THE DEVELOPMENT OF THE WEB BASED CO2SYS PROGRAM

Shigui Yuan

The University of Montana

Follow this and additional works at: https://scholarworks.umt.edu/etd

Let us know how access to this document benefits you.

Recommended Citation

This Professional Paper is brought to you for free and open access by the Graduate School at ScholarWorks at University of Montana. It has been accepted for inclusion in Graduate Student Theses, Dissertations, & Professional Papers by an authorized administrator of ScholarWorks at University of Montana. For more information, please contact scholarworks@mso.umt.edu.
THE DEVELOPMENT OF THE WEB BASED CO2SYS PROGRAM

By

Shigui Yuan

B.S. Chemistry, Hohai University, Nanjing, P.R. China, 1993
M.S. Chemistry, Hohai University, Nanjing, P.R. China, 2000
Ph.D. Chemistry, The University of Montana, Missoula, MT, 2006

Professional Paper

Presented in partial fulfillment of the requirements
for the degree of

Master of Science
in Computer Science

The University of Montana
Missoula, MT

Autumn 2006

Approved by:

Dr. David A. Strobel, Dean
Graduate School

Dr. Yolanda Reimer
Computer Science

Dr. Jesse Johnson
Computer Science

Dr. Michael DeGrandpre
Chemistry
The Development of The Web Based CO2SYS Program

Chairperson: Yolanda Reimer

A web-based version of CO2SYS program has been implemented to replace the current DOS based version system. The user does not have to download anything to a local computer, instead they can run the calculations online freely. For this new designed program, all the user inputs and options are displayed in one single window instead of several small black and white DOS screens. All the calculation results are listed in a single page, as well. The user can change any inputs and constants before and after the data calculation, i.e., recalculation. Much more powerful error checking has been built into this web-based system. It also provides useful directions and guidance for the user. The user can get access to the helpful information for each input and constant. Typographical error information, which is listed separately from their individual reference paper, is incorporated with the reference through the hyperlinks. Moreover, this new system presents an attractive and dynamic appearance to users.
Acknowledgements

First I want to give thanks to my advisor Dr. Yolanda Reimer for her relentless support and guidance during my graduate study. Yolanda has spent a lot of her precious time helping me with this project. My committee members Dr. Michael DeGrandpre and Dr. Jesse Johnson have consistently provided valuable advice and guidance for this study. Michael recruited me from China six years ago and gave me a chance to fulfill my dream to get a Ph.D. degree in Chemistry. All the faculty, staff, and graduate students at the department of computer science gave me a warm environment to study. I really appreciate their help.

I could not have made it to this stage in my life without all the love and support from my family back in China. I also want to thank my host family Mr. and Mrs. Risho for their caring hearts. I also enjoyed the time I spent with all my friends I have met in this beautiful town.
Table of Contents

Abstract ........................................................................................................... ii
Acknowledgement ........................................................................................... iii
List of Tables .................................................................................................... vi
List of Figures ................................................................................................... v

Chapter 1 Introduction .................................................................................... 1
1.1 Overview ................................................................................................... 1
1.2 Background ............................................................................................... 1
1.3 CO2SYS program information ................................................................. 2
   1.3.1 Input CO₂ parameters ....................................................................... 2
      1.3.1a pH ............................................................................................... 3
      1.3.1b Alkalinity ..................................................................................... 3
      1.3.1c TCO₂ ........................................................................................... 3
      1.3.1d fCO₂ and pCO₂ ........................................................................... 4
   1.3.2 Single input mode and batch input mode ........................................... 4
      1.3.2a Single input mode ....................................................................... 4
      1.3.2b Batch input mode ....................................................................... 5
1.4 The project objectives ............................................................................. 6
   1.4.1 Advantages of DOS based CO2SYS program ................................. 6
   1.4.2 Limitations of DOS based CO2SYS program ................................. 7
   1.4.3 The objectives of this project ........................................................... 8
1.5 Programming tools ................................................................................ 10
1.6 Summary .................................................................................................. 11

Chapter 2 Preliminary system design .............................................................. 12
2.1 Overview .................................................................................................. 12
2.2 Potential users of the system .................................................................. 12
2.3 Functional requirements ......................................................................... 13
   2.3.1 Freshwater or seawater sample under the single input mode ........... 13
   2.3.2 Freshwater or seawater samples under the batch mode ................. 15
   2.3.3 Other functionalities for the system ................................................. 17
2.4 Preliminary system design ..................................................................... 17
   2.4.1 Freshwater under the single mode ................................................... 18
   2.4.2 Freshwater under the batch mode ................................................... 22
   2.4.3 Seawater under the single mode ...................................................... 23
   2.4.4 Seawater under the batch mode ...................................................... 26
2.5 Summary .................................................................................................. 26

Chapter 3 Cognitive walkthrough and heuristic evaluation ......................... 27
3.1 Overview .................................................................................................. 27
3.2 Informal system testing without real users ............................................ 27
3.3 Cognitive walkthrough ......................................................................... 28
3.4 Heuristic evaluation ............................................................................... 30
3.5 Summary ................................................................................................. 33

Chapter 4 User testing .................................................................................. 35
4.1 Overview .................................................................................................. 35
4.2 User testing participants ........................................................................ 36
4.3 Ethical issues while working with users……………………………………..36
4.4 The system evaluated by users………………………………………………38
4.5 Select the tasks for user testing……………………………………………51
4.6 Problems found from the user testing……………………………………..52
4.7 Results from the post-test questionnaire…………………………………..54
4.8 Results from the post-test interview………………………………………..54
4.9 Changes to the system following the user testing………………………...57
4.10 Summary……………………………………………………………………66

Chapter 5  Conclusion and future work……………………………………………67
  5.1 Project conclusion……………………………………………………………67
  5.2 Future work…………………………………………………………………67

Reference……………………………………………………………………………...69
Appendix A  Cognitive walkthrough…………………………………………………70
Appendix B  Heuristic evaluation……………………………………………………105
Appendix C  User testing……………………………………………………………116
List of Tables

Table 4.1 The demographic information of all users for testing sessions……………….37
Table 4.2 All the information from the user post-test questionnaire…………………….55
Table 4.3 All the information from the user post-test interview…………………………56
Table C1 The data capture form for test scenario one……………………………….…130
Table C2 The data capture form for test scenario two………………………………….131
Table C3 The data capture form for test scenario three………………………………...132
Table C4 The data capture form for test scenario four…………………………………133
Table C5 The data capture form for test scenario five………………………………….134
Table C6 Summarized is completeness of the user testing………………………….….137
# List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>The calculation result display screen for single input mode</td>
<td>5</td>
</tr>
<tr>
<td>1.2</td>
<td>The calculation result display screen for batch input mode</td>
<td>7</td>
</tr>
<tr>
<td>2.1</td>
<td>The flow diagram for the calculations of web based CO2SYS program</td>
<td>14</td>
</tr>
<tr>
<td>2.2</td>
<td>The complete screen layout of data calculation for freshwater sample</td>
<td>19</td>
</tr>
<tr>
<td>2.3</td>
<td>The calculation result display page with the inputs and options</td>
<td>20</td>
</tr>
<tr>
<td>2.4</td>
<td>The preliminary design web page of data calculation</td>
<td>21</td>
</tr>
<tr>
<td>2.5</td>
<td>The complete page layout of data calculation for seawater sample</td>
<td>23</td>
</tr>
<tr>
<td>2.6</td>
<td>The calculation result display page with the inputs and options</td>
<td>24</td>
</tr>
<tr>
<td>2.7</td>
<td>The preliminary design web page of data calculation</td>
<td>25</td>
</tr>
<tr>
<td>3.1</td>
<td>The complete screen layout of data calculation for freshwater sample</td>
<td>31</td>
</tr>
<tr>
<td>3.2</td>
<td>The web page layout of data calculations for freshwater sample</td>
<td>34</td>
</tr>
<tr>
<td>4.1</td>
<td>The complete revised web page layout of data calculation</td>
<td>39</td>
</tr>
<tr>
<td>4.2</td>
<td>The revised web page layout for freshwater sample</td>
<td>40</td>
</tr>
<tr>
<td>4.3</td>
<td>The revised web page layout of data calculation for freshwater sample</td>
<td>42</td>
</tr>
<tr>
<td>4.4</td>
<td>A pop-up message as feedback after the program finishes the calculation</td>
<td>43</td>
</tr>
<tr>
<td>4.5</td>
<td>The output file data display page of data calculation</td>
<td>44</td>
</tr>
<tr>
<td>4.6</td>
<td>An error message from the system after an invalid input</td>
<td>45</td>
</tr>
<tr>
<td>4.7</td>
<td>A pop-up message with detailed information of the input for “salinity”</td>
<td>46</td>
</tr>
<tr>
<td>4.8</td>
<td>Error message given by the program due to multiple invalid inputs</td>
<td>47</td>
</tr>
<tr>
<td>4.9</td>
<td>The introduction page for the web-based CO2SYS program</td>
<td>48</td>
</tr>
<tr>
<td>4.10</td>
<td>The reference page for the web-based CO2SYS program</td>
<td>49</td>
</tr>
<tr>
<td>4.11</td>
<td>The contact information of web-based CO2SYS program</td>
<td>50</td>
</tr>
<tr>
<td>4.12</td>
<td>The web page for “Help” of web-based CO2SYS program</td>
<td>51</td>
</tr>
<tr>
<td>4.13</td>
<td>Error message for unrealistic user input for TC and fCO2</td>
<td>58</td>
</tr>
<tr>
<td>4.14</td>
<td>A separate window containing the content of the output file data</td>
<td>59</td>
</tr>
<tr>
<td>4.15</td>
<td>Error message with further directions after the invalid input</td>
<td>61</td>
</tr>
<tr>
<td>4.16</td>
<td>The printer friendly version of the data calculation</td>
<td>62</td>
</tr>
<tr>
<td>4.17</td>
<td>The Microsoft version of the data calculation results</td>
<td>63</td>
</tr>
<tr>
<td>4.18</td>
<td>The Microsoft excel version of the data calculation results</td>
<td>64</td>
</tr>
<tr>
<td>4.19</td>
<td>One of the sample file used for the batch mode calculation</td>
<td>65</td>
</tr>
<tr>
<td>4.20</td>
<td>One of the sample file used for the batch mode calculation</td>
<td>65</td>
</tr>
<tr>
<td>4.21</td>
<td>One of the sample file used for the batch mode calculation</td>
<td>66</td>
</tr>
<tr>
<td>5.1</td>
<td>One of the sample file used for the calculation under the single mode</td>
<td>68</td>
</tr>
<tr>
<td>A1</td>
<td>The initial web page layout of data calculation for freshwater sample</td>
<td>71</td>
</tr>
<tr>
<td>A2</td>
<td>The complete web page layout of data calculation</td>
<td>73</td>
</tr>
<tr>
<td>A3</td>
<td>Data calculation results in a single web page for freshwater samples</td>
<td>76</td>
</tr>
<tr>
<td>A4</td>
<td>The complete web page layout of data calculation for freshwater</td>
<td>79</td>
</tr>
<tr>
<td>A5</td>
<td>The complete web page layout of data calculation for seawater</td>
<td>86</td>
</tr>
<tr>
<td>A6</td>
<td>The web page displaying the calculation results for seawater sample</td>
<td>95</td>
</tr>
<tr>
<td>A7</td>
<td>The web page layout of data calculation for seawater sample</td>
<td>97</td>
</tr>
<tr>
<td>B1</td>
<td>The web page layout of data calculation for freshwater sample</td>
<td>106</td>
</tr>
<tr>
<td>B2</td>
<td>The data calculation results display for freshwater</td>
<td>107</td>
</tr>
<tr>
<td>B3</td>
<td>The screen layout of data calculation for freshwater</td>
<td>109</td>
</tr>
<tr>
<td>B4</td>
<td>The web page layout of data calculation for seawater</td>
<td>111</td>
</tr>
<tr>
<td>B5</td>
<td>The data calculation results display for freshwater</td>
<td>112</td>
</tr>
</tbody>
</table>
Figure B6 The screen layout of data calculation for freshwater..........................114
Chapter 1

Introduction

1.1 Overview

This project focuses on the development of a web based version of the CO2SYS program to replace the current popular DOS based version. This chapter will first discuss the background of this project, the DOS version CO2SYS program information, the project objectives and programming tools used for the system implementation.

1.2 Background

Oceanographers always have disputes over how to calculate inorganic carbon species when they study the ocean carbonate system. Many scientists have determined the dissociation constants of carbon dioxide (CO$_2$) in the ocean, but none of these sets of constants has been widely accepted. To clarify the confusions and facilitate oceanic studies, the CO2SYS program was developed by Dr. Ernie Lewis and Dr. Doug Wallace at the Brookhaven National Laboratory and was released in 1998 (http://cdiac.esd.ornl.gov/oceans/co2rprt.html). The program is based on previously developed programs CO2SYSTM.EXE, FCO2TCO2.EXE, PHTCO2.EXE and CO2BTCH.EXE (http://cdiac.esd.ornl.gov/oceans/co2rprt.html). Marine scientists can use the CO2SYS program to calculate the relating parameters of the carbon dioxide (CO$_2$) system in freshwater and seawater. It uses two of the four measurable parameters of the CO$_2$ system (total alkalinity (TA), total inorganic CO$_2$ (TCO$_2$), pH, and either
fugacity ($f_{CO_2}$) or partial pressure ($p_{CO_2}$) of CO$_2$) to calculate the other two parameters under a set of input and output conditions (temperature and pressure) chosen by the user.

This program is written in compiled MICROSOFT QuickBASIC and runs as a DOS based .EXE file on any Windows based PC. This program is open source and can be downloaded from http://cdiac.esd.ornl.gov/oceans/co2rprt.html to the local computer. It can run in either a single-input mode or a batch-input mode. A large variety of options for various constants and parameters are also provided for the calculations. The reference information for each parameter and constant is available on the screens, as well. The main goal of this project is to develop a web-based version of the CO2SYS program to replace the current DOS-based one. With the new version of the program, users only need to visit the website for data calculations without downloading any software to a local computer. It saves resources on the local computer and avoids the memory overload during data processing.

1.3 The CO2SYS program information

1.3.1 Input CO$_2$ parameters

Alkalinity and TCO$_2$ are characteristics of the aqueous solution and do not depend on temperature and pressure. $f_{CO_2}$, $p_{CO_2}$ and pH are dependent on temperature and pressure. Any two of these parameters (excluding the combination of $f_{CO_2}$ and $p_{CO_2}$), along with the other constituents of water such as salinity, enable the users to define the other two parameters under a certain temperature and pressure.
1.3.1a pH

pH is a measure of the acidity and alkalinity of a solution, numerically equal to 7 for neutral solutions, increasing with alkalinity and decreasing with acidity. The pH values normally range from 0 to 14. It is calculated with the formula below:

\[
pH = - \log [\alpha_{\text{H}^+}] \tag{1.1}
\]

\([\alpha_{\text{H}^+}]\) is the activity of hydrogen ions in the aqueous solution. pH is a master variable affecting the equilibrium and kinetics of a wide range of chemical processes in the water.

1.3.1b Alkalinity

Alkalinity is a measure of the buffer capacity of water, or the capacity of water to neutralize acids (Pankow, 1991). Measuring alkalinity is important in determining a stream’s ability to neutralize acidic pollution from rainfall or wastewater. Alkalinity does not refer to pH, but instead refers to the ability of water to resist changes in pH. The presence of buffering materials helps neutralize acids as they are added to the water. These buffering materials are primarily bases: bicarbonate (\(\text{HCO}_3^-\)), carbonate (\(\text{CO}_3^{2-}\)), hydroxide (\(\text{OH}^-\)), borate, silicate, phosphate, ammonium, sulfides and organic ligands.

1.3.1c TCO₂

TCO₂ is the total concentration of all carbon dioxide-related species dissolved in the aqueous phase (Pankow, 1991). It can be calculated from the formula below:

\[
\text{TCO}_2 = [\text{CO}_2] + [\text{H}_2\text{CO}_3] + [\text{HCO}_3^-] + [\text{CO}_3^{2-}] \tag{1.2}
\]

where \([\text{CO}_2]\) is the carbon dioxide dissolved in the water and some of it reacts with water molecules to form \([\text{H}_2\text{CO}_3]\). \([\text{H}_2\text{CO}_3]\) can dissociate into \([\text{HCO}_3^-]\) first, and then into...
[CO$_3^{2-}$]. There are equilibria between these species in the aqueous condition. Temperature and pressure variations can affect the equilibria, as well.

1.3.1d  $f_{\text{CO}_2}$ and $p_{\text{CO}_2}$

$f_{\text{CO}_2}$ in water is defined to be the fugacity of CO$_2$ in air which is in equilibrium with the water (Pankow, 1991). $p_{\text{CO}_2}$ is defined to be the product of the mole fraction of CO$_2$ in wet air (100% water-saturated) and the total pressure. Both $f_{\text{CO}_2}$ and $p_{\text{CO}_2}$ are proportional to the concentration of dissolved CO$_2$ ([CO$_2$]). The fugacity is about 0.3% to 0.4% lower than the partial pressure over the range of interest because of the non-ideality of CO$_2$.

1.3.2 Single input mode and batch input mode

1.3.2a Single input mode

For the single input mode, the user is first asked to choose the values of the constants, the salinity, the concentration of the phosphate and the silicate for seawater. Then the user can enter the input and output conditions such as temperature and pressure, and two of the known CO$_2$ system parameters, which may be those obtained from the measurements conducted in the laboratory. Lastly, the program takes all these inputs and calculates the other two CO$_2$ system parameters.

Then TA and TCO$_2$, independent of the temperature and pressure, are used to calculate pH and $f_{\text{CO}_2}$ or $p_{\text{CO}_2}$ at the output conditions. For both the input conditions and output conditions, the calculation results will include (Figure 1.1):

a.) The sensitivity of the output results to small variations of the inputs
b.) The composition of the total alkalinity and carbon speciation

c.) \( f_{CO_2} \) and \( pCO_2 \) in fully saturated (wet) air

d.) The molar fraction of \( CO_2 \) in dry air (\( xCO_2 \)) with 1 atm total pressure

e.) The degree of saturation (Omega) for calcite and aragonite

f.) The revelle factor (homogeneous buffer factor)

\[ pH \] values on four different \( \text{pH} \) scales (Total, Seawater, Free, NBS)

h.) The values of \( pK_1 \), \( pK_2 \), \( pK_W \), and \( pK_B \) on the chosen \( \text{pH} \) scale

---

**Figure 1.1** The calculation result display screen for single input mode with seawater samples from the DOS based CO2SYS program. The input parameters (TA and TCO\(_2\)) and other constituents (salinity, phosphate and silicate) are above the dashed line on the top of the screen. The output parameters (\( pH \) on the total \( \text{pH} \) scale and \( fCO_2 \)) at both input conditions and output conditions are shown with the sensitivities of output results to small variations of the input parameters and conditions.

1.3.2b Batch input mode

For the batch mode, the program takes a large set of data for the input after the user chooses the constants and input parameter types (i.e., TA and TCO\(_2\), etc). This data set, maybe from field studies or cruise data, is stored in a file created in Microsoft Excel.
or other spreadsheet programs. Data are read from the input file and the calculation results are printed into an output file.

The user is able to input the name of the output file, the number of header lines and ID fields before the data, a numeric number representing the missing value, an option to mark the missing values with a flag and additional information the user wants to record into the output file (e.g., the source of the input data).

The top of the output file includes the input file name and date, the choices and pH scale, the other extra information for the output file and the label fields for the data results (i.e., TA, TCO₂, etc). Following are lines of comma-separated calculation results for each sample data input. In addition to the data from the input, each line also contains the following (Figure 1.2):

a.) The other two CO₂ system parameters at the input conditions

b.) pH and \( f_{\text{CO}_2} \) (or \( p_{\text{CO}_2} \)) at the output conditions

c.) Omega for calcite and aragonite, \([\text{HCO}_3^-]\) and \([\text{CO}_3^{2-}]\) both at the input conditions and at the output conditions

d.) If a flag option is chosen for missing value, an extra field will be added at the end of the each line. Its value will be the designated missing value if there is at least one missing value for the data input, zero if there is none

1.4 The project objectives

1.4.1 Advantages of the DOS based CO2SYS program

Since the DOS based CO2SYS program has been released to the marine chemistry world, it has received very good reviews. Its popularity is due to the fact that it:
1) Runs under DOS on any windows based PC
2) Runs fast and stand alone
3) Has simple key stroke operation
4) Provides large variety of choices for constants
5) Can be run under the single input mode and batch input mode
6) Has detailed information for each constant and reference on the screen
7) Is robust and accurate

**Figure 1.2** The calculation result display screen for batch input mode with seawater samples from the DOS based CO2SYS program. The top of the screen includes the input data file, output data file, date of execution, constants and pH scale used for calculation. Next the additional information for the output data file is listed. For each data line, the ID field, salinity (Sal), phosphate (TP), silicate (TSi), input temperature (Tinp), input pressure (Pinp), output temperature (Tout), output pressure (Pout), TA, TCO$_2$, etc, are displayed.

### 1.4.2 Limitations of the DOS based CO2SYS program

With the ever increasing demands in oceanographic research, the DOS based CO2SYS program has shown some weaknesses, which are:

1) The DOS window is too small for data display
2) Sometimes changes can be made only by restarting the program. If the user enters a negative value for TA by accident, the program forces the user to restart the program before he can change the input.

3) Limited freedom to move between different functionalities of the program. For example, the user has to finish one execution before he can switch from the single mode to the batch mode, or from the batch mode to the single mode.

4) Sometimes tedious to run through unnecessary steps. If the user only wants to change the input temperature, he has to run through the constants selection screen and input parameter selection again before he is able to enter different input values.

5) Too many window screens needed for one single calculation. There are three screens for inputs and four screens for output result display. If default values for input mode or constants, etc need to be changed, at most another five screens will be required to make the changes.

6) Not enough error checking for user data inputs. For example, negative values are allowed for the input of temperature, phosphate, etc.

1.4.3 Objectives of this project

The proposed web based program will include all the basic functionalities and keep all the calculation methods of the existing DOS based program. All the input values and choices will be displayed on one page instead of several screens in the DOS version. The calculation results will be on another page if it is running in the single mode. The
new interface will be more spacious and attractive than the current black and white screen. All the inputs and choices can be changed at any time before calculations are executed.

More strict and powerful error checking for user input will be built into the new system. For example, the user cannot enter a negative number, a string, or a number outside the valid range (0 - 40 °C) for the input temperature. The program will give an error message and provide helpful hints for valid inputs. All the relevant information for each input and option will be provided nearby through hyperlink. The users can check this information when they try to undertake an action. For the DOS version program, all the program information is included in a separate document (http://cdiac.ornl.gov/oceans/co2rprt.html). In the new version, the program information will be presented alongside with the calculation program through several individual components such as “Introduction” and “Help”, etc. Especially, the typographical error information of the papers referenced by the CO2SYS program will be put together with the corresponding reference papers, which is separated in the program document for the DOS version. It is very convenient for the user to find the program information with the new version.

The following chapters will discuss the different stages of the development of this project: preliminary system design (Chapter 2), cognitive walkthrough and heuristic evaluation, i.e. system testing without real users (Chapter 3), user testing (Chapter 4), and project conclusion and future work (Chapter 5).
To develop this new version of the CO2SYS program, the PHP (Personal Home Page) programming language used on the server side. JavaScript is used for the client side programming. Macromedia Dreamweaver MX 2004 is the tool used to build websites and applications. Apache is the server where all PHP programming code is processed.

PHP, is the server side scripting language designed specifically for the web (http://www.php.net). It is an open source product and can be accessed, altered, and redistributed for free. Compared to other competitors such as Perl, ASP, JSP and Allaire Cold Fusion, PHP has several advantages: high performance, interfaces to many different database systems, built-in libraries for many common web tasks, low cost, easy to learn and use, portability and availability of source codes and free samples. PHP code can be embedded inside a HTML page that will be executed each time the page is visited. After the execution, HTML or other outputs will be displayed on the page.

JavaScript, developed originally by Netscape, is a lightweight interpreted programming language (http://java.sun.com/javascript/). It is the only scripting language currently supported by the popular web browsers. Normally it is used to extend the capabilities of HTML such as input validations and pop-up windows. The JavaScript embedded into the HTML web page will be loaded with the pages and the JavaScript code will be executed when an event is triggered. The events can be clicking a button, opening a new window, etc.

Apache server is a robust, commercial-grade, powerful and free source code implementation of an HTTP web server (http://www.apache.org). A team, named
“Apache group”, first developed this software. Hundreds of users have subscribed ideas, code, and documentation to this development project. Currently it is a very popular tool for online server development.

Macromedia Dreamweaver MX 2004 is the professional option for building web sites and applications (http://www.adobe.com/products/dreamweaver/). It provides a powerful combination of visual layout tools, application development features and code editing support. It enables developers and designers with different backgrounds and skills to create visually appealing, standard-based sites and applications.

1.6 Summary

Project background information, DOS based CO2SYS program information, project objectives and programming tools are presented. In the next chapter, we will discuss the preliminary design of the proposed web based CO2SYS program.
Chapter 2
Preliminary system design

2.1 Overview

As we described in Sections 1.2 and 1.4.3, the goal of this project is to develop a web-based version of the CO2SYS program to replace the current DOS based version. The new version keeps all the functionality and calculation methods of the DOS version. The functional requirements for the new version are based on the DOS version. This chapter will first discuss potential users of the new system, list all the functional requirements with individual design priority ranking, and lastly present preliminary system design with screen shots.

2.2 Potential users of the system

Since this program basically calculates four important terms of the CO\textsubscript{2} system in water chemistry, the potential users for this newly designed program will be mostly water chemists. The users do not have to have high computer literacy because this system will be very user friendly and straightforward. The users just need some basic knowledge of web browsers, which is not a problem nowadays due to the popularity of PCs. They could be male or female researchers. Here we assume these chemist users already have some general knowledge about the scientific terms in this program. As we mentioned in Section 1.4.3, the users can check the detailed information of the terms and options through the hyperlinks nearby or through the on-line help (e.g., the definition of total alkalinity, etc).
2.3 Functional requirements

Basically there will be two types of water samples (freshwater or seawater) and two kinds of input mode (single mode or batch mode) for the system. This system has four scenarios on the input window: freshwater under the single mode or under the batch mode, seawater under the single mode or under the batch mode. Below are the preliminary functional requirements and their individual priority level for each scenario.

2.3.1 Freshwater or seawater sample under the single input mode

Due to the similarities between freshwater and seawater samples under the single mode for calculations, all of the functional requirements for both are listed together. The design priority ranking for each function is included with the functional requirement descriptions below. The functions exclusively for seawater samples are marked, as well. A clear description of the data calculations under the single mode is shown in Figure 2.1.

1. Users should be able to choose the water type (High)
2. Users should be able to choose the input mode (i.e., single mode) (High)
3. Users should be able to choose the constants for calculations (seawater only, High)
4. User should be able to choose the type of constants for $K_{\text{SO}_4}$ (seawater only, High)
5. Users should be able to choose the type of pH scale (seawater only, High)
Figure 2.1 The flow diagram for the calculations of the web based CO2SYS program. This diagram adopts the similar flow for the data calculations in the DOS version CO2SYS program. The left side of the diagram lists the calculation steps for both freshwater and seawater under the single mode. The steps for both freshwater and seawater under the batch mode are on the right side. The steps only applicable for seawater are marked with an asterisk (*).
6. Users should be able to input the value for salinity, phosphate and silicate (seawater only, High)

7. Users should be able to enter the input temperature and pressure (High)

8. Users should be able to enter the output temperature and pressure (High)

9. Users should be able to choose between $f$CO$_2$ and $p$CO$_2$ (High)

10. Users should be able to choose two input parameters and input the value for each (High)

11. Users should be able to see the error message and re-input when they input invalid data (High)

12. Users should be able to change the options and values above before the execution or re-calculate with different options and input values (High)

13. Users should be able to run the execution (High)

14. Users should be able to see the execution results, which include the values of the input parameters and output parameters (High)

15. Users should be able to save the results to a local file directory (High)

16. Users should be able to check the reference links beside each step above (High)

2.3.2 Freshwater or seawater samples under the batch mode

As for the single mode, all the functional requirements for both freshwater and seawater under the batch mode are listed together. Each functional requirement is presented with its design priority ranking below and marked if it is only for seawater.
Figure 2.1 includes a detailed list of the steps for the data calculations under the batch mode.

1. Users should be able to choose the water type (High)

2. Users should be able to choose the input mode (i.e., batch mode) (High)

3. Users should be able to choose the constants for calculations (seawater only, High)

4. Users should be able to choose the type of constants for $K_{SO_4}$ (seawater only, High)

5. Users should be able to choose the type of pH scale (seawater only, High)

6. Users should be able to choose between $fCO_2$ and $pCO_2$ (High)

7. Users should be able to choose two input parameters (High)

8. Users should be able to input or choose the local input data file and change the input file at any time (High)

9. Users should be able to input the number of header lines (High)

10. Users should be able to input the number of ID fields (High)

11. Users should be able to input the designated value of missing value in the input data file (High)

12. Users should be able to input or choose the local output data file and change the output file at any time (High)

13. Users should be able to find the file format information for input and output data file (High)

14. Users should be able to choose the option of a missing value flag in the output data (High)
15. Users should be able to enter additional information that can be added to the output data file (Medium)

16. Users should be able to see the error message and re-input when they input invalid data (High)

17. Users should be able to change the options and values above before the execution or re-calculate with different options and input values (High)

18. Users should be able to run the execution (High)

19. Users should be able to check the reference links beside each step above (High)

2.3.3 Other functionalities for the system

1. Users should be able to get access to the reference information at any time (High)

2. Users should be able to get access to the reference papers and their typographical errors for each reference (Medium)

3. Users should be able to get access to the contact information of the authors for comments and suggestions (Medium)

4. Users should be able to get access to the detailed documentation for the system (High)

2.4 Preliminary system design

Based on the functional requirements listed above and the flow diagram for the data calculations shown in Figure 2.1, the preliminary system design of the web-based CO2SYS program was developed. Since the new version keeps all the functionalities and
calculation methods as the old version, this design follows the similar sequence as the
DOS version for data calculation, which will be shown later. However, this design
provides better visualizations and simpler layout for user inputs. Here we only show the
web pages for four typical scenarios of data calculations (Section 2.3). The details of
documentation web pages will be discussed in later chapters.

2.4.1 Freshwater under the single mode

After the system starts, the user chooses “Freshwater” as the water type and
“Single mode” as the input mode. The user can continue to enter the information for input
and output conditions. After “fCO2” is chosen for CO2 and “TA, TC” is selected for
input parameters, the complete layout for user inputs and options under the single mode
for freshwater sample is displayed (Figure 2.2). Compared to the DOS version, this
design adds the option of water type, which clearly points out the water sample type for
the user. The choices for the input mode, $f\text{CO}_2$ or $p\text{CO}_2$ and input parameters can be
made by the drop-down menu instead of jumping between screens as in the DOS version.
All the inputs and options are also displayed in one page instead of several small screens
in the DOS version. All inputs and options can be changed at any time before the
“Execute” button is clicked. In the DOS version, an input cannot be changed (unless
restarted) once it is entered.

Once the “Execute” button is clicked, a separate web page opens with the
calculation results (Figure 2.3). All the information of the calculation results, which take
four screens to display in the DOS version, is presented in a single page. The results are
Figure 2.2 The preliminary design web page of data calculation for freshwater sample under the single mode after the user enters the input and output conditions. The user can choose from the drop-down menus (A), check the relevant information from the hyperlink (B), input a value (C), or find more information in the documentation of program (D). Once if the “Execute” button is clicked, the program will do the calculation and present the data results (Figure 2.3).
Section A contains the water type, input parameters, input and output conditions; pH and $f\text{CO}_2$ at both input and output conditions (B), the sensitivity of pH and $f\text{CO}_2$ to small variations of the inputs (C), the compositional contributions to TA (D) and TC (E), $f\text{CO}_2$ and $p\text{CO}_2$ in 100% saturated air, the mole fraction of CO$_2$ in dry air ($x\text{CO}_2$, 1 atm pressure), Revelle factor, and the values of dissociation constants ($pK$’s, F) are also included.

displayed in a similar layout as in the DOS version, which lowers the learning curve when the user switch from the DOS version program to the new version. Color bars in
Figure 2.4 also provide clear indications of the data sections for the user, which is more visually attractive than the black and white screens.

Figure 2.4 The preliminary design web page of data calculation for freshwater sample under the batch mode. The user can click the “File format” hyperlink to get the specific information about input file format.
2.4.2 Freshwater under the batch mode

To run the data calculations for freshwater under the batch mode, the user can keep the water type as “Freshwater” and change the input mode to “Batch mode” from the page in Figure 2.2. If “fCO2” is chosen for CO$_2$ and “TA, TC” is selected for input parameters, the page layout will become the one shown in Figure 2.4. As mentioned in Section 1.2.2b, the batch mode takes a large set of data from an input file for the calculation. The user can type the name of the file directory as in the DOS version. If the file directory is too long, the user can find the directory with the built-in “Browse” function, which is not available in the DOS version. After the user finalizes all the inputs and options and clicks the “Execute” button, the program will read the data from the input file and store the calculation results into the output file.

2.4.3 Seawater under the single mode

If the water type is changed from “Freshwater” to “Seawater” and the input mode from “Batch mode” to “Single mode” from the page in Figure 2.4, the program starts to do the calculations for seawater sample under the single mode. Besides the inputs and options shown in Figure 2.2, the user will be asked to choose the constants (e.g., Roy et al., 1983), the value for K$_{SO4}$ (e.g., Dickson), the pH scale (e.g., Total scale (mol/kg-SW)) and enter the values for salinity, phosphate and silicate. After the “Execute” button is clicked, the user should be able to see the calculation results displayed in a separate page (Figure 2.6).
**Figure 2.5** The complete page layout of data calculation for seawater sample under the single mode. The calculations for seawater sample are more complex than for freshwater sample as shown in Figure 2.2 because seawater has more ingredients than freshwater such as salts (salinity), phosphate, etc.
Figure 2.6 The calculation result display page with the inputs and options shown in Figure 2.5 for seawater sample under the single mode. Due to more ingredients in seawater than in freshwater, several things in this figure do not appear in Figure 2.3: phosphate and silicate (A), the sensitivity of pH and fCO$_2$ to small variations of extra input (salinity, B), additional contributors to TA (C), omega (the degree of saturation) for calcite and for aragonite (D), pH at different scales and activity coefficient ($f_H$) (E), dissociation constant for boric acid (F).
Figure 2.7 The preliminary design web page of data calculations for seawater sample under the batch mode. This page is similar to Figure 2.4 except three additional options: the constants, the value for $K_{SO_4}$ and the pH scale.
2.4.4 Seawater under the batch mode

If the input mode in Figure 2.5 is changed from “Single mode” to “Batch mode” and the water type is still the same (seawater), that means the program is running the calculations for seawater sample under the batch mode. In addition to the steps described in Section 2.4.2, the program asks the user to choose the constants, the value for $K_{\text{SO}_4}$, and the pH scale, which is the same as for seawater under the single mode. The values of salinity, phosphate and silicate, which are required to input in Figure 2.5, are included in the input data file and thus not shown here. After all the inputs and options are finished, the user can click the “Execute” button and the program will run the calculations. All the calculations results will be recorded into the output file.

2.5 Summary

Based on the DOS version program and the functional requirements listed in Section 2.3, the preliminary design of the web based CO2SYS program is generated. In the next chapter, we will discuss the informal testing of this preliminary system design without real users.
Chapter 3

Cognitive walkthrough and heuristic evaluation

3.1 Overview

In the previous chapter, the preliminary design of the web based CO2SYS program was discussed. This chapter will talk about how to evaluate this preliminary design with two simplified methods: cognitive walkthrough and heuristic evaluation.

3.2 Informal system testing without real users

Usability of a web page interface is very important for users. They can get frustrated, keep making mistakes, waste their time and be unable to finish tasks. So usability evaluations are critical to the development processes of the interface. The field of human-computer interaction (HCI) recommends conducting usability evaluations early in the design process, especially in the stages of early prototyping (Rieman et al., 1995). If the critical flaws can be found at an early stage, they also have higher chances to be found and fixed due to the relative simplicity of the system. When the project gets more and more complicated at later stage, it will be extremely difficult to find and fix the problems. So usability evaluations at an early stage save lot of time and effort for the project in the long run. Empirical user testing is the most comprehensive evaluation technique, which is expensive and requires a lot of time and the involvement of real users. To save time and cost, most developers only use this technique at the later stage or the end of the design cycle. Sometimes the changes are very costly and difficult to make, so the recommendation from users are often ignored unfortunately.
Here we introduce two types of methods that can be used to evaluate an interface without real users. The first one is “Cognitive walkthrough”, a task oriented method for evaluating the design of a user interface and checking how well the interface can support first time use without formal training (i.e., exploratory learning) (Polson et al., 1992; Rieman et al., 1995; Wharton, et al., 1992). This evaluation can be performed by the system designer at the early stage of design. The second one is “Heuristic evaluation”, a technique used to check how well the interface complies with a list of general principles and guidelines for interface design.

3.3 Cognitive walkthrough

Cognitive walkthrough has two phases: preparation and evaluation. For the preparation phase, four kinds of information have to be ready before the walkthrough starts: 1.) A preliminary description or prototype of this interface. It does not have to be complete, but it should include as many details as possible. As we described in Section 2.4, a preliminary design of the web page layout is ready for the evaluation. 2.) A list of tasks that should represent those used for the task-centered design of the system. Based on the typical four scenarios in Section 2.3, four main tasks are proposed here for the walkthrough evaluation:

i.) Given two parameters (TA and TC), calculate the other two parameters (pH and $f_{CO_2}$) for freshwater under single mode

ii.) Given two parameters (TA and TC), calculate the other two parameters (pH and $f_{CO_2}$) for freshwater under batch mode
iii.) Given two parameters (TA and TC), calculate the other two parameters (pH and $f$CO$_2$) for seawater under single mode

iv.) Given two parameters (TA and TC), calculate the other two parameters (pH and $f$CO$_2$) for seawater under batch mode

3.) A complete, written list of the actions used to complete each individual task on the interface. 4.) An idea of who the users will be and what kind of experience they will have when they first use the system, which is discussed in Section 2.2. This understanding should be developed through the task and user analysis.

After the preparation phase is complete, the designer can start the cognitive walkthrough process. During the walkthrough process, the designer tries to tell a story about the prospective user’s motivation and examines each individual step in the correct action sequence. To critique if the story is believable or not, four questions below can be asked to help the designer evaluate whether the user will choose the correct action.

1.) Will users be trying to produce whatever effect the action has?

2.) Will users be able to locate the control (button, menu, switch, etc.) for the action?

3.) Will the control users found help them reach the goal they want?

4.) Will the interface provide understandable feedback for users after each action?

The first question above talks about what the user is thinking. Most of the time users do not think or act the way the designer expects them to. If users want to calculate for freshwater sample under the batch mode, will they set the water type to “Freshwater” and the input mode to “Batch mode” before they run the calculations? The second question deals with the difficulty of finding the right control by the user. Sometimes it is
really hard to find the control, not to mention the right control. Will users know the right controls to set the water type and input mode for any specific data calculations, for example? The third question concerns the identification of the control. If users find the control, can they tell this is the right control that can implement the tasks they want? These three questions are interactive. For example, a clearly labeled control may remind users what needs to be done even if they might not do the right thing initially. The final question asks about the feedback after the action is complete. Users need to see the feedback from the system, even the simplest, so they will know the system has taken the action. The detailed walkthrough information including the listed actions for each task is described in Appendix A.

After the walkthrough with the main tasks above on the preliminary prototype, we found that this version of design passed most of the evaluations at this development stage. The only problem we caught during this walkthrough process is that there is no feedback from the system after the user clicks the “Execute” button to run the calculation under the batch mode, as shown in Figure 3.1 and in Appendix A. After the calculation is complete, the system should give a message about the status of the calculation for the user. The solution for this problem will be discussed in the next chapter.

3.4 Heuristic evaluation

Heuristic evaluation, mainly proposed by Nielsen (1989), and Nielsen and Molich (1990), are general guidelines or principles used to guide interface designs. It is inexpensive relative to other evaluation methods and easy to use. The evaluators do not have to have formal training and advanced planning is not required.
Heuristic evaluation is a user-centered method, which is different from the cognitive walkthrough as a task-oriented technique. Task-oriented techniques can detect interface problems occurring while the user tries to finish the tasks and describe the
importance of the problems under the live situation. But it is impossible to evaluate every
task the user will perform. Each user may have a different sequence of actions and
control executions. Moreover, each task is evaluated separately and the cross-task
interactions are not identified. So this task-free user-centered method, heuristic
evaluation, is brought in to catch problems that cognitive walkthrough may have missed.

Based on their own experience, Nielsen and Molich (1990) have summarized nine
heuristics for this evaluation method:

1. Simple and natural dialog: the interface should not have irrelevant or rarely
   used information. All the information should be arranged to match the task.
2. Speak the user’s language: system-specific engineering terms should be
   avoided and concepts from the user’s world should be considered.
3. Minimize user memory load: keep the information on the screen until not
   needed so the user does not have to remember things from one action to
   another.
4. Be consistent: one action sequence the user learned from one place should
   also apply in another place.
5. Provide feedback: give the user clear message of the effect from their actions.
6. Provide clearly marked exits: the user should be able to enter or exit from any
   part of the system without any difficulties or damages to the system.
7. Provide shortcuts: shortcuts can help experienced users with efficiency by
   bypassing unnecessary steps and information.
8. Good error messages: they can tell the user what the problem is and how to
   correct it.
9. Prevent errors: help the user avoid making mistakes and save their time and effort.

The four main tasks used in Section 3.3 for the cognitive walkthrough are evaluated with these nine heuristics again. The detailed results for this evaluation are included in Appendix B. Most of the preliminary design fits the user interface standards. There are two main problems found from this evaluation. There is no feedback from the interface when data calculations are performed under batch mode, which is the same as the one found from the cognitive walkthrough process. Secondly, lack of error checking for the user input makes the program unable to handle the errors for the user as shown in Figure 3.2. When the user enters an invalid value, the system should generate an error message to help the user fix the error. Useful information for each input should be provided to help the user from entering invalid inputs. The detailed solutions to these two problems will be addressed at next stage of the development.

3.5 Summary

Two types of evaluation methods, cognitive walkthrough and heuristic evaluation, were used by the designer to evaluate the preliminary design of the web-based CO2SYS program. Since the system is still at a premature stage and it is not totally functioning yet. The problems found from these two evaluation processes would be fixed during further implementation from the preliminary design of the system. Real users would be brought in to test the mature version of the program, which is discussed in the next chapter.
Figure 3.2 The web page layout of data calculations for freshwater sample under the single mode after the user enters an invalid value for the input temperature. The invalid input is marked with a red circle.
Chapter 4

User testing

4.1 Overview

The quality of the system interface cannot be truly evaluated until it is evaluated by real potential users. The previous system evaluations (cognitive walkthrough and heuristic evaluation) performed by the designer are very useful at the early stage of the design. However, some of the big problems may be overlooked since the designer knows so much of the system. Some things are assumed to be obvious to the designer, which may not be clear to users. Since the system is still in the early stage of design, it may not be very useful for real users to try it out yet. When more implementation is added to the preliminary system and the system becomes mature, user testing should then be conducted. Real or potential users should be brought in to try the system and record the problems found during their testing.

User testing needs potential users as testers, a mature version of system for users to interact with, and a list of tasks for users to perform on the system (Lewis and Rieman, 1993). This chapter will discuss the selection of testers and some ethical issues related to user testing, detailed information related to the prototype system used for evaluation, and a list of tasks for users to perform with the system. The chapter concludes with a summary of the testing results, followed by solutions to the problems discovered during testing.
4.2 User testing participants

The goal of this user testing is to find what will happen when real users use this system. So the best testers are those who are representative of the real users. If it is really hard to find real users, people with some approximation to real users can be brought in as alternatives for the testing. Five students majoring in water chemistry studies were invited to this user testing. Their background ranges from undergraduate student to postdoctoral researcher, which is representative of the actual users of the system. Each user has been asked to complete a background questionnaire prior to testing. The purpose of this questionnaire is to acquire an understanding of the background knowledge of the chosen users (e.g., the familiarity with the DOS version). The questionnaire and the detailed user demographic information are listed in Appendix C. Table 4.1 summarizes all the users’ demographic information.

4.3 Ethical issues while working with users

Since all users are asked to test the system voluntarily, it is the responsibility of the testing director to ensure that the testing sessions are not overly stressful for users. It is not an easy job for users to test a system that looks totally new to them. It is possible that some users cannot finish the assigned tasks and have to terminate the testing with embarrassment, disappointment and sadness. Before the test, all the users were asked to participate with voluntary and informed consent and also told that they can quit at any time without any reason. They were also reassured that it is the system being tested, not their abilities in any way. If problems happen during the test, it means the system has some flaws that need to be fixed. If users get constantly distressed, the test should be
paused or even stopped to protect users. All the information obtained from the test will be assigned a code and should be kept away from users’ identification for privacy reasons.

**Table 4.1** The demographic information of all users for testing sessions.

<table>
<thead>
<tr>
<th></th>
<th>User 1</th>
<th>User 2</th>
<th>User 3</th>
<th>User 4</th>
<th>User 5</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Student status</strong></td>
<td>Postdoctoral</td>
<td>Undergraduate</td>
<td>Graduate</td>
<td>Graduate</td>
<td>Graduate</td>
</tr>
<tr>
<td><strong>How much are you familiar with the web?</strong></td>
<td>Very familiar</td>
<td>Very familiar</td>
<td>Somewhat familiar</td>
<td>Familiar</td>
<td>Familiar</td>
</tr>
<tr>
<td><strong>Do you fill in forms at online websites very often?</strong></td>
<td>Sometimes</td>
<td>Sometimes</td>
<td>Often</td>
<td>Sometimes</td>
<td>Often</td>
</tr>
<tr>
<td><strong>Have you ever used a web-based program (e. g., online calendar, online document management, etc)?</strong></td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Have you ever used an online calculator type program (e. g., loan calculator program, online expense program, etc)?</strong></td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Have you ever used a command line program (e. g., DOS, Unix, etc)?</strong></td>
<td>Yes</td>
<td>Familiar</td>
<td>Not familiar</td>
<td>Familiar</td>
<td>Somewhat familiar</td>
</tr>
<tr>
<td><strong>How much do you know about the DOS version of the CO2SYS program?</strong></td>
<td>Very familiar</td>
<td>Very familiar</td>
<td>Very familiar</td>
<td>Familiar</td>
<td>Very familiar</td>
</tr>
<tr>
<td><strong>How much do you know about the water chemistry terms such as alkalinity, dissolved inorganic carbon, partial pressure of CO₂ gas, etc?</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Before the test starts, a test script was read to the user about the purpose of this test. Then a subject information and consent form was presented to the user. This test script and the consent form address the ethical issues we mentioned above. Since the testers are human subjects, we have complied with all the guidance and rules from Institutional Review Board (IRB) of the University of Montana. All testing materials
were approved by the IRB and only the consent forms with IRB approval stamps were used. After the testers signed the consent forms and filled in the background questionnaire, they proceeded to test the system with a list of tasks described later in this chapter. After the test was complete, they were asked to comment on this system in a post-test questionnaire and interview. “Test script”, “Subject information and consent form”, “Post-test questionnaire” and “Post-test interview” used for this testing are shown in Appendix C.

4.4 The system evaluated by users

A large amount of implementation detail has been added to the preliminary design of the system. Here we present the specific version of the program used for the user testing. Figure 4.1 shows the revised web page used for the freshwater sample calculation under the single mode. Compared to Figure 2.2 shown at the preliminary design stage, a pop-up window is added when the system starts which gives a brief introduction about this new system. The format of pressure input is changed only for “dbar” without “meters” for simplicity. The inputs of two input parameters (i.e., TA and TC), which are held on the same line in Figure 2.2, are split into two separate inputs for clarity. A “start over” button is added to allow the user to reset the program back to the beginning. After the “Execute” button is clicked, a separate window pops out with all the calculation results (Figure 4.2). The input parameters and conditions are marked with yellow color background, which provides clearer visual indication for the user than the layout shown in Figure 2.3.
Figure 4.1 The complete revised web page layout of data calculation for freshwater sample under the single mode.
Figure 4.2 The revised web page layout for freshwater sample under the single mode. The inputs and options are shown in Figure 4.1.
The batch file input and output file input are changed from previous version in Figure 2.4. As shown in Figure 4.3, the program prints out the input and output file directories, which provide nice feedback for the user. The user can also change the file directory with the “Change input file” or “Change output file” buttons. An extra input for output file is added to allow the user to record additional information to the output file. This input is not implemented in Figure 2.4 at the preliminary design stage due to its medium design priority level (Section 2.3.2). If the user clicks the “Execute” button to start the calculation, the program will give a feedback message after the calculation is finished (Figure 4.4). As discussed in the cognitive walkthrough process (Sections 3.3) and heuristic evaluation process (Section 3.4), the program does not provide clear feedback for the user after the completion of the calculation. The message shown in Figure 4.4 easily solves this problem. If users want to check the content of the output file (i.e., calculation results), they can click “OK” in Figure 4.4 and the detail of the output file is listed in Figure 4.5.
Figure 4.3 The revised web page layout of data calculation for freshwater sample under the batch mode.
Figure 4.4 A pop-up message as feedback after the program finishes the calculation process under the batch mode. The inputs and options for the calculation shown on the web page are shown in Figure 4.3.
Figure 4.5 The output file data display page of data calculation for freshwater sample under the batch mode. This page comes up after clicking “OK” button on the pop-up message shown in Figure 4.4.

As stated in Section 3.4, the preliminary design of the system did not provide good error messages for invalid user input, nor did it help prevent the user from making
errors. The revised system shown in Figure 4.6 enables error checking for the user input. The user can also get further information of each specific input from the hyperlink nearby (Figure 4.7). This hyperlink provides helpful information about the valid format and range of the input, thus helping to prevent errors. If the user forgets to correct the invalid input before the execution of the data calculation, the program will scan through the whole page and check for invalid inputs. If there are any, an error message will come up to remind the user (Figure 4.8). The program only takes valid inputs for data calculation.

![Image](image.png)

**Figure 4.6** An error message from the system after an invalid input for “Input temperature”. The input temperature should be a non-negative number and fall between 0 and 40 °C.
A pop-up message with detailed information of the input for “salinity”. This message includes the definition of salinity, the valid range of the input and its default value. The user can also check other hyperlinks for useful information such as “phosphate” and “silicate.”
Figure 4.8 Error message given by the program due to multiple invalid inputs on the screen. After the user clicks “Execute” button, the program first checks the validity of all the inputs. The calculation does not proceed until all the inputs are valid.

The preliminary design does not include the functionalities mentioned in Section 2.3.3. Four proposed functions listed in Section 2.3.3 are addressed in Figure 4.9 to 4.12.
Figure 4.9 basically talks about the initiatives of CO2SYS program and the functional capabilities of the program. All the reference papers used for the calculations of CO2SYS program are listed in multiple pages together with their individual typographical errors, if any. Valuable comments and suggestions are welcomed by the authors through the contact information shown in Figure 4.11. The detailed documentation of CO2SYS program can be accessed through the linkages shown in Figure 4.12 for the user’s convenience.

![Figure 4.9](image-url)

**Figure 4.9** The introduction page for the web-based CO2SYS program. To display this page, the user can click the linkage “Introduction” on the left column.
Figure 4.10 The reference page for the web-based CO2SYS program. The user can check the typographical error information for the corresponding reference with the hyperlink, e.g., the pop-up message for reference “Dickson_1979b”.


Dickson, A. G. 1990a. Standard potential of the reaction: $\text{AgCl(s)} + 1/2 \text{H}_2(\text{g}) = \text{Ag(s)} + \text{HCl(aq)}$, and the standard acidity constant of the ion $\text{HSO}_4^-$ in synthetic seawater from 273.15 to 318.15 K. *Journal of Chemical Thermodynamics* 22:113–127. Typographical errors


Figure 4.11 The contact information of the web-based CO2SYS program for potential users.
4.5 Select the tasks for user testing

As mentioned in Section 4.1, the essence of the user testing is to ask potential users to test the system. The user tries to finish the tasks assigned by the developer and the developer records how well the user can finish the tasks. Since good testers should be typical of real users, the test tasks should be designed to simulate what the real tasks will
look like. Cautions have to be taken to avoid making the tasks easier or bend the tasks in the direction that the system supports most. We have designed 5 scenarios that contain different tasks for the testers to practice. The tasks in each scenario rank from easy to difficult and involve most of the main functionalities on the system.

During the test, we need to gather enough information about how the testers try to finish the tasks with the current system. These include what is happening step by step, why it happens, what is the problem, and how successful to finish the tasks, etc. All the test tasks and their corresponding data capture forms are shown in Appendix C.

4.6 Problems found from user testing

Table C6 in Appendix C summarizes how well each user completes the tasks. Here we list the problems all users encountered when they tried to finished the tasks (listed in Appendix C) of the system:

a.) When the user enters the full directory of either the input file or output file, then hits the “Enter” key, there is no response from the program. The user has to click the mouse somewhere outside of the input area to reactivate the program. This happens because the browser (e.g., Internet Explorer) does not know the “Enter” key is the end of the user input.

b.) If the user enters the values for TC and $f_{CO_2}$ (e.g., 5 for TC and 6000 for $f_{CO_2}$) that triggers an error message, the message still comes out even if the user resets these two values back to valid ones. This problem happens because the program automatically closes the error message window when all the inputs are valid.
c.) Under the single mode, when input parameters are TA and TC (such as 5 and 6000), the error message that only happens when TC and $f_{\text{CO}_2}$ are input parameters is still triggered.

d.) When running on the seawater single mode with TC and $p_{\text{CO}_2}$ as the input parameters, the calculation results displayed in the window are not centered.

e.) If the user clicks the “Home” hyperlink on the left column of the window (Figure 4.2) and starts a new calculation, the programs keeps asking the user to refresh the page and can not move on to do the calculation. This problem happens because the program checks if there is more than one option with “Selection” as the first choice. If yes, it will refresh the page.

f.) If the user clicks any hyperlink on the left column in Figure 4.5 such as “Introduction” and tries to run another batch mode calculation, the program cannot display the new content of the output data file. Instead it always shows the previous content.

g.) One of the testers has suggested highlighting the input box with red color when the input value is not valid. Sometimes it is hard to see which one is invalid after the input box loses focus of the keyboard.

h.) When there is no error in the user input, the error message still comes out and closes very quickly after the user clicks the “Execute” button. When the user changes one of the inputs to invalid one and recalculates (definitely the calculation can not continue due to the input error), the still open result display will be closed automatically by the system. The purpose of this design is to prevent the confusion from the user. This kind of design makes one of the testers complain about the annoying pop-up of the window.
For task 4/3, users were asked to change one of the “Constants” options and keep the rest of the inputs. They have to re-enter all the inputs again after the “Constants” is changed. They think it is annoying to do that. The program should keep the previous inputs automatically if they are not changed by the user.

4.7 Results from post-test questionnaire

Each user was asked to answer the questions in the post-test questionnaire right after they finished the testing. The results of the questionnaire are listed in Table 4.2. User 1 and user 4 think it is difficult or somewhat difficult to change the option and recalculate because the program asked them to input all the values again after just one of them is changed. They think the program should “remember” them, which is also addressed in Section 4.6.

User 3 thinks it is somewhat difficult to determine and recover from the errors because of the initial confusions by the meaning of the user testing and by the invalid values of those inputs. The background knowledge tells them the value is not right. It is common for users without any previous testing experience. Finally this user was able to work it out and finished the testing.

4.8 Results from post-test interview

All users were interviewed after they finished the post-test questionnaire. The results of the interview are shown in Table 4.3. User one has suggested highlighting all the invalid user inputs after the error message comes out. In addition to the problem (f) described in Section 4.6, he also asked if there is a “print” function in this program. Since
this is web-based online program, the user can click “print” or “print preview” on the browser menu to print out the page. However, It will be nice for the user to be able to print out a “printer friendly version” of the calculation results.

Table 4.2 All the information from the user post-test questionnaire is listed below. The user can choose one of the responses for each statement listed here. The possible responses are “very easy”, “easy”, “Somewhat difficult”, “difficult” and “very difficult”.

<table>
<thead>
<tr>
<th></th>
<th>User 1</th>
<th>User 2</th>
<th>User 3</th>
<th>User 4</th>
<th>User 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>I felt that the completions of the tasks overall was</td>
<td>Easy</td>
<td>Easy</td>
<td>Easy</td>
<td>Easy</td>
<td>Easy</td>
</tr>
<tr>
<td>I felt that finding the help information on the web page was</td>
<td>Easy</td>
<td>Easy</td>
<td>Easy</td>
<td>Easy</td>
<td>Easy</td>
</tr>
<tr>
<td>I felt that changing the options or inputs and recalculating were</td>
<td>Difficult</td>
<td>Very easy</td>
<td>Very easy</td>
<td>Somewhat difficult</td>
<td>Very easy</td>
</tr>
<tr>
<td>I felt that obtaining the feedback from each action was</td>
<td>Easy</td>
<td>Very easy</td>
<td>Very easy</td>
<td>Very easy</td>
<td>Very easy</td>
</tr>
<tr>
<td>I felt that determining and recovering from errors were</td>
<td>Very easy</td>
<td>Very easy</td>
<td>Somewhat difficult</td>
<td>Easy</td>
<td>Easy</td>
</tr>
<tr>
<td>I felt that understanding the input format information was</td>
<td>Easy</td>
<td>Very easy</td>
<td>Difficult</td>
<td>Very easy</td>
<td>Easy</td>
</tr>
</tbody>
</table>

User two thinks this program is straightforward and helpful in guiding the user through each step. User three thinks this program is very user friendly and easy to navigate.

User four thinks the program is easy to use, has good usage of prompts, and does not require much of the user. She recommends that the program should be less “mouse-dependent”, i.e., “Tab” and “Enter” keys should make the program moving. She thinks it
is better to remind the user on the error message so that they can check the help information on the hyperlink beside the user input.

Table 4.3 All the information from the user post-test interview is listed below.

<table>
<thead>
<tr>
<th>Question</th>
<th>User 1</th>
<th>User 2</th>
<th>User 3</th>
<th>User 4</th>
<th>User 5</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>What do you think about this system generally?</strong></td>
<td>Good idea, but still need a little work</td>
<td>Straightforward and helpful in guiding the user through each step</td>
<td>Very user friendly and easy to navigate</td>
<td>Easy to use, good use of prompts, does not require much of the user</td>
<td>Easy to follow, has lots of input choices, good error checking</td>
</tr>
<tr>
<td><strong>What additional functionalities do you think should we add to the system?</strong></td>
<td>Highlight errors, fewer pop-ups, if one input is changed, still keep the other inputs</td>
<td>None</td>
<td>None</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td><strong>Do you have some ideas about the changes we should make to the interface?</strong></td>
<td>Is there is “print” option?</td>
<td>None</td>
<td>None</td>
<td>Maybe make it less “Mouse-dependent”, be able to hit “Enter” key to move on</td>
<td>Add extra text for the user to click outside the input box to move to next step</td>
</tr>
<tr>
<td><strong>If you have ever experienced with the DOS based version of this CO2SYS program, can you compare and contrast these two versions of programs?</strong></td>
<td>This version is potentially easier to use</td>
<td>Much easier to use</td>
<td>None</td>
<td>This is much easier to use and follow, data retrieval is nice and good setup</td>
<td>Web version is more user friendly, easier to calculate batch mode and recalculate</td>
</tr>
<tr>
<td><strong>Do you have any other suggestions?</strong></td>
<td>None</td>
<td>None</td>
<td>None</td>
<td>None</td>
<td>None</td>
</tr>
</tbody>
</table>
User five thinks this program is easy to follow and has a lot of input choices and good error checking. She mentions the same problem as other users about the “Enter” key. She likes this new online version more than the previous DOS version.

Compared to the DOS version system, user 1 and user 2 consider this new online version a good idea and it should be potentially easy to use. User 3 thinks this version is better and does not provide further comments. User 4 thinks this version is much easier to use and follow. Its data retrieval is nice and has good set up. She is eager to know when this version will be ready to use. User 5 thinks this web version is more user-friendly and easier to calculate under the batch mode and recalculate after changing the inputs.

4.9 Changes to the system following user testing

After the user testing, the CO2SYS program has been changed to according to the problems found during the tests and comments and suggestion from users. After the user finishes an input, they can hit the “Tab” key, the “Enter” key, or click the mouse anywhere outside the input area to move forward. The focus of the keyboard can be moved between inputs or options by hitting the “Tab” key or the “Enter” key. This solves the problem (a) described in Section 4.6 about the malfunction of the “Enter” key.

When the program runs under the single mode, the error message due to unrealistic inputs (e.g., TC is 5 and $f_{CO_2}$ is 6000) only comes out when the input parameters are TC and $f_{CO_2}$ (Figure 4.13). This error message has to be closed before users can make adjustments to the inputs on the input window. This design addresses several problems that happened during the user testing such as problems (b) and (c), and error message not closed by the user before changing the inputs. For problem (d), the
output data format is aligned to the center for TC and $p\text{CO}_2$ as the input parameters. The bug in problem (e) is fixed by disabling the unnecessary checking of the existence of more than one option with “Selection” as the first choice.

**Figure 4.13** Error message for unrealistic user input for TC and $f\text{CO}_2$ under the single mode for freshwater samples. Note that the focus of the keyboard is on the error message window instead of on the input window. The input window cannot be accessed without closing the error message window.
To solve problem (f) in Section 4.6, the window to display the content of output file data is simplified as shown in Figure 4.14. The previous design for this window is demonstrated in Figure 4.5. The top part and the left column of the window in the revised design are both taken out for simplicity. After the user runs a second calculation under the batch mode, this window will be automatically reloaded with the new content of the output file.

Figure 4.14 A separate window containing the content of the output file data after the calculation under the batch mode.
If the user enters an invalid input, the program will pop out an error message and the background of the input box will be highlighted in red (Figure 4.15). The program will automatically re-set the input to the default values (e.g., “0” for the input pressure). The error message also reminds the user to get helpful information from the hyperlink beside each input. The input background will change back to normal after a valid input. This design makes all the text inputs either stay on the default values or valid values from the user. So the program does not have to check the validity of each text input before the calculation. The error checking procedure for the whole web page mentioned in Figure 4.8 is taken out of the system and problem (h) is easily resolved.

As shown in Figure 4.16, the type of constants, pH scale and $K_{SO_4}$ used for calculation are included for the calculation result display. All the information of inputs and options on the input window are displayed on the top of the page and the bottom of the page shows the calculation results. On the left column, a “printer friendly version” link is added so the user can get access to the printer friendly page. The user can also get Microsoft Word version (Figure 4.17) and Excel version (Figure 4.18) of the calculation results through the links on the left side of the page (Figure 4.16 D and E).
Figure 4.15 Error message with further directions after the invalid input for input temperature during the data calculation under the single mode for freshwater samples.
Figure 4.16 The printer friendly version of the data calculation results under the single mode for seawater samples. The type of constants, pH scale and $K_{SO_4}$ used for the calculation (A), the unit for $fCO_2$ (B), printer friendly version menu, Microsoft word version (D) and Microsoft excel version (E) of calculation results are also added to the system.
Figure 4.17 The Microsoft word version of the calculation result under the single mode for seawater samples.
The program can also “remember” the latest values of the user input so the user does not have to re-enter all the values again, which solves problem (I). This design saves the user’s time and makes the program more convenient to use. Moreover, the input data can be separated by single space, tab space, comma (;), colon (:) or a combination of them (Figure 4.19). For the DOS version, the ID field MUST be within double quotes for space-separated data. This is not required for this web-based version. The input file can be also a comma-separated values (CSV) format file (Figure 4.20) or
Microsoft Excel file format (.xls). Both the input file and output file can be in any text file format, CSV file format, or Microsoft Excel file format. The user has more flexibility on the data format of the input data file and the output data file.

Figure 4.19 One of the sample file used for the batch mode calculation for freshwater samples. This file is in text format and the data can be separated by single space, tab space, comma(,), colon(:), and semicolon(;).

Figure 4.20 One of the sample file used for the batch mode calculation for freshwater samples. This file is in CSV file format.
Figure 4.21 One of the sample file used for the batch mode calculation for freshwater samples. This file is in Microsoft Excel (.xls) file format.

4.10 Summary

Testing the system with real users and its results and subsequent changes to the system are presented. A project conclusion and discussion of the future work will be addressed in the next chapter.
Chapter 5

Conclusion and future work

5.1 Project conclusion

This project implements a web-based version of CO2SYS program to replace the current DOS based version program. It has kept all the calculation methods and functionalities from the DOS version. All the user inputs can be entered in one window (“Input window”) instead of several small black and white DOS interface windows. The calculation results will be displayed on another window (“Output window”) if it is under the single input mode. All the inputs can be changed at any time before the calculations and recalculation can be done very easily. This new system presents an attractive and dynamic appearance to users.

Much more powerful error checking has been built into this system. The program also provides useful directions and guidance for the usage of the program. Sufficient helpful information is embedded with the inputs or options through hyperlinks for users. Typographical error information, which is separated from the corresponding reference in the documentation of the DOS version, is incorporated with the reference and available online in this system. There are 77 functions, 42 files and approximately 10,000 lines of code in this program.

5.2 Future work

During the development of this program, we have interacted with the developers of the DOS version CO2SYS program and other scientists who frequently use the current
DOS version program. They have showed tremendous interest on this new exciting web-based system. They all expect this new version can be installed on the server of CDIAC (Carbon Dioxide Information Analysis Center of Oak Ridge National Laboratory) and is ready to use as soon as possible. So the next step is to set up this system on the CDIAC server so it can be available online for the public use.

New quantifications of the constants used in this program have been implemented since the release of the DOS version program. So it is important to incorporate all these new changes to the system. For all the calculations for freshwater, ionic strength effect has not been considered for all the constants used for calculations. As shown in Figure 5.1, the program should be able to do calculation for freshwater samples with the consideration of ionic strength effect. More attractive visual effects can be added into the system with Macromedia Flash as well.

![Figure 5.1](image.png)

Figure 5.1 One of the sample files used for the calculation under the single mode for freshwater samples. Compared to Figure 4.19, this data set has taken out of the data of salinity, phosphate, and silicate, which are also zero for freshwater samples. The ionic strength of each sample with the red circle is added into each data set.
References


Appendix A

Cognitive walkthrough

**Main task 1**: Given two parameters (TA and TC), calculate the other two parameters (pH and $f\text{CO}_2$) for freshwater under single mode.

Action list for this task:

- a) Choose “Freshwater” as the water type
- b) Choose “Single mode” as the input mode
- c) Enter the input temperature
- d) Enter the input pressure
- e) Enter the output temperature
- f) Enter the output pressure
- g) Choose “$f\text{CO}_2$”
- h) Choose “TA, TC” as the input parameters
- i) Enter the values for TA and TC
- j) Run the execution
- k) Save the results
- l) Check the reference for each step above

a.) Choose “Freshwater” as the water type (Figure A1)

- *Will users be trying to produce whatever effect the action has?* Yes. It is the first option on the top of the screen and it is obvious to the user. There is also a guide to ask the user to make a choice first.

- *Will users be able to locate the control (button, menu, switch, etc.) for the action?* Yes.
➢ Will the control users found help them reach the goal they want? Yes. The user can clearly see they have chosen “Freshwater”.

➢ Will the interface provide understandable feedback for users after each action? Yes. The user knows they have set the water type to “Freshwater”.

![Image of CO2SYS program](image)

**Figure A1** The initial web page layout of data calculation for freshwater sample under the single mode.

b.) Choose “Single mode” as the input mode (Figure A1)

➢ Will users be trying to produce whatever effect the action has? Yes. There is a guide to direct the user to make a choice.

➢ Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes. It is clear on the menu.
Will the control users found help them reach the goal they want? The user can clearly see that they have chosen “Single mode”.

Will the interface provide understandable feedback for users after each action? Yes. The user knows they have set the input mode to “single mode”.

c.) Enter the input temperature (Figure A2)

- Will users be trying to produce whatever effect the action has? Yes. There is a guide for this input.
- Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes. It is clear with a default value.
- Will the control users found help them reach the goal they want? Yes. It is quite clear to the user.
- Will the interface provide understandable feedback for users after each action? Yes. The user can clearly see the value they just entered.

d.) Enter the input pressure (Figure A2)

- Will users be trying to produce whatever effect the action has? Yes. There is a guide for input.
- Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes. It is clear with a default value.
- Will the control users found help them reach the goal they want? Yes. It is quite clear to the user.
- Will the interface provide understandable feedback for users after each action? Yes. The user can clearly see the value they just entered.
e.) Enter the output temperature (Figure A2)

➢ Will users be trying to produce whatever effect the action has? Yes. There is a guide for this input.
- Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes. It is clear with a default value.

- Will the control users found help them reach the goal they want? Yes. It is quite clear to the user.

- Will the interface provide understandable feedback for users after each action? Yes. The user can clearly see the value they just entered.

f.) Enter the output pressure (Figure A2)

- Will users be trying to produce whatever effect the action has? Yes. There is a guide for input.

- Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes. It is clear with a default value.

- Will the control users found help them reach the goal they want? Yes. It is quite clear to the user.

- Will the interface provide understandable feedback for users after each action? Yes. The user can clearly see the value they just entered.

g.) Choose “fCO2” (Figure A2)

- Will users be trying to produce whatever effect the action has? There is guide for the option

- Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes. It is a menu option

- Will the control users found help them reach the goal they want? Yes. It is clear to the user for the options.
➢ Will the interface provide understandable feedback for users after each action? Yes. The user can see the option they just chose.

h.) Choose TA and TC as the two input parameters (Figure A2)

➢ Will users be trying to produce whatever effect the action has? Yes. There is a direction for the options

➢ Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes. It is a menu option

➢ Will the control users found help them reach the goal they want? It is clear to the users.

➢ Will the interface provide understandable feedback for users after each action? The user can see the option they have made.

i.) Enter the values for TA and TC (Figure A2)

➢ Will users be trying to produce whatever effect the action has? Yes. There is a direction for this input request.

➢ Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes. It is clear

➢ Will the control users found help them reach the goal they want? Yes.

➢ Will the interface provide understandable feedback for users after each action? Yes. The entered values are displayed.
### Calculation results

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Value</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>TA (umol/kg-SW)</td>
<td>2300.0</td>
<td>586.4</td>
<td>8.1542</td>
</tr>
<tr>
<td>TC (umol/kg-SW)</td>
<td>2100.0</td>
<td>3.3</td>
<td>1.7</td>
</tr>
<tr>
<td>pH (as HOH)</td>
<td>7.9091</td>
<td>-2.8</td>
<td>0.0021</td>
</tr>
<tr>
<td>pCO₂</td>
<td>0.0021</td>
<td>3.3</td>
<td>0.0022</td>
</tr>
<tr>
<td>Temp (°C)</td>
<td>20.00</td>
<td>5.00</td>
<td></td>
</tr>
<tr>
<td>Pressure (bar)</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Alkalinity (mg/L)</td>
<td>3.8</td>
<td>7.5</td>
<td></td>
</tr>
<tr>
<td>HCO₃⁻</td>
<td>1928.2</td>
<td>1940.9</td>
<td></td>
</tr>
<tr>
<td>CO₂⁻</td>
<td>152.8</td>
<td>143.7</td>
<td></td>
</tr>
<tr>
<td>CO₂(g)</td>
<td>19.0</td>
<td>15.4</td>
<td></td>
</tr>
<tr>
<td>Total Inorganic C</td>
<td>2100.0</td>
<td>2100.0</td>
<td></td>
</tr>
<tr>
<td>pCO₂ (atm)</td>
<td>586.4</td>
<td>295.3</td>
<td></td>
</tr>
<tr>
<td>pCO₂ (atm)</td>
<td>586.4</td>
<td>295.3</td>
<td></td>
</tr>
<tr>
<td>CO₂ in dry air at 1 atm (ppm)</td>
<td>602.0</td>
<td>299.1</td>
<td></td>
</tr>
<tr>
<td>Revelle factor</td>
<td>11.80</td>
<td>12.20</td>
<td></td>
</tr>
<tr>
<td>pH₁</td>
<td>5.903</td>
<td>6.054</td>
<td></td>
</tr>
<tr>
<td>pH₂</td>
<td>9.610</td>
<td>9.385</td>
<td></td>
</tr>
<tr>
<td>pOH</td>
<td>13.418</td>
<td>14.069</td>
<td></td>
</tr>
</tbody>
</table>

**Figure A3** Data calculation results in a single web page for freshwater samples under the single mode. This figure is the same as Figure 2.3.
j.) Run the execution (Figure A2)

- **Will users be trying to produce whatever effect the action has?** Yes. It clearly shows it will execute the calculations.
- **Will users be able to locate the control (button, menu, switch, etc.) for the action?** Yes.
- **Will the control users found help them reach the goal they want?** Yes.

*Will the interface provide understandable feedback for users after each action?* It will generate the results.

k.) Save the results (Figure A3)

- **Will users be trying to produce whatever effect the action has?** It is not obvious unless the user wants to save the result. Otherwise, the user can write down the result.
- **Will users be able to locate the control (button, menu, switch, etc.) for the action?** Yes. It is on the drop-down menu of the browser.
- **Will the control users found help them reach the goal they want?** Yes. It will ask for file directory for data storage.

*Will the interface provide understandable feedback for users after each action?* It will show the file for data storage.

l.) Check the reference for each step above (Figure A1, A2)

- **Will users be trying to produce whatever effect the action has?** It clearly shows a question mark for help
- **Will users be able to locate the control (button, menu, switch, etc.) for the action?** Yes.
Will the control users found help them reach the goal they want? Question mark (?) is commonsense for help

Will the interface provide understandable feedback for users after each action? Yes. It will give the hints for each step.

**Main task 2:** Given two parameters (TA and TC), calculate the other two parameters (pH and fCO₂) for freshwater under batch mode

Action lists for this task:

a) Choose “Freshwater” as the water type
b) Choose “Batch mode” as the input mode
c) Choose fCO₂
d) Choose TA and TC as the two input parameters
e) Choose the input data file from the local directory
f) Check the information of input data file format.
g) Enter the number of header lines
h) Enter the number of ID fields
i) Enter the designated value of missing value in the input data file
j) Choose the file directory for output result
k) Determine if a missing value flag in the output data is necessary
l) Run the execution
m) Check the reference for each step above
Figure A4 The complete web page layout of data calculation for freshwater under the batch mode. This figure is the same as Figure 2.4.
a.) Choose “Freshwater” as the water type (Figure A4)

- Will users be trying to produce whatever effect the action has? Yes. It is the first option on the top of the screen and it is obvious to the user. There is also a guide to ask the user to make a choice first.
- Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes.
- Will the control users found help them reach the goal they want? Yes. The user can clearly see they have chosen “Freshwater”.
- Will the interface provide understandable feedback for users after each action? Yes. The user knows they have set the water type to “Freshwater”.

b.) Choose “Batch mode” as the input mode (Figure A4)

- Will users be trying to produce whatever effect the action has? Yes. There is a guide to direct the user to make a choice.
- Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes. It is clear on the menu.
- Will the control users found help them reach the goal they want? The user can clearly see they have chosen “Single mode”.
- Will the interface provide understandable feedback for users after each action? Yes. The user knows they have set the input mode to “single mode”.

c.) Choose “fCO2” (Figure A4)

- Will users be trying to produce whatever effect the action has? There is guide for the option
• Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes. It is a menu option

• Will the control users found help them reach the goal they want? Yes. It is clear to the user for the options.

• Will the interface provide understandable feedback for users after each action? Yes. The user can see the option they just chose.

d.) Choose TA and TC as the two input parameters (Figure A4)

• Will users be trying to produce whatever effect the action has? Yes. There is a direction for the options

• Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes. It is a menu option

• Will the control users found help them reach the goal they want? It is clear to the users.

• Will the interface provide understandable feedback for users after each action? The user can see the option they have made.

e.) Choose the input data file from the local directory (Figure A4)

• Will users be trying to produce whatever effect the action has? Yes. There is a direction to lead the user for the input.

• Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes.

• Will the control users found help them reach the goal they want? The “browse” button clearly tells the purpose of this control.
- Will the interface provide understandable feedback for users after each action? Yes. It shows a file is chosen.

  f.) Check the information of input data file format (Figure A4)
  
  - Will users be trying to produce whatever effect the action has? Yes. The label clearly shows that.
  - Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes.
  - Will the interface provide understandable feedback for users after each action? Yes. The user will see the content of this linkage.

  g.) Enter the number of header lines (Figure A4)
  
  - Will users be trying to produce whatever effect the action has? Yes. There is a direction for the user to input.
  - Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes.
  - Will the control users found help them reach the goal they want? Yes. The user knows the headlines information is required.
  - Will the interface provide understandable feedback for users after each action? Yes. The user will see the information they just entered.

  h.) Enter the number of ID fields (Figure A4)
  
  - Will users be trying to produce whatever effect the action has? Yes. There is a direction for the user to input.
➢ Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes.

➢ Will the control users found help them reach the goal they want? Yes. The user knows the ID field information is required.

➢ Will the interface provide understandable feedback for users after each action? Yes. The user will see the information they just entered.

i.) Enter the designated value of missing value in the input data file (Figure A4)

➢ Will users be trying to produce whatever effect the action has? Yes. There is a direction for the user to input.

➢ Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes.

➢ Will the control users found help them reach the goal they want? Yes. The user knows missing value information is required.

➢ Will the interface provide understandable feedback for users after each action? Yes. The user will see the designated value for missing value.

j.) Choose the file directory for output result (Figure A4)

➢ Will users be trying to produce whatever effect the action has? Yes. There is a direction to lead the user for the input.

➢ Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes.

➢ Will the control users found help them reach the goal they want? The “browse” button clearly tells the purpose of this control.
Will the interface provide understandable feedback for users after each action? Yes. It shows a file is chosen.

k.) Determine if a missing value flag in the output data is necessary (Figure A4)

- Will users be trying to produce whatever effect the action has? Yes. There is a direction for this input.
- Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes.
- Will the control users found help them reach the goal they want? Yes. It is an option.
- Will the interface provide understandable feedback for users after each action? Yes. It will show the result clearly.

l.) Run the execution (Figure A4)

- Will users be trying to produce whatever effect the action has? Yes. It clearly shows it will execute the calculations.
- Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes.
- Will the control users found help them reach the goal they want? Yes.
- Will the interface provide understandable feedback for users after each action? No. It does not have clear feedback unless the user checks the output file in the directory. A message as a feedback should prompt up when the execution is done.

m.) Check the reference for each step above (Figure A4)
➢ Will users be trying to produce whatever effect the action has? It clearly show a question mark for help

➢ Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes.

➢ Will the control users found help them reach the goal they want? Question mark (?) is commonsense for help

➢ Will the interface provide understandable feedback for users after each action? Yes. It will give the hints for each step.

**Main task 3:** Given two parameters (TA and TC), calculate the other two parameters (pH and fCO$_2$) for seawater under single mode

Action lists for this task:

a.) Choose “Seawater” as the water type

b.) Choose “Single mode” as the input mode

c.) Choose the constants for calculations

d.) Choose a value for $K_{SO4}$

e.) Enter the value for salinity

f.) Enter the value for phosphate

g.) Enter the value for silicate

h.) Enter the input temperature

i.) Enter the input pressure

j.) Enter the output temperature

k.) Enter the output temperature
Figure A5 The complete web page layout of data calculations for seawater under the single mode. This figure is the same as Figure 2.5.

1.) Enter the output pressure

m.) Choose fCO2

n.) Choose TA and TC as the two input parameters
o.) Enter the values for TA and TC

p.) Run the execution

q.) Save the results

r.) Check the reference for each step above

a.) Choose “Seawater” as the water type (Figure A5)
   - Will users be trying to produce whatever effect the action has? Yes. It is the first option on the top of the screen and it is obvious to the user. There is also a guide to ask the user to make a choice first.
   - Will the control users found help them reach the goal they want? Yes. The user can clearly see they have chosen “Seawater”.
   - Will the interface provide understandable feedback for users after each action? Yes. The user knows they have set the water type to “Seawater”.

b.) Choose “Single mode” as the input mode (Figure A5)
   - Will users be trying to produce whatever effect the action has? Yes. There is a guide to direct the user to make a choice.
   - Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes. It is clear on the menu.
   - Will the control users found help them reach the goal they want? The user can clearly see they have chosen “Single mode”.

Will the interface provide understandable feedback for users after each action? Yes. The user knows they have set the input mode to “single mode”.

c.) Choose the constants for calculations (Figure A5)

- Will users be trying to produce whatever effect the action has? Yes. There is a direction for the user to choose.
- Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes.
- Will the control users found help them reach the goal they want? Yes. It clearly shows that.
- Will the interface provide understandable feedback for users after each action? Yes. The user can see the option they chose.

d.) Choose the value for $K_{SO_4}$ (Figure A5)

- Will users be trying to produce whatever effect the action has? Yes. There is a direction for the user to choose.
- Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes.
- Will the control users found help them reach the goal they want? Yes. It clearly shows that.
- Will the interface provide understandable feedback for users after each action? Yes. The user can see the option they chose.

e.) Choose the type of pH scale (Figure A5)
- Will users be trying to produce whatever effect the action has? Yes. There is a direction for the user to choose.

- Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes.

- Will the control users found help them reach the goal they want? Yes. It is clearly shows that.

- Will the interface provide understandable feedback for users after each action? Yes. The user can see the option they chose.

f.) Enter the value for salinity (Figure A5)

- Will users be trying to produce whatever effect the action has? Yes. There is a direction for the user to enter.

- Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes.

- Will the control users found help them reach the goal they want? Yes. It is clearly shows that.

- Will the interface provide understandable feedback for users after each action? Yes. The user can see the value they entered.

g.) Enter the value for phosphate (Figure A5)

- Will users be trying to produce whatever effect the action has? Yes. There is a direction for the user to enter.

- Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes.
Will the control users found help them reach the goal they want? Yes. It is clearly shows that.

Will the interface provide understandable feedback for users after each action? Yes. The user can see the value they entered.

h.) Enter the value for silicate (Figure A5)

- Will users be trying to produce whatever effect the action has? Yes. There is a direction for the user to enter.
- Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes.
- Will the control users found help them reach the goal they want? Yes. It is clearly shows that.
- Will the interface provide understandable feedback for users after each action? Yes. The user can see the value they entered.

i.) Enter the input temperature (Figure A5)

- Will users be trying to produce whatever effect the action has? Yes. There is a guide for this input.
- Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes. It is clear with a default value.
- Will the control users found help them reach the goal they want? Yes. It is quite clear to the user.
- Will the interface provide understandable feedback for users after each action? Yes. The user can clearly see the value they just entered.

j.) Enter the input pressure (Figure A5)
• Will users be trying to produce whatever effect the action has? Yes. There is a guide for input.

• Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes. It is clear with a default value.

• Will the control users found help them reach the goal they want? Yes. It is quite clear to the user.

• Will the interface provide understandable feedback for users after each action? Yes. The user can clearly see the value they just entered.

k.) Enter the output temperature (Figure A5)

• Will users be trying to produce whatever effect the action has? Yes. There is a guide for this input.

• Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes. It is clear with a default value.

• Will the control users found help them reach the goal they want? Yes. It is quite clear to the user.

• Will the interface provide understandable feedback for users after each action? Yes. The user can clearly see the value they just entered.

l.) Enter the output pressure (Figure A5)

• Will users be trying to produce whatever effect the action has? Yes. There is a guide for input.

• Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes. It is clear with a default value.
Will the control users found help them reach the goal they want? Yes. It is quite clear to the user.

Will the interface provide understandable feedback for users after each action? Yes. The user can clearly see the value they just entered.

m.) Choose “fCO2” (Figure A5)

- Will users be trying to produce whatever effect the action has? There is guide for the option
- Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes. It is a menu option
- Will the control users found help them reach the goal they want? Yes. It is clear to the user for the options.
- Will the interface provide understandable feedback for users after each action? Yes. The user can see the option they just chose.

n.) Choose TA and TC as the two input parameters (Figure A5)

- Will users be trying to produce whatever effect the action has? Yes. There is a direction for the options
- Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes. It is a menu option
- Will the control users found help them reach the goal they want? It is clear to the users.
- Will the interface provide understandable feedback for users after each action? The user can see the option they have made.

o.) Enter the values for TA and TC (Figure A5)
Will users be trying to produce whatever effect the action has? Yes. There is a direction for this input request.

Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes. It is clear.

Will the control users found help them reach the goal they want? Yes.

Will the interface provide understandable feedback for users after each action? Yes. The entered values are displayed.

p.) Run the execution (Figure A5)

Will users be trying to produce whatever effect the action has? Yes. It clearly shows it will execute the calculations.

Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes.

Will the control users found help them reach the goal they want? Yes.

Will the interface provide understandable feedback for users after each action? It will generate the results.

q.) Save the results (Figure A6)

Will users be trying to produce whatever effect the action has? It is not obvious unless the user wants to save the result. Otherwise, the user can write down the result.

Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes. It is on the drop-down menu of the browser.

Will the control users found help them reach the goal they want? Yes. It will ask for file directory for data storage.
- Will the interface provide understandable feedback for users after each action? It will show the file for data storage.

r.) Check the reference for each step above (Figure A5)

- Will users be trying to produce whatever effect the action has? It clearly show a question mark for help

- Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes.

- Will the control users found help them reach the goal they want? Question mark (?) is commonsense for help

- Will the interface provide understandable feedback for users after each action? Yes. It will give the hints for each step.
Figure A6 The web page displaying the calculation results for seawater sample under the single mode. This figure is the same as Figure 2.6.
Main task 4: Given two parameters (TA and TC), calculate the other two parameters (pH and $f_{\text{CO}_2}$) for seawater under batch mode

Action lists for this task:

a.) Choose “Seawater” as the water type
b.) Choose “Batch mode” as the input mode
c.) Choose the constants for calculation
d.) Choose a value for $K_{\text{SO}_4}$
e.) Choose the type of pH scale
f.) Choose $f_{\text{CO}_2}$
g.) Choose TA and TC as the two input parameters
h.) Choose input data file from the local directory
i.) Check the information of input data file format
j.) Enter the number of header lines
k.) Enter the number of ID fields
l.) Enter the designated value of missing value in the input data file
m.) Choose the file directory for output result
n.) Determine if a flag is necessary for a missing value in the output data file
o.) Run the execution
p.) Check the reference for each step above
Figure A7: The web page layout of data calculation for seawater sample under the batch mode. This figure is the same as Figure 2.7.
a.) Choose “Seawater” as the water type (Figure A7)

- *Will users be trying to produce whatever effect the action has?* Yes. It is the first option on the top of the screen and it is obvious to the user. There is also a guide to ask the user to make a choice first.

- *Will the control users found help them reach the goal they want?* Yes. The user can clearly see they have chosen “Seawater”.

- *Will the interface provide understandable feedback for users after each action?* Yes. The user knows they have set the water type to “Seawater”.

b.) Choose “Batch mode” as the input mode (Figure A7)

- *Will users be trying to produce whatever effect the action has?* Yes. There is a guide to direct the user to make a choice.

- *Will users be able to locate the control (button, menu, switch, etc.) for the action?* Yes. It is clear on the menu.

- *Will the control users found help them reach the goal they want?* The user can clearly see they have chosen “Batch mode”.

- *Will the interface provide understandable feedback for users after each action?* Yes. The user knows they have set the input mode to “batch mode”.

c.) Choose the constants for calculations (Figure A7)

- *Will users be trying to produce whatever effect the action has?* Yes. There is a direction for the user to choose.
- Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes.
- Will the control users found help them reach the goal they want? Yes. It is clearly shows that.
- Will the interface provide understandable feedback for users after each action? Yes. The user can see the option they chose.

d.) Choose the value for $K_{SO_4}$ (Figure A7)
- Will users be trying to produce whatever effect the action has? Yes. There is a direction for the user to choose.
- Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes.
- Will the control users found help them reach the goal they want? Yes. It is clearly shows that.
- Will the interface provide understandable feedback for users after each action? Yes. The user can see the option they chose.

e.) Choose the type of pH scale (Figure A7)
- Will users be trying to produce whatever effect the action has? Yes. There is a direction for the user to choose.
- Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes.
- Will the control users found help them reach the goal they want? Yes. It is clearly shows that.
Will the interface provide understandable feedback for users after each action? Yes. The user can see the option they chose.

f.) Choose “fCO2” (Figure A7)

- Will users be trying to produce whatever effect the action has? There is a guide for the option.
- Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes. It is a menu option.
- Will the interface provide understandable feedback for users after each action? Yes. The user can see the option they just chose.

g.) Choose TA and TC as the two input parameters (Figure A7)

- Will users be trying to produce whatever effect the action has? Yes. There is a direction for the options.
- Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes. It is a menu option.
- Will the control users found help them reach the goal they want? It is clear to the users.
- Will the interface provide understandable feedback for users after each action? The user can see the option they have made.

h.) Choose the input data file from the local directory (Figure A7)

- Will users be trying to produce whatever effect the action has? Yes. There is a direction to lead the user for the input.
Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes.

Will the control users found help them reach the goal they want? The “browse” button clearly tells the purpose of this control.

Will the interface provide understandable feedback for users after each action? Yes. It shows a file is chosen.

i.) Check the information of input data file format (Figure A7)

Will users be trying to produce whatever effect the action has? Yes. The label clearly shows that.

Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes.

Will the control users found help them reach the goal they want? Yes. It will tell the user the content of this linkage

Will the interface provide understandable feedback for users after each action? Yes. The user will see the content of the linkage.

j.) Enter the number of header lines (Figure A7)

Will users be trying to produce whatever effect the action has? Yes. There is a direction for the user to input.

Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes.

Will the control users found help them reach the goal they want? Yes. The user knows the headlines information is required.
k.) Enter the number of ID fields (Figure A7)

- **Will the interface provide understandable feedback for users after each action?** Yes. The user will see the information they just entered.

- **Will users be trying to produce whatever effect the action has?** Yes. There is a direction for the user to input.

- **Will users be able to locate the control (button, menu, switch, etc.) for the action?** Yes.

- **Will the control users found help them reach the goal they want?** Yes. The user knows the ID field information is required.

- **Will the interface provide understandable feedback for users after each action?** Yes. The user will see the information they just entered.

l.) Enter the designated value of missing value in the input data file (Figure A7)

- **Will users be trying to produce whatever effect the action has?** Yes. There is a direction for the user to input.

- **Will users be able to locate the control (button, menu, switch, etc.) for the action?** Yes.

- **Will the control users found help them reach the goal they want?** Yes. The user knows missing value information is required.

- **Will the interface provide understandable feedback for users after each action?** Yes. The user will see the designated value for missing value.

m.) Choose the file directory for output result (Figure A7)

- **Will users be trying to produce whatever effect the action has?** Yes. There is a direction to lead the user for the input.
➢ Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes.

➢ Will the control users found help them reach the goal they want? The “browse” button clearly tells the purpose of this control.

➢ Will the interface provide understandable feedback for users after each action? Yes. It shows a file is chosen.

n.) Determine if a flag is necessary for a missing value in the output data file (Figure A7)

➢ Will users be trying to produce whatever effect the action has? Yes. There is a direction for this input.

➢ Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes.

➢ Will the control users found help them reach the goal they want? Yes. It is an option.

➢ Will the interface provide understandable feedback for users after each action? Yes. It will show the result clearly.

o.) Run the execution (Figure A7)

➢ Will users be trying to produce whatever effect the action has? Yes. It clearly shows it will execute the calculations.

➢ Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes.

➢ Will the control users found help them reach the goal they want? Yes.
Will the interface provide understandable feedback for users after each action? No. It does not have clear feedback unless the user checks the output file in the directory. A message as a feedback should prompt up when the execution is done.

➤ Will users be trying to produce whatever effect the action has? It clearly show a question mark for help

➤ Will users be able to locate the control (button, menu, switch, etc.) for the action? Yes.

➤ Will the control users found help them reach the goal they want? Question mark (?) is commonsense for help

➤ Will the interface provide understandable feedback for users after each action? Yes. It will give the hints for each step.
Appendix B

Heuristic evaluation

**Main task 1**: Given two parameters (TA and TC), calculate the other two parameters (pH and $f$CO$_2$) for freshwater under single mode. The heuristic evaluation of main task one is based on the page layout in Figure B1 and the results of the evaluation are:

- Simple and natural dialog: the user can clearly see the guide and input the corresponding data.
- Speak the user’s language: the terminology is standard in water science area or can check the “?” for help.
- Minimize user memory load: the user can always see the previous input before the execution of calculation.
- Be consistent: the user can just follow the input order and it is simple.
- Provide feedback: the user can always see the input they made (Input error checking will be implemented on the next stage of the development).
- Provide clearly marked exits: the user can shut down the browser or go back to previous choices.
- Provide shortcuts: the user can jump between functions by making different choices due to the simplicity of the system.
- Good error messages: this functionality is not implemented at this stage and will be finished on the next stage of the development.
- Prevent errors: drop-down menu forces the user to choose either one of options. Errors in the manual inputs cannot be detected at this time due to incompletion of this functionality.
The web page layout of data calculations for freshwater sample under the single mode. This figure is the same as Figure 2.2 and A2.

The heuristic evaluation of the calculation result display page is based on Figure B2 and the results of the evaluation are:
Simple and natural dialog: this screen basically shows the calculation results and does not require any user input.

Speak the user's language: the terminology is common and clear to the user.

Minimize user memory load: in addition to the calculation results, it also shows the input and output conditions.

**Figure B2** The data calculation results display for freshwater under the single mode. The inputs and options are shown in Figure B1. This figure is the same as Figure 2.3 and A3.
> Be consistent: the user does not have to make any inputs for this screen.

> Provide feedback: this screen is actually a feedback for the action of clicking “Execute” button in Figure B1.

> Provide clearly marked exits: the user can shun down the browser or go back to previous screen with basic browser usage experience.

> Provide shortcuts: shortcuts are not necessary.

> Good error messages: this screen does not have any user input and error message does not apply.

> Prevent errors: this is simply a system output screen and errors should not occur.

Main task 2: Given two parameters (TA and TC), calculate the other two parameters (pH and $fCO_2$) for freshwater under batch mode. The heuristic evaluation of main task two is based on the page layout in Figure B3 and the results of the evaluation are:

> Simple and natural dialog: the user can just follow the guide to make the choices or inputs.

> Speak the user’s language: the terminology should be familiar to users as water chemists.

> Minimize user memory load: the user can clearly see the previous inputs before the execution.

> Be consistent: the user can easily follow the order of the blank.

> Provide feedback: the user can clearly see the previous inputs. But the feedback from the action of clicking “Execute” button is not implemented yet.
The screen layout of data calculation for freshwater under the batch mode. This figure is the same as Figure 2.4 and B4.

- Provide clearly marked exits: the user can go back to previous step by restarting the browser or make different choices to go back.
- Provide shortcuts: the user can get access to the input directly.
Good error messages: this functionality is not implemented at this stage and will be finished on the next stage of the development.

Prevent errors: drop-down menu forces the user to choose either one of options. Errors in the manual inputs cannot be detected at this time due to incompletion of this functionality.

Main task 3: Given two parameters (TA and TC), calculate the other two parameters (pH and $fCO_2$) for seawater under single mode. The heuristic evaluation of main task three is based on the page layout in Figure B4 and the results of the evaluation are:

- Simple and natural dialog: the user can clearly see the direction and input the data.
- Speak the user’s language: the user can understand the terminology easily or can check the “?” for help.
- Minimize user’s memory load: all the inputs are displayed on the same page.
- Be consistent: the user can input the data in order easily.
- Provide feedback: the user can always see the input they made (Input error checking will be implemented on the next stage of the development).
- Provide clearly marked exits: the user can shut down the browser or go back to previous choices.
- Provide shortcuts: the user can jump between functions by making different choices due to the simplicity of the system.
- Good error messages: this functionality is not implemented at this stage and will be finished on the next stage of the development.
- Prevent errors: drop-down menu forces the user to choose either one of options. Errors in the manual inputs cannot be detected at this time due to incompletion of this function.

Figure B4 The web page layout of data calculation for seawater under the single mode. This figure is the same as Figure 2.5 and A5.
**Figure B5** The data calculation results display for freshwater under the single mode. The inputs and options are shown in Figure B4. This figure is the same as Figure 2.6 and A6.
The heuristic evaluation of the calculation result display page is based on Figure B5 and the results of the evaluation are:

- Simple and natural dialog: this calculation output screen does not require any user input.
- Speak the user’s language: the terminology should be clear to the user.
- Minimize user memory load: in addition to the results, it also shows the input and output conditions.
- Be consistent: no input or extra actions are necessary for this screen.
- Provide feedback: this screen is actually a feedback for the action of clicking “Execute” button in previous screen.
- Provide clearly marked exits: the user can shun down the browser or go back to previous screen with basic browser usage experience.
- Provide shortcuts: shortcuts are not necessary.
- Good error messages: this screen does not have any user input and error message does not apply.
- Prevent errors: this is simply a system output screen and errors should not occur.
Figure B6 The screen layout of data calculation for freshwater under the batch mode. This figure is the same as Figure 2.7 and A7.
Main task 4: Given two parameters (TA and TC), calculate the other two parameters (pH and $f$CO$_2$) for seawater under batch mode. The heuristic evaluation of main task four is based on the page layout in Figure B6 and the results of the evaluation are:

- Simple and natural dialog: the program directs the user to make the inputs
- Speak the user’s language: the terminology should be familiar to users as water chemists
- Minimize user memory load: all the user inputs are displayed on the same screen
- Be consistent: the user can easily follow the order of the blank
- Provide feedback: the user can clearly see the previous inputs. But the feedback from the action “Execution” is not implemented yet.
- Provide clearly marked exits: the user can go back to previous step by restarting the browser or make different choices to go back.
- Provide shortcuts: the user can get access to the input directly.
- Good error messages: this functionality is not implemented at this stage and will be finished on the next stage of the development.
- Prevent errors: drop-down menu forces the user to choose either one of options. Errors in the manual inputs cannot be detected at this time due to incompletion of this function.
Appendix C

User testing

1. Background questionnaire for the testers

   Code #_______

   a.) How much are you familiar with the web (Circle one)?

      No familiarity  Somewhat familiar  Very familiar

      1   2   3   4   5

   b.) Do you fill in forms at online websites very often (Circle one)?

      Never  Sometimes  Always

      1   2   3   4   5

   c.) Have you ever used a web-based program (e.g., online calendar, online document management, etc)?

      Yes_______ No_______

   d.) Have you ever used an online calculator type program (e.g., loan calculator program, online expense program, etc)?

      Yes_______ No_______

   e.) Have you ever used a command line program (e.g., DOS, Unix, etc)?

      Yes_______ No_______

   f.) How much do you know about the DOS version of the CO2SYS program (Circle one)?

      No familiarity  Somewhat familiar  Very familiar

      1   2   3   4   5
g.) How much do you know about the water chemistry terms such as alkalinity, dissolved inorganic carbon, partial pressure of CO$_2$ gas, etc (Circle one)?

<table>
<thead>
<tr>
<th>No familiarity</th>
<th>Somewhat familiar</th>
<th>Very familiar</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>

2. Demographic information of the testers

Tester 1 is a postdoctoral researcher in the Department of Chemistry. He is very familiar with the web. Sometimes he fills in forms at online websites. He has used a web-based program and an online calculator type program. He has also used a command line program such as DOS, Unix, etc. He is very familiar with the DOS version of CO2SYS program and the water chemistry terms such as alkalinity, dissolved inorganic carbon, etc.

Tester 2 is an undergraduate student in the Department of Chemistry. He is very familiar with the web. Sometimes he fills in the forms at online websites. He has used a web-based program and an online calculator type program before. He has also used a command line program such as DOS, Unix, etc. He is very familiar with the DOS version of CO2SYS program and the water chemistry terms such as alkalinity, partial pressure of CO$_2$ gas, etc.

Tester 3 is a graduate student in the Department of Chemistry. He is somewhat familiar with the web. He often fills in forms at online websites. He has ever used a web-based program and an online calculator type program. He also used a command line program such as Unix, etc. He is not familiar with the DOS version of the CO2SYS
program and very familiar with the water chemistry terms such as alkalinity, dissolved inorganic carbon, etc.

Tester 4 is a graduate student in the Department of Chemistry. He is familiar with the web and sometimes fills in the forms at online websites. He has never used a web-based program or an online calculator type program. He has used a command line program such as DOS, etc. He is familiar with the DOS version of the CO2SYS program and water chemistry terms such as alkalinity, dissolved inorganic carbon, etc.

Tester 5 is a graduate student in the Department of Chemistry. He is familiar with the web and often fills in forms at online websites. He has ever used a web-based program and a command line program, but not any online calculator type program. He is somewhat familiar with the DOS version of the CO2SYS program and very familiar with the water chemistry terms such as dissolved inorganic carbon and partial pressure of CO₂, etc.

3. Test script

Hello. Welcome to Web Based CO2SYS system, a redesigned web based program replacing the old DOS version system. We’d like to take this opportunity to thank you for volunteering your time to help me test this new exciting system. I will be reading from this script to ensure consistency between all of our participants.

Now a little about this web based CO2SYS program. This program calculates relating parameters of carbon dioxide (CO₂) system in seawater and freshwater. The four parameters are total alkalinity (TA), total inorganic CO₂ (TC), pH and either fugacity
($f\text{CO}_2$) or partial pressure of CO$_2$ ($p\text{CO}_2$). Two of these parameters will be used to calculate the other two parameters under a set of input conditions and output conditions (such as temperature and pressure) chosen by the user. It can run in single input mode or in batch input mode and has many options for constants and parameters. Please note that this system is still in its development stage and that it may break down under certain circumstances. If that does happen, we will restart the program.

During the testing, a camera will be set up only with your permission and point to the screen. The camera will not point to your body. The purpose is to record how the system interacts with the user. It will be very useful for data analysis afterwards and system improvements. If you feel uncomfortable with this, we will stop the camera immediately.

Understand that this exercise is to test the product and its usability and in no way implies your abilities. If at any time you feel uncomfortable, please inform me and I will terminate the exercise immediately. Your opinion and ideas are important to me. Whenever possible, please speak your thoughts freely. Do not be concerned about offending me. If you forget to think aloud, I’ll remind you to keep talking.

As you’re working through this system, I won’t be able to provide help or answer questions. This is because I want to create the most realistic situation possible. Even though I won’t be able to answer your questions during the exercise, I will note your questions and answer them at the end of the exercise.

Do you have any questions before we start?
4. Subject information and consent form

Title: The development of web-based CO2SYS program

Project director:

Shigui Yuan
Department of Computer Science
The university of Montana
406-243-2883
Shigui.yuan@umontana.edu

Special instruction to the potential subject:

This consent form may contain words that you are not familiar with. If that is the case, please ask the person who gave you this form to explain them to you.

Purpose:

You are invited to participate in a research study conducted by Shigui Yuan from Department of Computer Science at the University of Montana. This web page based CO2SYS system replaces the old DOS based version program and enables to calculate two of the four parameters of the carbon dioxide (CO$_2$) system in seawater and in freshwater. The purpose of this study is to analyze how well this newly designed web-based system can resolve the user inputs and provides correct answer and meaningful information for the user.

Procedures:

If you decide to participate in this observational study, you will be asked to meet with Shigui Yuan in CS 419 of social science building for the program testing. During the
study session, you will be presented with a series of tasks that involve how to actually use this system. After completing these tasks, you will be requested to complete a questionnaire. I am estimating that the study session will take between 30-40 minutes.

During the testing, a video camera will point to the screen and record your activities on the computer screen. The purpose is to record how the system behaves instead of you. It will be very useful for future analyses. If you do not like it, we can stop that at any time.

Risks and discomforts:

It is important that you realize that during these studies, *it is the web based CO2SYS system that is being tested, not your data manipulation skills*. There is no time limit imposed, and it is not critical for you to finish any of the tasks. In many places, the instructions are purposefully abstract because our goal is to design intuitive interfaces that don't require complex training or manuals to operate. I just ask that you try to complete each task as best as you can. If you get really stuck on any particular task, and it appears that you are unable to continue, I will let you know when it’s okay to move on. Please remember that it's okay if you get stuck – this just means that this system needs more work! The risks associated with participating in this study are deemed to be minimal. That is, you will not be subjected to any pain or stress beyond that normally encountered in everyday life.

Benefits:

Although you may not benefit from taking part in this study, your participation will provide valuable feedbacks for our system to avoid the potential problems in the future real applications. Here we really appreciate your time and help.
Confidentiality:

Any data information and video clips that are obtained in connection with this study will be locked in a secure drawer with confidentiality. They can be accessed only by the researcher director and will be disclosed only with your permission or as required by law. Subject identities will be kept confidential by assigning each subject a code number. All data will be marked and identified using this code number. No personal information for any participant will be attached to the data.

Compensation for injury:

Although we believe that the risk of taking part in this study is minimal, the following liability statement is required in all University of Montana consent forms.

In the event that you are injured as a result of this research you should individually seek appropriate medical treatment. If the injury is caused by the negligence of the University or any of its employees, you may be entitled to reimbursement or compensation pursuant to the Comprehensive State Insurance Plan established by the Department of Administration under the authority of M.A.C., Title 2, Chapter 9. In the event of a claim for such injury, further information may be obtained from the University Claims representative or University legal Counsel.

Voluntary participation/withdrawal:

Your participation is voluntary. Your decision whether or not to participate will not affect your relationship with Shigui Yuan or The University of Montana in any way. If you decide to participate, you are free to withdraw your consent and discontinue participation at any time without penalty.
Questions:

If you have any questions, please feel free to contact his faculty advisor Yolanda Reimer, (406-243-4618) in the Computer Science Department at any time. If you have any questions regarding your rights as a participant, you may contact the IRB chair through the UM Research Office at 243-6670.

Subject’s statement of consent:

I have read the above description of this research study. I have been informed of the risks and benefits involved, and all my questions have been answered to my satisfaction. Furthermore, I have been assured that any questions I may have will also be answered by the research director.

I voluntarily agree to take part in this study and I may withdraw my consent at any time and discontinue participation without any penalty. I understand I will receive a copy of this consent form.

Subject’s name (in print)_________________________________

Signature __________________________________Date_____________
Special Addendum for video and audio taping consent

I understand that the research director will video and audio record my activities on the computer screen when I test the program. I can ask to stop this recording at any time when I do not like it. I am already notified that the data information and video and audio clips obtained from this study will be locked in a secure place only for data analysis and will be disclosed only with my permission or required by law. All the data will be kept with a code number and my personal information will be separated from this data.

By signing the consent below, I agree that the research director will use the video camera as a research tool for this program testing study.

Subject’s name (in print)_________________________________

Signature ___________________________________ Date_____________
5. Testing tasks

Scenario 1:

User A logs onto the CO2SYS web home page and wants to run some calculations for freshwater under the single mode. Below are the tasks he wants to finish.

**Task 1**: Try to set the input temperature to “-12”, the input pressure to “erfasf”, the output temperature to “45”, the output pressure to “10005”, fCO$_2$ as the CO$_2$ type and TA and TC as the input parameters. “afad” is for TA and “-1230” for TC. After all the inputs are done, go ahead to run the calculations.

**Task 2**: Write down the content of the prompted message, and go ahead to check the information and the valid data input range, if any, for the following: “input temperature”, “input pressure”, “output temperature”, “output pressure”, “fCO$_2$”, “TA”, “TC”.

**Task 3**: Based on the information you have gained from **Task 2**, try to fix any errors from your previous inputs and run the calculation again.

**Task 4**: Write down the values of pH, fCO$_2$ and pK$_W$ under the input condition and the output condition. Check the information for “Note” and write down the additional uncertainties due to constants at output conditions for freshwater.

**Task 5**: Now it becomes a little more challenging. Try to find the value of TC that can generate pH of 7.9883 under the same input conditions as in **Task 3** (i.e. same temperature, pressure, and TA). Write down the value of TC.
Scenario 2:

User B logs onto the CO2SYS web home page and wants to run some calculations for freshwater under the batch mode. Below are the tasks he wants to finish.

**Task 1:** Try to set fCO$_2$ as the CO$_2$ type, TA and TC as the input parameters, the input file to “C:\case1.inp”, header lines to “sfsdf”, ID fields to “1.5”, designated value of missing value to “adsfa”, missing flag to “Yes”, the output file to “C:\case1.out”, the other lines to the output file to “This is the additional information to the output file”, then run the calculation.

**Task 2:** Write down the content of the prompted message, and go ahead to check the information and the valid data input range, if any, for the following: “header lines”, “ID fields”, “designated value of missing value”.

**Task 3:** Try to reset the header lines to “2”, the ID fields to “5”, the designated value of missing value to “-5”, and rerun the calculations.

**Task 4:** Write down the content of the prompted message and reset the ID fields to “2”, and rerun the calculations.

**Task 5:** Write down the content of the prompted message, reset the designated value of missing value to “-7” and rerun the calculations.

**Task 6:** Try to check the content of the output file and write down the last line of data on the output file data window.
Scenario 3:

User C logs onto the CO2SYS web home page and wants to run some calculations for seawater under the single mode. Below are the tasks he wants to finish.

**Task 1:** Try to set the constant to “Roy et al., 1993”, the pH scale to “Total scale (mol/Kg-SW)”, $K_{SO4}$ to “Dickson”, salinity to “57”, phosphate to “-12”, silicate to “safasa”, the input temperature to “18”, the input pressure to “12”, the output temperature to “15”, the output pressure to “2.5”, CO$_2$ type to “pCO$_2$”, TC and pCO$_2$ as the input parameter, TC to “5”, pCO$_2$ to “6000”, and go ahead to run the calculations.

**Task 2:** Write down the content of the prompted message, and go ahead to check the information and the valid data input range, if any, for the following: “salinity”, “phosphate”, “silicate”.

**Task 3:** Try to reset salinity to “32”, phosphate to “5.8”, silicate to “7.5” and rerun the calculations.

**Task 4:** Write down the content of the prompted message, and fix the errors with valid input values and rerun the calculation.

**Task 5:** Write down the values for TA, pH$_{tot}$, $f_H$ under the input and output conditions. Check the information for “Revelle factor” and write down the normal value range of revelle factor for seawater. Moreover, check the information for “Omega” and write down the definition formula for “Omega”.

**Task 6:** Now it is a little challenging. Try to find the value of pCO$_2$ that can generate pH$_{tot}$ of 7.4658 under the same input conditions as above (i.e. same temperature and pressure, salinity, TC, etc). Write down the value of pCO$_2$. 

127
Scenario 4

User D logs onto the CO2SYS web home page and wants to run some calculations for seawater under the batch mode. Below are the tasks he wants to finish.

Task 1: Try to set the constant to “Roy et al., 1993”, the pH scale to “Total scale (mol/Kg-SW)”, K$_{SO4}$ to “Dickson”, “fCO2” as the CO$_2$ type, TA and TC as the input parameters, the input file to “C:\case2.inp”, header lines to “2”, ID fields to “2”, designated value of missing value to “-7”, missing flag to “Yes”, the output file to “C:\case2.out”, the other lines to the output file to “This is the additional information to the output file”, then run the calculation.

Task 2: Go ahead to check the contents of the output data file and write down the first line of the numeric calculation result.

Task 3: Now a little bit challenging. Try to reset the constants to “Geosecs constants (NBS scale)” and keep the other input as in Task 1, and rerun the calculation.

Task 4: Go ahead to check the contents of the output data file and write down the first line of the numeric calculation result. Then compare the data with the one from Task 2. Are they the same (Yes or No)?
**Scenario 5:**

User E has heard a lot about the new online CO2SYS program and wants to learn more about it. He logs onto the CO2SYS web home page and tries to find the answers for the following questions.

**Task 1:** What types of input mode does this program accept?

**Task 2:** What is the email address of “Douglas Wallace”, one of the original authors of the DOS version program?

**Task 3:** What are the typographical errors for the reference of “Clayton, T.D., and R.H. Byrne. 1993”?  

**Task 4:** What are the typographical errors on page 261 of the reference of “Goyet, C., and E. Peltzer. 1994”?

**Task 5:** What is the release date for version 1.03 of CO2SYS program?

**Task 6:** What is the definition formula for $H_{\text{free}}$, which is on the free pH scale?
6. Data capture form

Table C1 The data capture form for test scenario one.

<table>
<thead>
<tr>
<th>Code#</th>
<th>Date</th>
<th>Time</th>
<th>Task description</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Try to set the input temperature to “-12”, the input pressure to “erfasf”, the output temperature to “45”, the output pressure to “10005”, fCO$_2$ as the CO$_2$ type and TA and TC as the input parameters. “afad” is for TA and “-1230” for TC. After all the inputs are done, go ahead to run the calculations.</td>
<td></td>
</tr>
<tr>
<td>1/1</td>
<td></td>
<td></td>
<td>Write down the content of the prompted message, and go ahead to check the information and the valid data input range, if any, for the following: “input temperature”, “input pressure”, “output temperature”, “output pressure”, “fCO$_2$”, “TA”, “TC”.</td>
<td></td>
</tr>
<tr>
<td>1/2</td>
<td></td>
<td></td>
<td>Based on the information you have gained from <strong>Task 2</strong>, try to fix any errors from your previous inputs and run the calculation again.</td>
<td></td>
</tr>
<tr>
<td>1/3</td>
<td></td>
<td></td>
<td>Write down the values of pH, fCO$_2$ and pK$_W$ under the input condition and the output condition. Check the information for “Note” and write down the additional uncertainties due to constants at output conditions for freshwater.</td>
<td></td>
</tr>
<tr>
<td>1/4</td>
<td></td>
<td></td>
<td>Now it becomes a little more challenging. Try to find the value of TC that can generate pH of 7.9883 under the same input conditions as in Task 3 (i.e. same temperature, pressure, and TA). Write down the value of TC.</td>
<td></td>
</tr>
</tbody>
</table>
### Table C2: The data capture form for test scenario two.

<table>
<thead>
<tr>
<th>Scenario#/Task#</th>
<th>Task description</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>2/1</td>
<td>Try to set fCO₂ as the CO₂ type, TA and TC as the input parameters, the input file to “C:\case1.inp”, header lines to “sfsdf”, ID fields to “1.5”, designated value of missing value to “adsfa”, missing flag to “Yes”, the output file to “C:\case1.out”, the other lines to the output file to “This is the additional information to the output file”, then run the calculation.</td>
<td></td>
</tr>
<tr>
<td>2/2</td>
<td>Write down the content of the prompted message, and go ahead to check the information and the valid data input range, if any, for the following: “header lines”, “ID fields”, “designated value of missing value”.</td>
<td></td>
</tr>
<tr>
<td>2/3</td>
<td>Try to reset the header lines to “2”, the ID fields to “5”, the designated value of missing value to “-5”, and rerun the calculations.</td>
<td></td>
</tr>
<tr>
<td>2/4</td>
<td>Write down the content of the prompted message and reset the ID fields to “2”, and rerun the calculations.</td>
<td></td>
</tr>
<tr>
<td>2/5</td>
<td>Write down the content of the prompted message, reset the designated value of missing value to “-7” and rerun the calculations.</td>
<td></td>
</tr>
<tr>
<td>2/6</td>
<td>Try to check the content of the output file and write down the last line of data on the output file data window.</td>
<td></td>
</tr>
</tbody>
</table>
Table C3 The data capture form for test scenario three.

<table>
<thead>
<tr>
<th>Code#</th>
<th>Task#</th>
<th>Date</th>
<th>Task description</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>3/1</td>
<td></td>
<td></td>
<td>Try to set the constant to “Roy et al., 1993”, the pH scale to “Total scale (mol/Kg-SW)”, $K_{SO_4}$ to “Dickson”, salinity to “57”, phosphate to “-12”, silicate to “safasa”, the input temperature to “18”, the input pressure to “12”, the output temperature to “15”, the output pressure to “2.5”, CO$_2$ type to “pCO$_2$”, TC and pCO$_2$ as the input parameter, TC to “5”, pCO$_2$ to “6000”, and go ahead to run the calculations.</td>
<td></td>
</tr>
<tr>
<td>3/2</td>
<td></td>
<td></td>
<td>Write down the content of the prompted message, and go ahead to check the information and the valid data input range, if any, for the following: “salinity”, “phosphate”, “silicate”.</td>
<td></td>
</tr>
<tr>
<td>3/3</td>
<td></td>
<td></td>
<td>Try to reset salinity to “32”, phosphate to “5.8”, silicate to “7.5” and rerun the calculations.</td>
<td></td>
</tr>
<tr>
<td>3/4</td>
<td></td>
<td></td>
<td>Write down the content of the prompted message, and fix the errors with valid input values and rerun the calculation.</td>
<td></td>
</tr>
<tr>
<td>3/5</td>
<td></td>
<td></td>
<td>Write down the values for TA, pH$<em>{tot}$, $f</em>{H}$ under the input and output conditions. Check the information for “Revelle factor” and write down the normal value range of revelle factor for seawater. Moreover, check the information for “Omega” and write down the definition formula for “Omega”.</td>
<td></td>
</tr>
<tr>
<td>3/6</td>
<td></td>
<td></td>
<td>Now it is a little challenging. Try to find the value of $p$CO$<em>2$ that can generate pH$</em>{tot}$ of 7.4658 under the same input conditions as above (i.e. same temperature and pressure, salinity, TC, etc). Write down the value of $p$CO$_2$.</td>
<td></td>
</tr>
</tbody>
</table>
Table C4: The data capture form for test scenario four.

<table>
<thead>
<tr>
<th>Code#</th>
<th>Task#</th>
<th>Scenario/#</th>
<th>Task description</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>4/1</td>
<td></td>
<td></td>
<td>Try to set the constant to “Roy et al., 1993”, the pH scale to “Total scale (mol/Kg-SW)”, $K_{SO_4}$ to “Dickson”, “$fCO_2$” as the CO$_2$ type, TA and TC as the input parameters, the input file to “C:\case2.inp”, header lines to “2”, ID fields to “2”, designated value of missing value to “-7”, missing flag to “Yes”, the output file to “C:\case2.out”, the other lines to the output file to “This is the additional information to the output file”, then run the calculation.</td>
<td></td>
</tr>
<tr>
<td>4/2</td>
<td></td>
<td></td>
<td>Go ahead to check the contents of the output data file and write down the first line of the numeric calculation result.</td>
<td></td>
</tr>
<tr>
<td>4/3</td>
<td></td>
<td></td>
<td>Now a little bit challenging. Try to reset the constants to “Geosecs constants (NBS scale)” and keep the other input as in Task 1, and rerun the calculation.</td>
<td></td>
</tr>
<tr>
<td>4/4</td>
<td></td>
<td></td>
<td>Go ahead to check the contents of the output data file and write down the first line of the numeric calculation result. Then compare the data with the one from Task 2. Are they the same (Yes or No)?</td>
<td></td>
</tr>
</tbody>
</table>
Table C5 The data capture form for test scenario five.

<table>
<thead>
<tr>
<th>Code#</th>
<th>Date</th>
<th>Task description</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>5/1</td>
<td></td>
<td>What types of input mode does this program accept?</td>
<td></td>
</tr>
<tr>
<td>5/2</td>
<td></td>
<td>What is the email address of “Douglas Wallace”, one of the original authors of the DOS version program?</td>
<td></td>
</tr>
<tr>
<td>5/4</td>
<td></td>
<td>What are the typographical errors on page 261 of the reference of “Goyet, C., and E. Peltzer. 1994”?</td>
<td></td>
</tr>
<tr>
<td>5/5</td>
<td></td>
<td>What is the release date for version 1.03 of CO2SYS program?</td>
<td></td>
</tr>
<tr>
<td>5/6</td>
<td></td>
<td>What is the definition formula for $H_{\text{free}}$, which is on the free pH scale?</td>
<td></td>
</tr>
</tbody>
</table>
7. Post test questionnaire

The statements below all relate to the tasks you just completed. Please circle the score that most represents your response to the following statements (leave the answer blank if the statement is not applicable).

a.) I felt that the completions of the tasks overall was

<table>
<thead>
<tr>
<th>Very easy</th>
<th>Somewhat difficult</th>
<th>Very difficult</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>

b.) I felt that finding the help information on the web page

<table>
<thead>
<tr>
<th>Very easy</th>
<th>Somewhat difficult</th>
<th>Very difficult</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>

c.) I felt that changing the options or inputs and recalculating were

<table>
<thead>
<tr>
<th>Very easy</th>
<th>Somewhat difficult</th>
<th>Very difficult</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>

d.) I felt that obtaining the feedback from each action was

<table>
<thead>
<tr>
<th>Very easy</th>
<th>Somewhat difficult</th>
<th>Very difficult</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>

e.) I felt that determining and recovering from errors were

<table>
<thead>
<tr>
<th>Very easy</th>
<th>Somewhat difficult</th>
<th>Very difficult</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>

f.) I felt that understanding the input format information was

<table>
<thead>
<tr>
<th>Very easy</th>
<th>Somewhat difficult</th>
<th>Very difficult</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>
8. Post test interview

a.) What do you think about this system generally?

b.) What additional functionalities do you think should we add to the system?

c.) Do you have some ideas about the changes we should make to the interface?

d.) If you have ever experienced with the DOS based version of this CO2SYS program, can you compare and contrast these two versions of programs?

e.) Do you have any other suggestions?
9. The completeness of the testing for each user and possible comments

Most of users forgot to check the hyperlinks beside each input for the valid input information (e.g., task 1/2, 2/2, or 3/2, etc) because their background knowledge reminded them of that. I have asked them about this and they said they have neglected those accidentally and I showed them after the testing. They really like the ideas of the hyperlink beside each input. For more challenging tasks such as task 1/5, 3/6, most of users just kept changing one of the inputs instead of the input parameter and got frustrated. Finally they were told to move on to the next steps.

Table C6 Summarized are completeness of the user testing with possible comments.

<table>
<thead>
<tr>
<th>Task list</th>
<th>User 1</th>
<th>User 2</th>
<th>User 3</th>
<th>User 4</th>
<th>User 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/1</td>
<td>Want to click “Tab” key to jump between input windows (Complete)</td>
<td>Complete</td>
<td>Confused about the invalid data input at the beginning and finally get started (Complete)</td>
<td>Complete</td>
<td>Complete</td>
</tr>
<tr>
<td>1/2</td>
<td>Forget to check the hyperlinks beside each input for input information (Incomplete)</td>
<td>Complete</td>
<td>Complete</td>
<td>Forget to check the hyperlink for input information</td>
<td>Forget to check the hyperlink for input information</td>
</tr>
<tr>
<td>1/3</td>
<td>Completed</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
</tr>
<tr>
<td>1/4</td>
<td>Completed</td>
<td>Complete</td>
<td>Complete</td>
<td>Does not check “Note” for information (the rest is complete)</td>
<td>Complete</td>
</tr>
<tr>
<td>1/5</td>
<td>Try to change TC values several times to match the pH value, instead of choosing TA, pH to find the true TC (Incomplete due to time waste)</td>
<td>Try to change TC values several times to match the pH value, instead of choosing TA, pH to find the true TC (Incomplete due to time waste)</td>
<td>Change the input parameter to “TA, pH” to find the TC value (Complete)</td>
<td>Try several TC to get the right pHtot and incomplete due to time waste</td>
<td>TA, TC (5, 6000) still triggers the error message for invalid combination of TC and fCO2 (Complete after clicking “Execute” button twice).</td>
</tr>
<tr>
<td>Page</td>
<td>Task</td>
<td>Completed</td>
<td>Status</td>
<td>Status</td>
<td>Status</td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>-----------</td>
<td>--------</td>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>2/1</td>
<td>First does not understand what “CO₂ type” is, “Enter” key does not work after input the “input file” name (complete)</td>
<td>Go to “Home” link to get started (complete)</td>
<td>Go to “Home” link to get started (Complete)</td>
<td>Go to “Input mode” to choose “Batch” to start (Complete)</td>
<td>Clicking the option “Batch mode” to get started</td>
</tr>
<tr>
<td>2/2</td>
<td>Want to highlight the invalid input box (Complete)</td>
<td>Complete</td>
<td>Complete</td>
<td>Browser the file and complete</td>
<td>Type the file name and click the browser button for the file input (Complete)</td>
</tr>
<tr>
<td>2/3</td>
<td>Completed</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
</tr>
<tr>
<td>2/4</td>
<td>Completed</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
</tr>
<tr>
<td>2/5</td>
<td>Completed</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
</tr>
<tr>
<td>2/6</td>
<td>Completed</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
</tr>
<tr>
<td>3/1</td>
<td>Click “Home” link to get started, completed</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
</tr>
<tr>
<td>3/2</td>
<td>With valid input (TC = 2100, pCO₂ = 600), the error message still comes out (complete after clicking the “Execute” twice)</td>
<td>Error message still comes out even if the input values are valid (TC= 2235, pCO₂ = 657), complete after clicking the “Execute” button twice</td>
<td>Same problems found as user 1 and user 2 and finally complete by executing again</td>
<td>Do not close the error message on the display window and with valid inputs the results do not display (Complete after closing the window)</td>
<td>Error message still comes out even with valid input (complete after executing again)</td>
</tr>
<tr>
<td>3/3</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
</tr>
<tr>
<td>3/4</td>
<td>Try to set pCO₂ several times to match pHtot value, instead of TC, pHtot (Incomplete due to time waste)</td>
<td>Change pCO₂ value several times to get the pHtot, (Incomplete due to time waste)</td>
<td>Try to change pCO₂ several times to get the right pHtot (Incomplete due to time waste)</td>
<td>Try to change pCO₂ values several times to find the right pHtot (Incomplete due to time waste)</td>
<td>Complete</td>
</tr>
<tr>
<td>3/5</td>
<td>Complete</td>
<td>Complete</td>
<td>Calculation results are not centered in the display window</td>
<td>Forget to check the hyperlink for the information (Incomplete)</td>
<td>Complete</td>
</tr>
<tr>
<td>3/6</td>
<td>Try to set pCO₂ several times to match pHtot value, instead of TC, pHtot (Incomplete due to time waste)</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
</tr>
<tr>
<td>Page</td>
<td>Task</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
</tr>
<tr>
<td>------</td>
<td>----------------------------------------------------------------------</td>
<td>----------</td>
<td>----------</td>
<td>----------</td>
<td>----------</td>
</tr>
<tr>
<td>4/1</td>
<td>“Enter” key does not work for “file input” (complete by clicking the mouse outside of the input)</td>
<td>Complete</td>
<td>Use browser to choose the input file instead of manual typing (Complete)</td>
<td>Clicking the “Batch” option to get started</td>
<td>“Enter” key does not work and “C:\case2.inp” does not work and triggers error checking message (complete by re-enter different name)</td>
</tr>
<tr>
<td>4/2</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
</tr>
<tr>
<td>4/3</td>
<td>Does not want to retype all the inputs after just changing one input</td>
<td>Complete</td>
<td>Keep the other options when just change one option</td>
<td>Complete</td>
<td>Complete</td>
</tr>
<tr>
<td>4/4</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
</tr>
<tr>
<td>5/1</td>
<td>Find the answer from the “input mode” option</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
</tr>
<tr>
<td>5/2</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
</tr>
<tr>
<td>5/3</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
</tr>
<tr>
<td>5/4</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
</tr>
<tr>
<td>5/5</td>
<td>Try several links of the program before find the answer (Complete)</td>
<td>Complete</td>
<td>Check several sections of the program for the answer (complete)</td>
<td>Quickly find the answer from “Help” section</td>
<td>Check several links on the web to find the answer</td>
</tr>
<tr>
<td>5/6</td>
<td>Go to the prompted window to find the answer instead of “Help” section (Complete)</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
<td>Complete</td>
</tr>
</tbody>
</table>