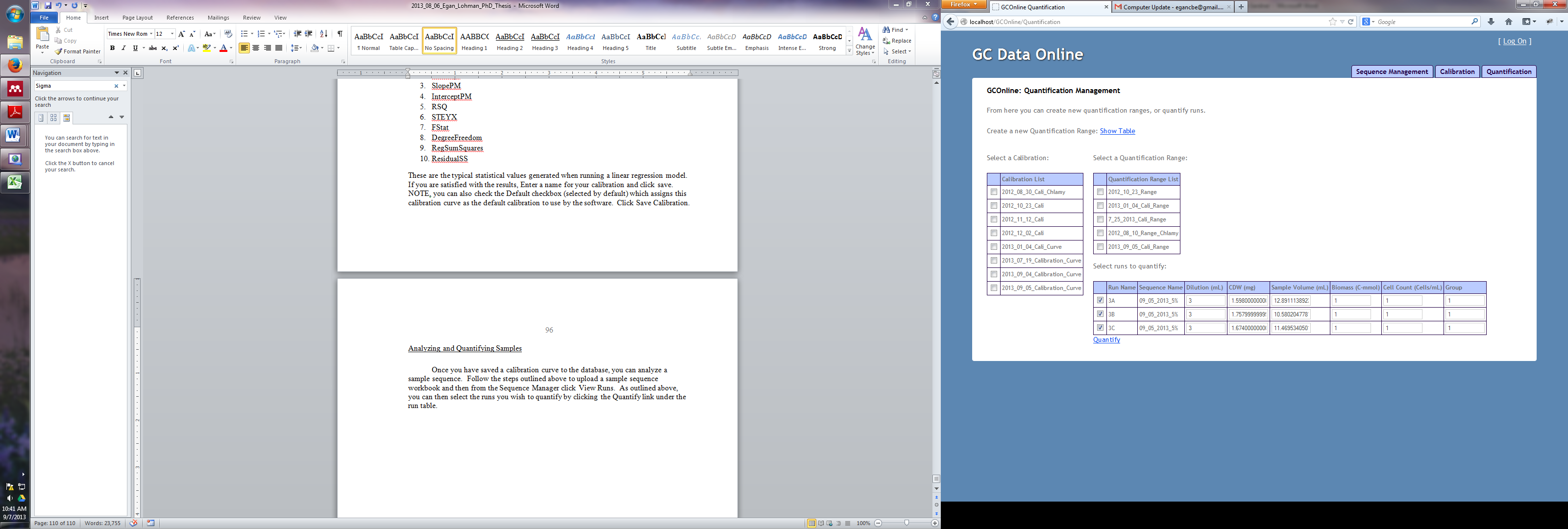


Figure B.6: GCOnline Quantification Management

You will now see a screen with two or three tables. The first table is a list of calibration curves, the second is a list of quantification ranges, and the third is a list of sample runs to be quantified. If this is your first time using the application, then you will only have the calibration list table, and the run list table. You will need to generate a quantification range before quantifying your samples. You can do this by clicking Show Table at the top of this screen (see Figure B.6). This will generate an empty table with each standard as a row. You will need to assign a retention time range which GCOnline will use to determine which standard calibration slope to use for quantifying each peak in your sample data. These time ranges should not overlap. To avoid overlap, but also avoid gaps, simply add 001 to the start of the next range. For example: If you wish to quantify all peaks for C10 FFA as any peak that falls between 2.45 – 3.65 minutes, then the range for C12 FFA should start at 3.65001. Assign a Calibration Range Name and click Create. Your new range will automatically show up in the Quantification Range table and be selected by default. You will get an error if you try to analyze a sequence without selecting both a Calibration and Quantification Range.

The Run List table has many columns, most of which have been filled out from the data you provided in the original worksheet. The final column Group is used to assign replicate samples for averaging and standard deviation. Assign the same group to samples you wish to be averaged. Click Quantify. GCOnline will attempt to analyze and quantify your data, and create an Excel workbook containing the results. The location the application uses to save this workbook to is hardcoded in the code. You will want to change this location to something suitable for your method of organizing files on your local hard drive. Please see the following sections for modifying the code.

## Files, Classes, Methods and Functions

This section details the most pertinent files, Classes, methods and functions used by GCOnline. To begin, GCOnline is written using the Model, View, Controller programming paradigm. An in-depth discussion of MVC is outside the scope of this documentation; please refer to online material for more information. Essentially however, the MVC paradigm separates tasks into three categories. All user interface functionality, (*i.e.* HTML, form fields, client side scripts, etc..) are encompassed in the View. All business logic is encompassed in the Model, and the Controller is responsible for aligning a View with its respective Model.

### Views

1. Calibration – Includes all views used for creating, modifying and deleting calibrations.
2. GCData – Includes all views used for uploading, saving, modifying and deleting GC data.
3. Quantification – Includes all views used for quantification

### Scripts

1. GCOnline relies heavily on the javascript library jQuery. <http://www.jquery.com>
2. Common.js – This javascript file contains all the frontend scripting code used to manipulate the document object model (DOM), send and receive data via asynchronous javascript (AJAX), and store relevant object models as javascript object notation (JSON).

### Models

1. There is only one model used for GCOnline. DataModel.cs contains all the properties, Classes and relevant Entity Framework database context needed for the application.

### Controllers

1. CalibrationController.cs – responsible for connecting Calibration views with the DataModel.
2. GCDataController.cs – responsible for connecting GCData views with the DataModel.
3. QuantificationController.cs – responsible for connecting Quantification views with the DataModel.

All classes, methods and functions are documented in the code itself. Additional information is provided by Microsoft Visual Studio Intellisense for usage information.

### Customizing the Application

It may be desireable to modify GCOnline for various purposes. Two such instances immediately come to mind and are outlined below. 1) You may wish to use different standards for your calibration and quantification. 2) You may wish to save quantification data to a different location in your local hard drive. Other modifications are outside the scope of this documentation.

### Modifying the Default Standards

You will need to modify the code in three places:

1. Scripts/Common.js – line 13. Here you will find an empty JSON object (this.standardsArray) containing a name, value pair for each standard. Modify this at will.
2. Quantification/Index.cshtml – line 6. Here you will find a string array (standardsArray) containing a list of standards. Modify this at will. NOTE: Your naming convention must match with the standardsArray in common.js.
3. Models/ModelData.cs – line 168. Here you will find a Dictionary called lookup which contains a name value pair for each standard. Modify this at will. NOTE: Your naming convention must match with stardardsArray in common.js.

Modifying the Local Hard Drive Location

1. Models/DataModel.cs – line 20. Here you will find a instance of ExcelQueryFactory which accepts 1 parameter, which is a string specifying the location to save the output Excel workbook. Modify this at will.