Data Management Website Development for a Non-Profit Organization

Jeffery Hall
University of Virginia
jah5fv@virginia.edu

Momin Khan
University of Virginia
mbk6wm@virginia.edu

Peter Le
University of Virginia
pal4ka@virginia.edu

Travis Pennetti
University of Virginia
tjb7mb@virginia.edu

Isaac Reilly
University of Virginia
ilr9xy@virginia.edu

Sharika Smith
University of Virginia
sbs5t@virginia.edu

ABSTRACT

We created a data management system for the Center for Nonprofit excellence that handles membership data, financial data, and external feedback. The system was split into two applications: a survey tool for feedback survey generation and analysis, and a data dashboard to handle financial and membership data. Previous to development of the application, this information was processed manually in a time consuming process. Use of the created applications allowed for automation of the data processing, creating an average estimated speed up of 100%.

General Terms

Documentation, Software Engineering

Keywords

QueXF, QueXML, Highcharts, CakePHP, CNE, OpenID

1. INTRODUCTION

The Center for Non-Profit Excellence (CNE) is a non-profit organization located in downtown Charlottesville, Virginia. The CNE's mission is to help other local non-profit organizations to be successful using education and financial assistance.

One of the methods of education that the CNE employs is frequent training sessions to teach CNE members and non-members various lessons about business, marketing, and other skills that are required to run a successful non-profit organization. These trainings end with printed surveys that are filled out by the attendees for feedback and comments. The CNE saves these surveys and uses them to improve future trainings. The CNE also has a great deal of financial information to keep track of every year, including membership information, fundraising results, and overall financial data.

Currently, the CNE conducts their training surveys and financial data management mostly manually. Surveys are manually made from a template for each training, and completed surveys are manually read and tallied to obtain the results. The financial data is kept in various separated databases, and is manually compiled into a spreadsheet for analysis and for sharing with the CNE board of directors.

As part of the Service Learning Practicum (SLP) course, our team was tasked with improving the CNE's survey and financial data processes using an interactive web system. SLP is a two-semester Computer Science course at the University of Virginia, where students are divided into teams of six to help a local non-profit organization by developing software, web sites, mobile applications and more. Each team has biweekly meetings with a customer representative from the non-profit, as well as a professional mentor from the web development industry.

Over the course of the past two semesters, we have developed two web applications for the CNE: an application for automatically generating, processing, and analyzing surveys for the CNE's trainings ("Survey Tool"), and an application for automating the storage, analysis, and sharing of financial data for the CNE board of directors meetings ("Data Dashboard"). Working closely with the CNE during the development process has helped us create tools that will be highly useful to the CNE's mission.

2. BACKGROUND

The chief focus of both the survey tool and the data dashboard is ease of use. The survey tool will be used predominantly by one CNE employee. The user is not expected to have a very high level of technical expertise. The dashboard will be viewed by all of the CNE board members at their monthly meetings. As such, the results must be easily viewed and should display relatively quickly. The website utilizes CakePHP (http://cakephp.org/), which operates using a model-view-controller schematic. QueXML (http://quexml.sourceforge.net/) is used to auto-generate the surveys. It can define different question types, such as multiple choice or free form, as well as what constitutes an acceptable answer. QueXF (http://quexf.sourceforge.net/) is used to analyze the results by using a banding file to register responses on scanned PDF of returned surveys. The survey tool and data dashboard both utilize Highcharts (http://www.highcharts.com/), which uses JavaScript to dynamically create interactive charts on the website.

3. RELATED WORK

Our survey tool uses optical recognition technology that is commonly used in scanning software. Multiple choice questionnaires such as those used in educational exams recognize entries that have been filled by the user and tracks the data over time to generate statistics about the data, similar to our survey tool. However, many of these technologies require special surveys that must be purchased from a company such as Scantron and scanned with their scanning tools, which are costly. Additionally, the survey forms used are designed for general use, this means that the actual survey has to be printed separately from the answer form and the answer form may include more entries than necessary.

More scanning software exists that performs a function similar to QueXF; however, these scanning tools are inferior either in cost, functionality, or available documentation. QueXF is open source and allows for customizable forms.

4. SYSTEM DESIGN

Though the Survey Tool and Data Dashboard are similar in function, they are designed for separate, unrelated purposes, and are accessed by different users at the CNE. No data is exchanged between the two tools. As such, for the sake of simplicity the tools were separated into two separate CakePHP applications, each with its own database, models, and views. Even so, certain design decisions were replicated across both tools.

Both the Survey Tool and Data Dashboard use Twitter Bootstrap[7] for their user interfaces. Twitter Bootstrap was chosen because of the wide array of features available, combined with the existence of a CakePHP Bootstrap plugin[6] simplifying integration with the system. The Bootstrap theme chosen was picked so that the site's appearance would mimic that of the CNE external website[2]. Charts, meanwhile, are generated using HighCharts[3], which like Bootstrap was chosen due to the large number of features and availability of a CakePHP plugin[4].

4.1 Survey Tool

4.1.1 System Architecture

In addition to a controller that manages account information (called UsersController), the CNE Survey Tool uses two models (along with associated views and controllers) called SurveyGroup and Survey. A SurveyGroup, shown in the views as *Training*, contains information about a given training session, such as date, location, and attendance; this is where the three editable short response questions are set. A *Survey* contains the saved responses for each individual survey taken. SurveyGroups are associated with Surveys through the CakePHP 'hasMany' relationship; conversely, each Survey associates with a SurveyGroup through the 'belongsTo' relationship.

As this system will only be used by employees of the CNE, the survey tool uses the OpenID protocol[5] to restrict access only to those with CNE email accounts. OpenID authentication is handled through a third party CakePHP component[1]. Each entry in the users table contains an OpenID field that is used to match a Survey Tool account to an authenticated OpenID account, which is then logged in using the CakePHP Auth component.

Surveys are created using the tool QueXML[9]. This tool takes an XML template file and information stored in the SurveyGroup and creates a PDF of the survey to be printed. At the same time, it creates a banding file denoting the response areas, which are used when the surveys are scanned into the system.

The completed surveys are analyzed using another tool related to QueXML: QueXF[8]. QueXF uses the banding file created alongside the PDF to look for markings in the appropriate response areas. It outputs the results in a CSV containing responses for each survey completed for a given training.

Query criteria are set through one of two ways. HTTP Post data can be entered using a form that allows for queries based off of all SurveyGroup Fields. An HTTP Get request can also be used to view the analysis for a single SurveyGroup should users choose to do so.

4.1.2 Design Decisions

Numerous tools exist for generation and analysis of surveys. QueXF and QueXML were chosen because of available documentation. Frequent communication with the developer was also a necessity, making the QueXF suite an ideal choice since the developer was still actively working on the project.

Even with support from the developer, difficulties arose when integrating QueXF into the survey tool. Numerous configuration variables needed to be set to accommodate the paper size used in the United States. In addition, the banding for the "training sources" question, which allows for multiple answers to be chosen, had to be written as many individual questions in order to display all answers correctly in the CSV. Displaying these multiple individual questions horizontally required additional changes to the survey template file and QueXF's configuration.

Rare yet persistent issues with the OpenID component led to the requirement that either a different tool be used or another way to login be available for when OpenID fails to authenticate. As not using the component would require accessing OpenID directly and would likely create additional unintended behavior, a password is added to each account that can be used by users to login should OpenID fail.

4.2 Data Dashboard

4.2.1 System Architecture

The Data Dashboard organizes financial and membership data in a format similar to the Survey Tool: a FinancialReport or MembershipReport model stores the report date and associates with multiple FinancialRow or MembershipRow models using the hasMany relationship. Each row model contains one row of information, such as an expense or an organization, for the associated financial or membership report, respectively.

Fundraising data is stored in a similar format. A FundraisingYear model stores the fundraising goals for a given year. FundraisingMonth models then store the amount received and outstanding for a given month. Like FinancialReport and FinancialRow, FundraisingYear is associated with FundraisingMonth using the hasMany relationship.

Unlike the survey tool, users login to the dashboard using an email and password combination. In order to view data, a user account must be activated by an administrator. An email notification is sent to all administrators upon creation of an account. This is done using the built-in CakePHP e-mail component.

As the dashboard is replacing an older process already in use, the system creates graphs that mimic those on the current dashboard. Analysis is always done using the most recent reports.

4.2.2 Design Decisions

The organization of financial and membership data was chosen because of the varying amount of data that can be added to the system each month. The sources of income or expenses could change at any time, making the table structure mandatory. Similarly, the number of members frequently fluctuates.

Financial data was organized in the same structure because of the desire to keep track of all data on a month by month basis. Though only the most recent fundraising data is used when generating graphs, the system still stores amounts for every month individually. These can be viewed by any activated user account.

5. PROCEDURE

Since the Survey Tool and Data Dashboard are separate web applications they require separate procedures for operation.

5.1 Survey Tool

Logging In

You must log in to the site using a '@thecne.org' e-mail address by clicking on the "Please Log In" button in the upper navigation bar. After this, you will be redirected to the Google accounts login page where you must select your '@thecne.org' email address (same as above).
In the case that Google accounts does not authenticate, you will be redirected to a page where you may manually enter your username and password.

Creating an Account

Use the procedure above to attempt a log in. An account will be created for your '@thecne.org' e-mail address and you will be redirected to their profile page where you may change your profile password. The profile password is used in the case that you are not authenticated properly as identified in the section 'Logging In'.

Creating a Training – Administrator Only

As an administrator who is logged in, click the "Trainings" tab in the upper navigation bar. After being redirected the administrator can click "New Training" under the actions panel. This will direct you to a form where you can enter in the training's name, type, location, instructor, and date. You can also specify three free response questions for the survey, or leave them as the default text. You may click "Submit" to save the training.

5.1.4Generating the Survey – Administrator Only

If a training has been created, then it will be listed on the "Trainings" page where new trainings are displayed. To generate a survey, you can click on the name of the survey you would like to generate and then click "Generate PDF" in the actions panel. This will download a PDF of the survey into your browser downloads folder. You may open the downloaded PDF file and print out the desired number of surveys to be completed using the following print settings: Monochromatic color, 300 DPI resolution, letter-sized page (8.5"x11"), and print format set to fit to the page size.

Completing the Surveys

Distribute the surveys to be filled and collect them to be scanned once they have been completed.

Scanning the Surveys – Administrator Only

The administrator should scan the surveys with the following scanner settings: Monochromatic color, 300 DPI resolution, letter-sized page (8.5"x11"), and most importantly scanned to a PDF.

After all surveys have been scanned you can zip all of the files into a '.zip' folder. The zipped folder can be uploaded on the Surveys' view page via the "Upload Complete Surveys" button. This will redirect you to a form where you can select the zip file and upload it. Once you upload the surveys, it takes time for the system to analyze them and determine the results. Please wait 20 minutes for the surveys to be completed.

After waiting, click the "Upload Results CSV" button. Select the CSV containing the survey results that you previously downloaded, and also enter in the training attendance. Then hit submit. The results and attendance will be added to the survey tool.

Survey Analysis

There are two ways the user can view survey analysis. To view analysis for a single training, you navigate to that training's view page and click "View Analysis". For a more comprehensive search, you can click "Analysis" in the upper navigation bar. There, you can limit their trainings by various criteria and click "Analyze".

The graphs are presented in an "Accordion" Format. You can click the tab of the graph they want to see to expand it and view the graph. You can click again to close it if desired. You can hover her mouse over a data point in the graph to get more information about the data point.

If you wish to print a graph, click the print icon in the upper right corner of the graph. To export it to a PDF or JPEG, click the export button (denoted by an arrow pointing down to a line) and then select the desired file type.

5.2 Data Dashboard

Logging In

To log into the data dashboard, you can click the "Log In" Button in the upper navigation bar. Then enter their e-mail and password.

Creating an Account

To create an account, follow the above steps to navigate to the login page. Then click the link underneath the username and password text fields labeled "create one." This will prompt the user to supply a username and password. For their username, they should enter the e-mail they want to associate with this account.

After you create an account, wait for approval from the administrator. Please give the administrator time. Once you are approved, you may login via the above steps.

<u>*NOTE*</u>: The Username must be an email address. Putting in a non-email will not create the user.

New Accounts for the Administrator

When you create an account, you will receive an email notifying you of this. The email will contain a link to the user's profile. Follow the link and click on "Activate User" in the actions panel. This will activate the user so they can view the data. To add the user as an admin, select "Admin" it is not necessary for an Admin to also be activated; simply being an admin will also activate the user.

Viewing Data and Analysis

To view the data, click the "Financial Reports", "Membership Data" or "Fundraising Data" Tabs in the upper navigation bar. This will take you to a list of the appropriate reports. Each report contains information stored in slightly different ways. However, you navigate the pages in the same way. Click the "View" link lined up with a given report to view more details. You may click one of the field names at the top of the list to sort by that field.

Financial Reports are uploaded every month, as are Membership Reports. The fundraising page shows the goals for a given year. The details page for a year shows the specific amounts currently received and outstanding for that year.

To view the analysis of the data, click the "View Analysis" tab in the upper navigation bar. This will take you to a screen containing the eight data dashboard graphs. Hover your mouse over a data point for specific information about that point.

If you wish to print a graph, click the print icon in the upper right corner of the graph. To export it to a PDF or JPEG, click the export button (denoted by an arrow pointing down to a line) and then select the desired file type.

Financial Data

To add Financial Data, navigate to the financial reports tab and click "New Financial Report". The site will then prompt you for a "Financial Report" and a "Report Date". For "Financial Report" please export your quickbooks financial data for the month to a CSV and select that file. For "Report Date", choose the month and year the report is for. Then hit submit.

<u>*NOTE*</u>: Data can only be entered by the administrators!

Uploading a CSV will automatically create the report and all of the associated expenses and incomes. If you find that the CSV did not contain all of the expenses and incomes, click "New Financial Row" and manually enter in the data. Be sure to set the "Financial Report" field to the correct report!

If you find that a portion of the report was added incorrectly, view the details page for that report, then click "Edit" next to the row you want to edit. Then fill out the form.

To delete a report, click the "Delete" link next to the report when in the list view, or go to the report's details page and click "Delete Financial Report" underneath the Actions panel.

<u>*NOTE*</u>: Financial Data should be uploaded to the site every month

Membership Data

The Membership Data is uploaded and edited in the same way as the financial data. The only difference is the source of the data. Please export your membership data from the database to a CSV, and upload that file.

Fundraising Data

At the beginning of each year, create a new fundraising year by clicking the link marked "New Fundraising Year" in the actions panel after clicking on the "Fundraising Data" tab in the upper navigation bar. You must then enter in the year and fundraising goals for that year.

Each month, navigate to the fundraising page and click "New Fundraising Month". Then fill out the form. Be sure to select the appropriate year! For "Fundraising Year", all years are shown as the first day of that year, for example "2013-01-01". Do not worry about month or day; year is the only relevant information here.

For amounts received and outstanding, enter in the cumulative amount for the entire year up to that point.

Data can be edited and deleted in the same fashion as membership and financial data describe above.

6. RESULTS

The program suite resulted in improved workflow and time management at the CNE. It reduced the time spent on tabulation and creation of survey, financial, and membership results by automating the process, resulting in an estimated reduction in time spent on survey creation from 20 hours to 10 hours and membership and financial data tabulation from 20 hours to 10 hours. This yielded an overall improvement of 100%. The error rate is also decreased. This reduction of time allows for feedback to be given to speakers much closer to their presentation and results in a better correlation of survey feedback to presentation material. Similarly, the reduction in labor required in producing financial and membership data allows for the information to be reviewed more often than once per financial quarter and allows for historical analysis over previous months or years. This more frequent analysis combined with the addition of historical analysis and the presentation of membership and financial data side-by-side should allow for more informed decisions to be made in regards to membership drives and financial initiatives by the organization. In future endeavors, this system also allows for the CNE's functions to expand to more clients, as it will require none or negligible increases in time for the system to process the additional membership data, financial data, and surveys, meaning that the system will allow the CNE to offer more trainings and better reach a wider audience.

7. CONCLUSION

We designed a web application to meet a need of the Center for Nonprofit Excellence. Previously, all data was handled separately and manually. The CakePHP system allows the CNE to handle their survey, financial, and membership data in two similar websites (uploading of CSVs and data display style). Data does not need to be manually entered for each survey and instead can be scanned once and output into a CSV file. A standardized approach to handling centralized data benefits the CNE, freeing employees from easily mismanaging data where no standard operating protocol is established. The systems fulfill all the requirements the CNE had put forth for a data management solution.

8. FUTURE WORK

Both the survey tool and the data dashboard have several features that could benefit from further development and several features that could be implemented to provide additional functionality to the system. The survey tool would benefit from complete automation where the user is able to print out surveys, fill them out as needed, scan the surveys, and view the resulting data in the form of a graph without having to handle the CSV (Comma Separated Values) files. The optical recognition software QueXF could be modified to be more tolerant of various printing formats and provide native support for a horizontal, multiple-selection question. QueXF could be optimized to recognize the only survey format that is used and ignore code that deals with various survey layouts, character recognition, and other features of QueXF which are not used in our surveys. Removing unused features from QueXF may also optimize the analysis and data retrievable performed by QueXF. The surveys could also be modified to include more data collecting elements such as additional questions or the ability to collect the user data like e-mail, age, etc. The style of the surveys could also be improved to be more appealing to the user including elements such as bubble-fill answers, uniform section color, and elements that identify the survey with the CNE (Center for Nonprofit Excellence). Additional functionality for the survey tool could include a system which tracks the history of scanned surveys by saving the scanned PDF (Portable Document Format) files for some predetermined amount of time before deleting them from the database. This would allow the CNE to review scanned surveys in the case that they were scanned improperly or an error occurred during the optical recognition process.

In the future the system can be more complete with more streamlined processes, interface, and better analysis tools. The survey tool still requires an administrator to go through the back end to complete the data entry process, and it should ideally be in the front end to hide the lower layers of the application. The interface can be cleaner and more conducive to new users. Perhaps a more robust instruction system built into the website can be implemented. Finally, more ways to analyze data outside of what the CNE already had could be devised to further bolster the utility of the data.

8. ACKNOWLEDGMENTS

The Center for Nonprofit Excellence Data Dashboard Group in the Service Learning Practicum course at the University Of Virginia Department Of Computer Science would like to give a special thanks to Kara Williams and Valerie Washington from the Center for Nonprofit Excellence in Charlottesville, VA and Stephen Johnson for his mentoring assistance.

9. REFERENCES

[1] Cakebaker. 2012. OpenID Component for Cake PHP. Retrieved September 20, 2012 from http://code.42dh.com/openid/

[2] Center for Nonprofit Excellence. 2012. Center for Nonprofit Excellence. Retrieved September 20, 2012 from http://thecne.org/

[3] destinydriven. 2012. CakePHP High Charts Plugin. Retrieved October 15, 2012 from https://github.com/destinydriven/cakephp-high-charts-plugin

[4] Highsoft Solutions. 2012. Highcharts JS. Retrieved October 15, 2012 from http://www.highcharts.com/

[5] OpenID Foundation. 2012. The OpenID Foundation Website. Retrieved Spetember 20, 2012 from http://openid.net/

[6] slywalker. 2012. TwitterBootstrap – Plugin for CakePHP2 to use Bootstrap, from Twitter (CSS JS Framework) Retrieved October 15, 2012 from https://github.com/slywalker/TwitterBootstrap

[7] Twitter. 2012. Bootstrap. Retrieved October 15, 2012 from http://twitter.github.io/bootstrap/

[8] Australian Consortium for Social and Political Research Incorporated. 2012. Quexf, An Open Source, web based paper form verification and data entry system. Retrieved Septeber 20, 2012 from http://quexf.sourceforge.net/

[9] Australian Consortium for Social and Political Research Incorporated. 2012. QueXML, an Open Source, XML, multi-node questionnaire description toolkit. Retrieved September 20, 2012 from http://quexml.sourceforge.net/