Research and Implementation of Data Submission Technologies in Support of CReSIS Polar and Cyberinfrastructure Research Projects at Elizabeth City State University

Nadirah Cogbill, Matravia Seymore
Mentor: Jeff Wood   Principal Investigator: Dr. Linda B. Hayden
1704 Weeksville Road, Box 672
Elizabeth City, North Carolina 27909

Abstract- **The Center of Excellence in Remote Sensing Education and Research (CERSER) on the campus of Elizabeth City State University often has a requirement for the collection of information from various individuals and institutions in support of the CReSIS and Cyberinfrastructure research projects. This information may be of a general nature such as registrations or surveys, but may also contain sensitive information such as Social Security numbers.**

The 2009-2010 Multimedia Team implemented procedures for the development of online and print data collection methods. Three different approaches were utilized to construct general data submission forms and implement them on web servers utilized by CERSER. Web-based forms were created using Adobe Dreamweaver CS4, phpMyAdmin, Google Docs, and PHP. Printable forms were produced using Adobe LiveCycle Designer.

This investigation also researched the requirements and procedures required to develop a secure server for the collection of sensitive data. The implementation of a secure server will be taken up by a future research team.

I. INTRODUCTION

A. Overview

The Center of Excellence in Remote Sensing Education and Research (CERSER) is part of several national and international research projects involving undergraduate and graduate students involved in remote sensing research. The Center often has a need to gather information quickly from individuals both locally and remotely as well as the task of gathering data from remote sensors and research experiments.

B. Data Collection

A form allows users to input and submit data whether it is in print or via the Internet. This project focused on three areas of data collection: printed forms, local server developed forms, and remote server forms. Several applications and languages were used in this project to create various database formats.

Below are some of the basic elements of a form with brief explanations.

- Label – Identifies what a certain form element is used for
- Text Field - Used for gathering text data from the user
- Textarea – Employed to collect text data from the user when multiple line inputs are required or may be possible
- Check Box – Used with yes or no questions or multiple selections
- Radio Button – Much like Check boxes, but one selection can be made when used as a group
- List Menu - Display a large number of options and allow the user to select one or more
- Jump Menu - Drop down menus that redirect the viewer to a URL when an option is selected
- Hidden Field – For passwords and other data that may be secure
- Button – Utilized to submit or reset a form

II. GOOGLE DOCS

A. Overview

Google is a global technology leader that focuses on improving how people find and use information. Google holds the world's largest online index of websites and content. This website is free and can be accessed with an internet connection. Google was started by two college students in August of 1996. Confident in their product the two graduate students continued to work and perfect Google. Google Inc. was established in Menlo Park, California in September of 1998 thus establishing a legal site called google.com. Google was named after "googol," the term for the number one with one hundred zeros following.

Google docs is a free web-based word processor offered by Google. Within Google docs, users are able to make presentations, spreadsheets, and form applications. Google docs permits users to save and share their documents online with other users. Google docs are extremely effective due to their auto. Google docs allows users to return and edit their documents online thus eliminating the use of flash drives or other memory devices. Users can share the document with whoever they please and those who share the document are also able to edit the document. One of the shortcomings of

Google docs is that a user can potentially send their work to the wrong individual. Editing large documents in Google docs may also be a limitation. Storage capacity is shown to be 52 GB, but the upload limit is 1024 KB.

B. Form Development Procedure

Create an account with Google if an account does not already exist.

Click sign in on the top right hand side of the page. Login by providing your email along with password. Then click "Sign In".

A refreshed Google page will appear with more options. Click the last option "more" at the top left hand side. The "more" option will provide a drop down menu. Scroll down and click "Documents".

The Google docs window will appear. On the left hand side of the page, Click "Create new". A drop down will appear. Scroll down and select "Form".

A window will appear titled "Untitled Form". Just underneath that will be an option that allows you to fill in the "Question Title". Type in the question.

To enter the "Question Type" click on the drop down menu and a list of options will be presented. Choose the question type.

At the bottom of each question next to the "Done" button there is an option that allows for making the question required. Check the box and a red star will be presented with the question. This notifies the user that the question is required.

Below are some other question types

• Paragraph Text - Paragraph text allows for multi-line entries. This type of field is generally used when a longer answer is required.

• Multiple Choice/Radio buttons - Radio buttons are used for multiple choice questions requiring only one answer.

• Checkboxes - Checkboxes are used for a question with multiple answers that may apply. Checkboxes differ from Radio button in that they permit the selection of multiple answers.

• Choose from a List - Choose from a list is a drop down option where you there are multiple answers, but depending on the program, you can only choose one answer to the question.

At the very top of the page is a link that permits the developer to apply a template to make the form professional in appearance.

Once the information portion is complete, you can fill out the form in order to test that everything works. Once testing is done, click "Email this form" on the right hand side of the window.

A pop-up window will appear on the screen "send this form to others". Add in all the recipients you wish to send the form to. Once all recipients are included, select the "Send" button.

After a link to the form is emailed to the recipients, the recipients then click on the link and complete the form. Once they have completed the form, the submit button is selected which sends the information to a spreadsheet in Google docs.

The developer can then view the submitted data and the date/time the user submitted their data.

C. Prototype

The team created a form titled "CERSER Information Request". This form included both required and non-required fields. Presented inside of this form were two types of questions. The question types used were "Text" and "Choose from a List". The form asked the recipients general questions like their first, middle, and last name, date of birth, and both their cell and home phone numbers. The data was collected within one day of notifying the students and given to the office personnel in need of it. A screenshot of the form is shown in Figure 1.

Figure 1 CERSER Information Request

D. Analysis

Google doc forms are a quick and easily developed method of gathering non-sensitive data. No computer programming or scripting knowledge is needed to utilize this method of data submission. There are drawbacks to this technique though. The first being the inability to modify the structure of the form other than applying set templates. This is a minor drawback unless the developer needs to brand the form. The second drawback is the lack of security in these forms as they do not utilize secure protocols.

III. WEB BASED FORMS

A. Overview

A second method of gathering data from the web is to utilize a combination of tools implementing a form that can store gathered data into a variety of database formats. For this project the team utilized a combination of Dreamweaver, PHP: Hypertext Preprocessor (PHP), phpMyAdmin, and Structured Query Language (SQL). PHP and Dreamweaver file development took place on the development platform and was then uploaded to the server. SQL databases were created directly on the server utilizing the open source application phpMyAdmin.

B. Dreamweaver Overview

Adobe Dreamweaver is an application utilized in developing web sites. Dreamweaver is used on both the Macintosh and Windows operating systems and supports tools such as JavaScript, PHP, and CSS. The version utilized for this project is part of the Adobe CS4 Creative Suite which includes other Adobe software such as Illustrator, Photoshop, and Flash.

Dreamweaver is based on a What-You-See-Is-What-You-Get (WYSIWYG) design in that a developer can create code (HTML, JavaScript, PHP, etc…) while using an interface that appears much like a word processing application. A developer can view the code while they are developing in WYSIWYG mode or they can develop a page by coding directly in a separate view.

C. PHP Overview

PHP: Hypertext Preprocessor (PHP) is a general-purpose scripting language originally developed to create dynamic, personal home pages, thus the "PHP" acronym. It was originally created in 1995 and has been in constant development since then. PHP is a free software program released under the PHP License, instead of the GNU General Public License (GPL) as there are restrictions regarding the term PHP that conflict with the GNU GPL.

PHP is based strongly on the C, Java, and Perl languages. PHP is embedded into HTML and interpreted by a PHP processor module located on a server. Once the file is processed, it returns the generated HTML page back to the client (browser) hiding scripting, calculations, data calls, and other code implementations from the user.

D. MySql Overview

Structured Query Language (SQL) is a computer language designed to create and manipulate databases. This language is compatible with all major operating systems and includes data query and update, schema (tables, views, indexes, etc…) creation and modification, and data access regulation. SQL was developed in the early 1970's at IBM where it was used to manipulate and retrieve data in their database product, System R. The language was originally called SEQUEL, but was renamed SQL due to trademark conflicts with another company.

The most common operation in SQL is a query where the command "SELECT" is used to retrieve data from one or more tables without affecting the database. A query such as "SELECT *" would retrieve all columns of a specified table.

E. phpMyAdmin Overview

phpMyAdmin is a web based application created to perform administration tasks for the database application MySQL. phpMyAdmin was first developed in 1998 and became stable with a second set of developers in 2001. Through a browser interface, phpMyAdmin executes SQL statements from graphical interface selections. The ability to directly enter SQL statements is also available. The operations most used with phpMyAdmin are managing the components of the SQL software such as databases, tables, fields, indexes, and database permissions.

F. Procedure – Database Creation

SQL database creation takes place on the web site server using the phpMyAdmin application. Administrator's privileges are required for this operation on the server. Once the application is open in the browser window, proceed with the following steps:

Enter the name for the new database and select "Create"

The SQL window will display the statement used to create the database and the database will appear to the left with a notation that there are no tables present.

Enter the desired name for the table. There can be more than one table in a database. Enter the number of fields taking into account that one field will be needed as a key field. The key field is unique and is normally automatically incremented with each new record. Select "Go" once the entries are made.

The next step is to enter the parameters (field name, type, length, etc…). The field name is entered first.

The selection of a field type depends on the type of data entered and the length required to store it. INT fields will contain integers while VARCHAR and TEXT fields can contain any character. The difference being the amount of storage space allotted for each. (TEXT is larger than VARCHAR)

In the case of the "id" field, the name, type (INT), length (4), index (PRIMARY), and auto-increment selections were made. Primary make this a unique field that cannot be duplicated in any record. Autoincrement will increase each time a new record is added.

Enter the field name, type, and length values for the remaining fields in the table and select save.

The SQL creation statements are displayed along with the table structure of fields and modification selections. These selections allow you to modify, delete, or add fields as needed to the table.

The final step is to create the user privileges for the database by selecting the "Privileges" tab. Here the new user name, host (in this case localhost), and passwords are entered.

G. Procedure – HTML/PHP Creation

Creating forms in Dreamweaver is a two procedure process: 1. Create the form for data entry 2. Create the page to view data from the database. Dreamweaver utilizes the database created above in phpMyAdmin to create links between form fields and database fields. The steps shown below create a basic form utilizing PHP code to create connections to the database on the server.

1) Create a new PHP file in Dreamweaver by selecting File>New>PHP
2) Select Insert>Form>Form to insert the form container which will appear as a box with a dashed border
3) Next insert the desired fields into the form container. Ensure the "ID" field matches your database field name. The "Label" field will appear to the user to identify what must be entered into the field.
4) Select the inserted field to view the options palette for that field and the choices therein.
5) Continue inserting fields needed and modifying as needed
6) Insert a button and fill in the "ID", but leave the "Label" blank

7) The Next step is to establish a connection with the server. Select the "DATABASES" tab from the palettes on the right side of the window. Then select the "+" sign and MySQL.
8) Give the connection a name, place "localhost" in the MySQL server field, then fill in the user name and password that you created in phpMyAdmin.
9) Then press the "Select" button to retrieve a database.
10) Press "Test" to ensure the connection works.
11) Check that all entries are correct and press "OK"
12) Now select the "SERVER BEHAVIOR" tab from the palettes on the right side of the window and select the "+" symbol and then "Insert Record"
13) Select the connection previously connected and the name of the table should be shown and the values should match in the columns window.
14) In the "After inserting, go to:" field place the name of the file that the browser will go to after the submit button is pressed and select "OK".

The entry form is now complete with the PHP code being created by Dreamweaver. Save the file and close it.
1) The next step is to create a file to display the entries made in the previous form. Create a new PHP file in Dreamweaver by selecting File>New>PHP
2) Insert a table with two rows and the number of columns equal to the number of fields in the database. Fill in the cells in both rows with the names of the database fields.
3) Now select the "BINDINGS" tab from the palettes on the right side of the window and select the "+" symbol and then "Recordset (Query)"
4) Select the appropriate connection from the pull-down menu. If needed, you could filter and sort the recordset from this window.
5) Select test to view the database contents.
6) Turn Recordset1 down to view the fields
7) Select "ID" in row 2, column 1 of the table in the design window.
8) Drag the ID lightning icon from the recordset to ID in the table. ID will change to "{Recordset1.id}".
9) Do the same for the remaining cells in row two of the table.
10) This file will now show one record in the database. To show more first select the <TR> tag of the row containing the recordset.
11) Now select the "SERVER BEHAVIOR" tab from the palettes on the right side of the window and select the "+" symbol and then "Repeat Region".
12) You can select a set number of records to display or all of the records can be displayed. Then Select "OK".
13) Save the file and upload the two files created, along with the "connections" directory to the server.
14) The files are now ready for use.

H. Prototype – CERSER Upcoming Events

Adobe LiveCycle Designer…
To test the procedure in the last section a database was established to hold the upcoming events in the CERSER program. These events had not been appearing in some browsers due to their incompatibility with the XML files in use.
An entry form, with a password script from a third-party, was developed so that the events could be updated with any accessible browser. A database display file was also created to view the entered item and other database records.
Though the database worked as intended and is in use by the NIA web site, there are several enhancements that are needed and will be discussed in the Future Work section.

IV. ADOBE LIVECYCLE DESIGNER

A. Overview

Adobe LiveCycle Designer is an application that creates documents and forms and is a part of the Adobe Creative Suite. LiveCycle Designer is a designing tool for the productions of XML forms, which are presented as a PDF or HTML file. Documents and forms change when combined with data to adjust the content by including or excluding design elements, creating space, and numbering pages automatically. LiveCycle Designer makes graphical designing capabilities easier to create and organize templates.

LiveCycle Designer allows a developer to add or remove fields or whole sections of a form, show a help dialog box, add attachments, change a field's color, and create many more actions.

LiveCycle Designer started out as a part of PerForm. PerForm, an electronic forms software package created by Delrina (Canadian software). Delrina was bought by Symantec in 1995 and then Symantec was sold to JetForm in 1996.

B. Form Design Procedure

1) Open LiveCycle Designer
2) Select "New Form" from the opening window
3) The "New Form Assistant" is displayed to aid your first steps. Start by selecting "Blank Form" and press "Next".
4) The second screen of the assistant gives the choice of paper size, orientation, and number of pages.
5) Other page size choices can be made by selecting the Page Size pull-down menu. Select "Default" for now and then "Next"
6) The next screen displays the choices for how the data will be submitted. The "Email" selection allows the user to forward the data in an XML format. This form will be printed so only the print check box should be selected and then select "Finish".
7) You are now presented with a blank page. The hierarchal format of the document is presented on the left while object and formatting palettes are on the right.
8) The "Object Library" is located at the top right hand corner of the screen. It contains the common elements found on a form. These objects are added to the form by dragging them to the design view.
9) Underneath the Object Library are the formatting palettes. These include the Font, Paragraph, Object, Layout, and Accessibility palettes.
10) Located at the top left hand corner is the "Hierarchy" palette. The Hierarchy palette is a pictorial depiction of the objects in the Design View.

11) To create a title, drag the Text Object from the Object Library to the Design View. Enter and format your title.

12) To create a text field, drag the Text Field Object from the Object Library to the Design View. Edit the field label and expand/contract the field to the desired size.

13) The next item to add will be a checkbox. To insert a checkbox, drag the Checkbox Object from the Design Library to the Design View. Edit the field label. Including another Checkbox Object directly underneath the first Checkbox Field will result in a Checkbox Group.

14) To create a radio button, drag the Radio Button Object from the Object Library to the Design View. Edit the field label. By dragging another Radio Button from the Object Library directly underneath the already inserted Radio Button will create a Radio Group.

15) To create a radio group, drag another Radio Button from the Object Library directly underneath the already inserted Radio Button. Edit the field label.

16) Another item to be added will be a list box. To create a list box, drag the List Box Object to the Design View. List boxes provide users with a way to select one item. The options for this list are defined in the Object palette.

17) Once the form is completed, the developer can select the Preview PDF tab above the design window.

18) If the form is correct, select File>Publish to Repository to save the file in PDF format in the desired directory.

C. LiveCycle Prototypes

Two forms were revised and created with Adobe LiveCycle Design. The first was the Undergraduate Research Experience Evaluation Form. This is a form currently under development for students completing the ECSU URE program. It was designed to be completed inside Adobe Acrobat Reader and printed. This form is located at http://nia.ecsu.edu/ure_eval.pdf for development purposes only.

The second form was a revision of the CERSER Photo Release form. This form is utilized to obtain permission to use photos of individuals taken during CERSER activities. This form had previously been developed in Microsoft Word to be printed and completed by hand. The revision allows for the form information to be keyed in and then printed or printed and filled out.

D. Recommendations

Adobe LiveCycle Design is a powerful software package with capabilities far beyond what was experimented with during this project. The database capabilities are noted as being difficult for developers who are not skilled in working with data sources by the software documentation. Another feature, Emailing completed forms, utilizes XML formats which was beyond the scope of this project. LiveCycle is the best choice though when it comes to printable forms as it retains its formatting on most platforms.

V. HTTPS

A. Overview

Hypertext Transfer Protocol Secure (HTTPS) is a combination of the Hypertext Transfer Protocol with a secure protocol to provide encryption and secure identification of the server. Hypertext Transfer Protocol (HTTP) is the primary technology protocol on the Web that allows linking and browsing. HTTPS utilizes port 443 by default where standard HTTP uses port 80. HTTPS is designed to encrypt data transferred via the internet, thereby preventing eavesdropping and man-in-the-middle attacks.

Most implementations of the HTTPS protocol involve online purchasing or the exchange of private information. Accessing a secure server often requires some sort of registration, login, or purchase. Web browsers such as Internet Explorer and Firefox display a padlock icon to indicate that the website is secure and the URL address will begin with "https://" as shown in Figure 2.

Figure 2 Amazon.Com Security

The effectiveness of HTTPS can be limited by poor implementation of browser or server software or a lack of support for some algorithms. HTTPS secures data as it travels between the server and the client, once the data is decrypted at its destination; it is only as secure as the host computer.

The administrator of a web server must first create a public key certificate, an electronic document using a digital signature which identifies an individual or an institution, for the HTTPS web server. When a user accesses the secure site the certificate is downloaded thereby ensuring any further transactions between the server and the client will be encrypted. Certificates are obtained from a digital authority and range in cost from $10 to $1500 per year and must be renewed upon expiration.

Standard forms offer a quick and easy method of collecting data, but the limited security of the transmission prohibits the collection of sensitive information such as Social Security Numbers, birth dates, and family information. Properly installed, HTTPS can secure this information. More on the proposed usage of HTTPS will be discussed in the Future Work section of this paper.

VI. RESULTS

A. Forms

Data submission capabilities were researched during this project for online and print formats. The Adobe LiveCycle

Design was found to be the most capable when developing printable forms. The learning curve is steep to utilize the full capabilities of the software.

To gather a small amount of non-sensitive information quickly, the Google Docs form was the quickest to develop and deploy. Formatting of the form was limited and integration with program themes (CERSER, PolarGrid, Watershed Watch, etc…) would not be possible.

The development of forms utilizing the combination of Dreamweaver, PHP, and SQL is a versatile method, but requires a higher level of skill sets to complete. This type of form is versatile and can be coupled with a secure server to provide secure capabilities of data gathering. It can also be implemented within current framework styles of a site.

VII. FUTURE WORK

A. Secure Server

The next step in this development progression should be to build a secure server. This will allow currently used printable forms to collect sensitive data such as birth dates and Social Security numbers. Server platforms will need to be researched, server applications tested, and security certificates obtained to execute this step. An https server needs to be built to ensure security. In order for an https server to be installed a certificate needs to be accompanied with it. Along with a certificate, a certificate authority must also be provided. Certificate authorities can be found using Google.

B. Database Modifications for Upcoming Events

The currently developed database for the CERSER upcoming events will need to be enhanced to allow for the deletion and modification of individual records. These modifications will need to be performed from within Dreamweaver and be secured to prevent intrusion. At the present time anything that is written under upcoming events must be modified on the server performing modifications directly to the database. The file should be modified to include the capabilities to add, delete, or modify events as necessary. This modification will take place in Dreamweaver using the database tools built into the software.

C. File Submission

The submission of files is a significant part of data collecting. The uploading of files though PHP scripting should be researched along with safeguards to prevent incorrect file types from being uploaded.

REFERENCES

[1] "Google docs", Retrieved Jan 12, 2010 from the World Wide Web, http://docs.google.com/
[2] "Adobe LiveCycle Designer ES2", Retrieved Jan 12, 2010 from the World Wide Web, http://www.adobe.com/products/livecycle/designer/
[3] "Adobe Dreamweaver CS4", Retrieved Jan 12, 2010 from the World Wide Web, http://tryit.adobe.com/us/cs4/dreamweaver/
[4] "PHP", Retrieved Jan 14, 2010 from the World Wide Web, http://www.php.net/
[5] Lerdorf R., Tatroe K., Kaehms B., McGredy R., Programing PHP, Sebastopol, CA: O'Reily & Associates, Inc., 2002
[6] "mySQL", Retrieved Jan 14, 2010 from the World Wide Web, http://www.mysql.com/
[7] "phpMyAdmin", Retrieved Jan 14, 2010 from the World Wide Web, http://www.phpmyadmin.net/
[8] "HTML Source : HTML Tutorials", Retrieved Jan 15, 2010 from the World Wide Web, http://www.yourhtmlsource.com/forms/basicforms.html
[9] "https Protocol", Retrieved Jan 19, 2010 from the World Wide Web, http://msdn.microsoft.com/en-us/library/aa767735%28VS.85%29.aspx
[10] "Https: What it is and how it works", Retrieved Jan 19, 2010 from the World Wide Web, http://www.instantssl.com/ssl-certificateproducts/https.html
[11] Forta B., Sams Teach Yourself SQL in 10 Minutes (3rd Edition), Sams, April 10, 2004
[12] "SoftwareQuality.com", Retrieved Jan 19, 2010 from the World Wide Web, http://searchsoftwarequality.techtarget.com/sDefinition/0,,sid92_gci214006,00.html