

ProSurvey Web Application

By

Chris Deemer

Submitted to the Faculty of the
Information Engineering Technology Program
in Partial Fulfillment of the Requirements for
the Degree of Bachelor of Science
in Information Engineering Technology

University of Cincinnati
College of Applied Science

November 2005

ProSurvey Web Application

By

Chris Deemer

Submitted to the Faculty of the
Information Engineering Technology Program
in Partial Fulfillment of the Requirements for
the Degree of Bachelor of Science
in Information Engineering Technology

©Copyright 2005 Chris Deemer

The author grants to the Information Engineering Technology Program permission to reproduce copies of this document in whole or in part.

Chris Deemer

Date

Tamisra Sanyal, Academic Advisor

Date

Patrick Kumpf, Department Head

Date

Table of Contents

Section	Page
Table of Contents	iii
List of Figures	iv
Abstract	v
1. Statement of the Problem	1
2. Description of the Solution	1
2.1. User Profile	2
2.2. Design Protocols	2
3. Deliverables	7
4. Design and Development	7
4.1. Timeline	8
4.2. Budget	9
5. Proof of Design	9
6. Testing Procedures	
7. Conclusions	11
References	14

List of Figures

Figure	Page
Figure 1. Administrator User Controls Screen Shot	3
Figure 2. Use Case Diagram	4
Figure 3. Site Navigation	5
Figure 4. Database Diagram	6
Figure 5. Timeline	8
Figure 6. Budget	9

Abstract

ProSurvey Web Application is intended to minimize the burden of researchers that gather data via surveys, as well as eliminate the need for Web developers to make surveys available on the Internet. Through the use of ASP.Net and C#, surveys are displayed programmatically from data stored in a MS SQL Database. Responses are saved to the database and can be downloaded and imported into analysis programs. Designed with market, behavioral and psychological research in mind, a wide variety of desired question formats can be created using three templates: short answer, multiple choice and multiple choice with a write in option. This functionality coupled with field validation gives researchers total control of how a survey is taken; ensuring that the data gathered is valid and accurate.

1. Statement of Problem

Conducting a survey is one of the most widely used methods of gathering data, but traditional methods have inherent flaws. Although mass mailers typically have a higher response rate than other methods, there is a wait to get responses back. Telephone surveys are not typically received well, and with the advent of the “No Call” list as well as technologies to prevent unsolicited phone calls it has become difficult to gather data this way. Conducting a survey in person can be costly when paying for all of the subjects’ time. All of these methods are also subject to human errors, whether from improper taking of a survey or typos while consolidating data.

2. Description of the Solution

The purpose of this application is to quickly and easily enable market, behavioral and psychological researchers to post a survey instrument on the Internet that can instantly be taken by participants and save the response data to a relational database for analysis. This application will save the researcher time and guarantee data integrity, while encouraging participation through ease of use. Development of this application covers four areas of Information Technology.

Providing an easy to use interface is essential when developing a Web-based application. The Website must be simple to navigate as well as present necessary information quickly to keep potential users from going to another site. HTML is used to display the information. Form validation is handled on the server-side to ensure that any Web Browser can be used with this program without malfunction. The graphics were designed in Adobe Illustrator and Photoshop.

ASP.NET framework version 1.1 is used to completely separate the HTML from the source code. These code-behind pages were developed in C# using Visual Studio.NET 2003.

The code is responsible for formatting and sending data to the database as well as retrieving stored data, interpreting it, and displaying it in HTML to the user.

Microsoft SQL Server 2003 serves as my database for this application. The included Data Transformation Services will simplify the exportation of survey response data, and scheduled Server Agent jobs will minimize DBA responsibilities.

Although the bulk of this project does not involve networking, it is a Web application and involves the set-up of a server running IIS 6. Most Web-hosting companies will take care of this for you, as well as any security or system administration issues.

2.1 User Profiles

2.1.1 Administrators

The survey administrator will either be a college student, professor or other research professional seeking to post a survey on the web. The user should already have experience creating survey instruments and analyzing the data gathered. Users must be familiar with Web browsers for Internet navigation in order to create a survey.

2.1.2 Respondents

Respondents could be anyone, and any age for that matter, that has access to the Internet. However, in order to participate in a survey they need only be familiar with basic Web browser functions and Internet navigation.

2.1.3 Guests

The guest may be a first time visitor to the site, checking the validity of a Participation Code, or registering for increased functionality. Either way the only skills necessary are familiarity with Web browsers and Internet navigation.

2.2 Design

2.2.1 Graphic Design

One of many useful features of ASP.Net is the use of User Controls to reuse code from page to page. I have enlisted the use of two User Controls: the banner containing the logo of the Web Site, and the footer which will contain various contact, copyright and company information.. The left menu's links will change from general information to survey management tools after login. See Figure 1.



Figure 1. Administrator User Controls Screen Shot

The primary font used is Arial. The gray used in the buttons and in the logo is a grayscale at 43% darkness. The blue used in the header and footer in hexadecimal is #c3d9ff.

2.2.2 Use Case

The Administrator are able to create, edit and delete surveys, as well as download current response data. A respondent's only abilities are to take a survey, and once taken be able to send verification information to who ever they choose. This third party will be able to check this information via the welcome page. See Figure 2.

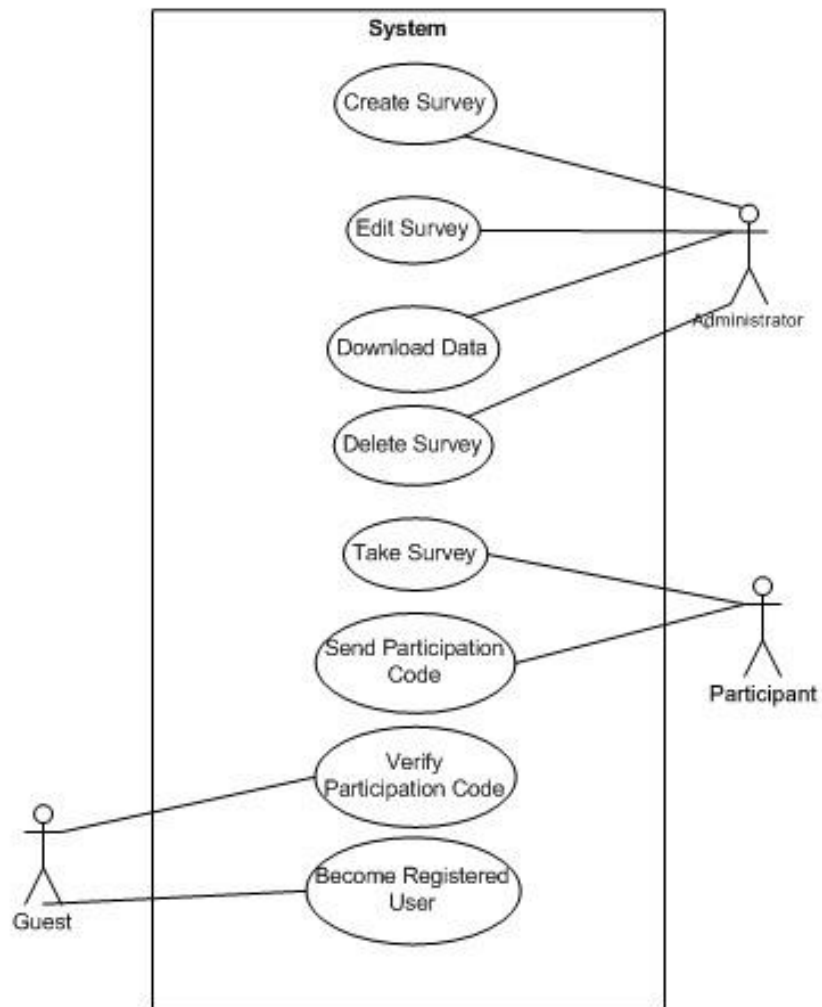


Figure 2. Use Case Diagram

2.2.3 Site Navigation

The welcome page contains information about the application, links for new users to register, and a link for existing users to login. Upon login Administrators have the ability to create new surveys, edit surveys that are not yet active for participation, delete surveys, view demonstrations of existing surveys, and download response data for analysis.

Respondents can view publicly available surveys and perform searches for surveys according to varying criteria. See Figure 3.

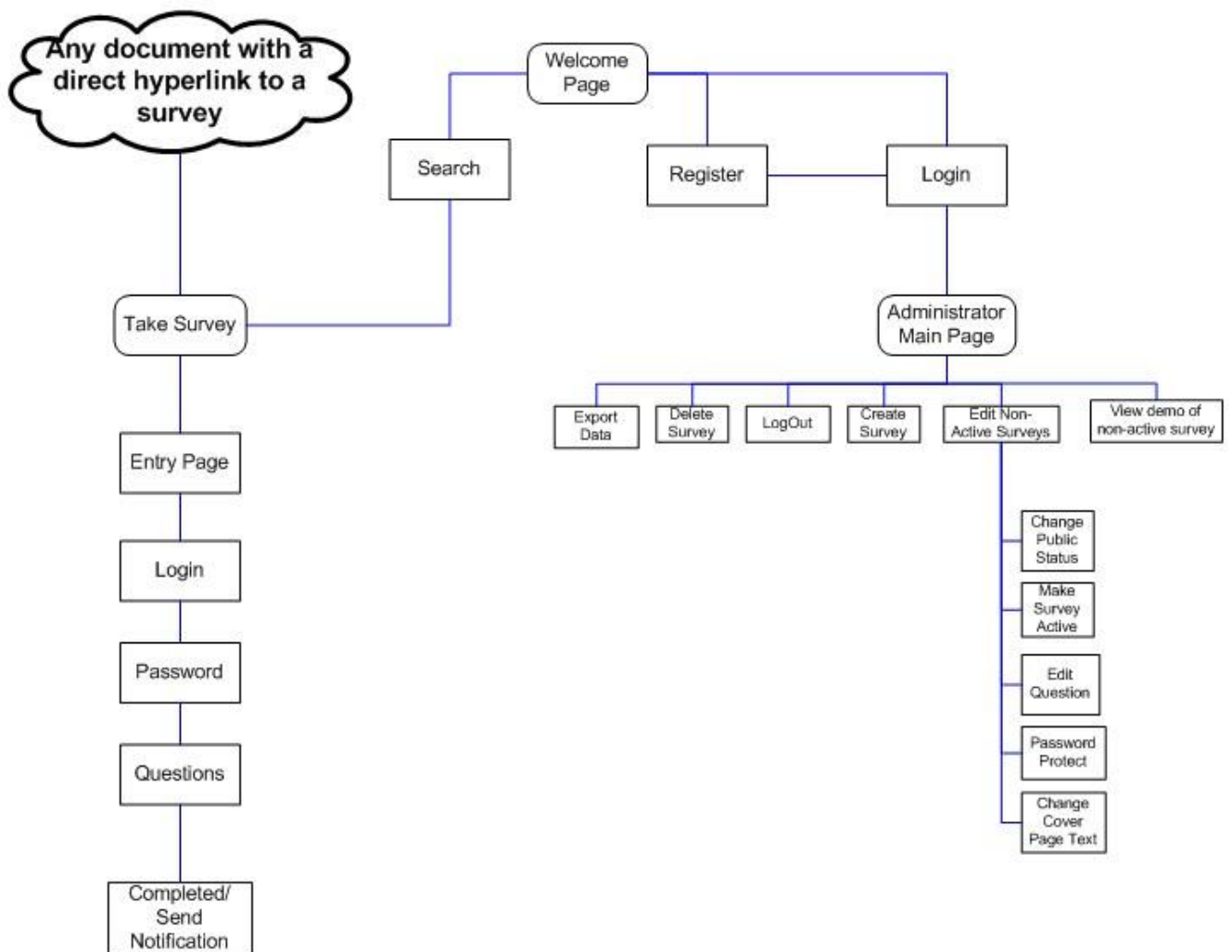


Figure 3. Site Navigation

2.2.4 Database Design

The Database design is fairly simple with only 7 main tables. An extra response table was included to hold the character strings associated with 'write-ins' and short answer questions. The denormalized table SurveysTaken was included in an attempt to prevent the database from overworking when querying for surveys a Participant has already taken as well as notifying Administrators of how many people have participated in each of their surveys.

See Figure 4.

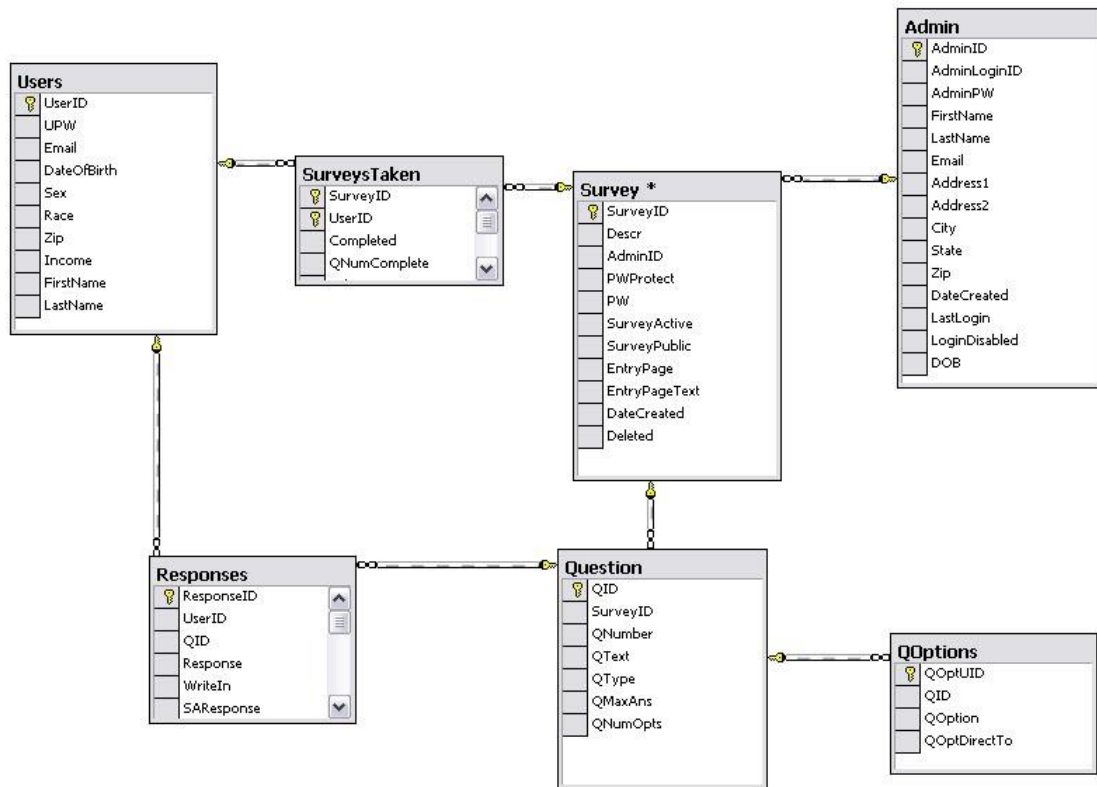


Figure 4. Database Diagram

3. Deliverables

- An application with a graphic user interface that enables researchers to easily create a complex internet survey.
- ASP.NET pages using C# to dynamically generate an HTML user interface from stored data.
- A SQL Server Database to store the survey format information as well as the response data.
- Data Transformation Service package(s) to handle the creation of downloadable files containing response data that are importable into analysis programs.
- A secure environment using a 128-bit encryption SSL certificate for respondents to participate in research surveys anonymously.
- A method for the respondents to inform third-parties of the completion of a survey, while maintaining the confidentiality of their responses.

4. Design and Development

4.1 Timeline

See Figure 5.

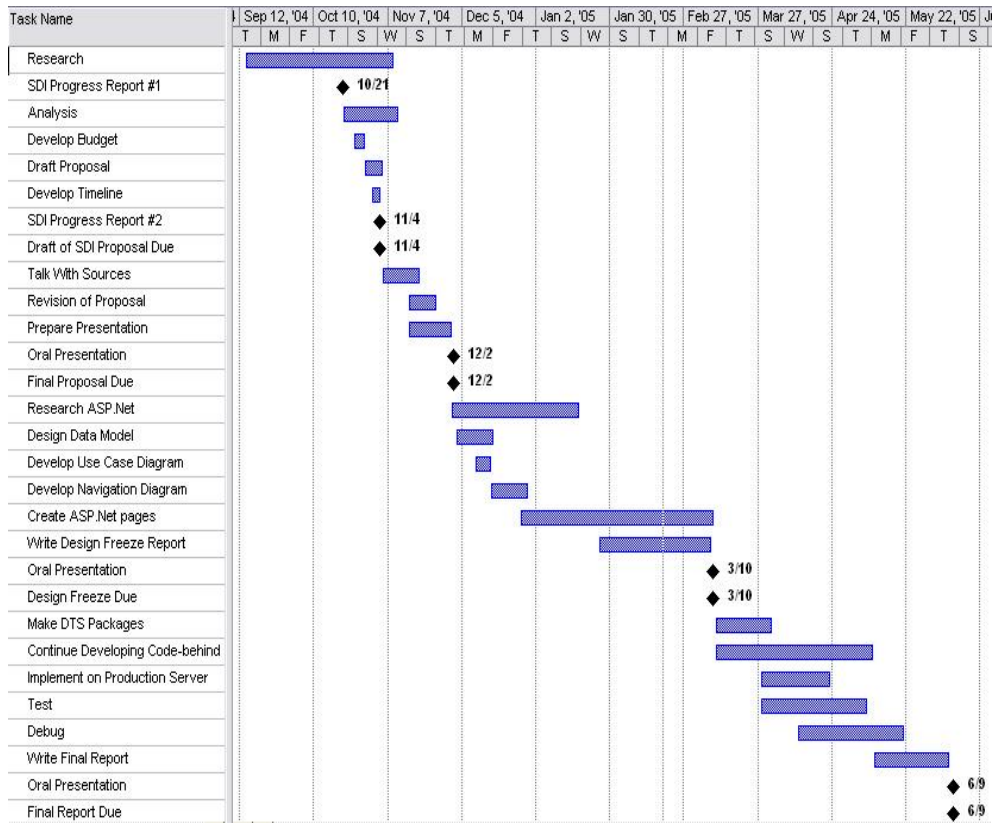


Figure 5. Time line

4.2 Budget

See Figure 6.

	Unit	Cost
Services	Web/DB Hosting ⁿ	\$5,988.00
	SSL Certificate †	199.00
	Domain Name Registration †	8.95
Software	Microsoft VS.Net ‡	1,079.00
	Adobe Photoshop †	649.00
	Adobe Illustrator †	499.00
Total	Initial Year Cost	\$8,422.95
	Yearly Recurring Cost	\$6,195.95

ⁿ Yearly price obtained from MaximumASP.com

† Yearly prices obtained from GoDaddy.com

‡ Prices obtained from Amazon.com

Figure 6. Budget

5. Proof of Design

Using the ASP.Net framework I was able to create a graphic user interface enabling researchers to create an online survey without the need of any technical expertise. C# and ADO.Net were used to store and retrieve all data to dynamically create surveys. Data is stored in a SQL Server 2000 database. This data is not only used to store user response data but also the data needed to generate each survey.

After researching the use of the Data Transformation Service included with SQL Server 2000 to export response data, it was not found to be the most practical solution for this operation. A DTS package created using the SQL Server GUI can only be called and run by a single user. Any other calls to the DTS Package while it is already running would result in an error. This being said it was also easier to format the data properly using ADO.Net and Data Tables. This also created a performance increase as it moved the formatting off of the database server and into the application itself. An open source class has also been utilized to convert the resultant data table into an excel spreadsheet and initiate the browser's download window.

An SSL certificate has also been utilized. Due to financial circumstances a Turbo SSL certificate was used in place of a more secure 128-bit certificate.

If the respondent wishes to notify someone of the completion of participation in a survey, there is the option to send an email on the last page of each survey. This email not only sends the necessary user information, but also a link to verify the validity of the email.

6. Testing Plan

Testing for this application was on-going with development similar to that of extreme or pair programming. After creating a survey, I verified that the data written to the database matches the user input. Once the data in the database was verified, participating in the survey was the next step. After verifying that the application worked with valid data, I checked invalid data for proper validation and error handling.

For security purposes as well as data integrity all input controls are properly validated. Inputting very long strings of data as well as irregular characters was done in every textbox to check for errors.

7. Conclusions

During the life of this project I designed a database, created the simple to use interface, and programmed the data access components to generate the user interface. The database was designed in the Enterprise Manager client tool that comes with SQL Server 2000. All Web pages and programming was done using Microsoft's Visual Studio 2003. Button graphics were created using both Adobe Illustrator and Photoshop. I did not fulfill every deliverable exactly as planned, but managed to accomplish the same end though not through the use of Data Transformation Services.

While learning ASP.Net I realized that there were a few main points that are crucial to ensure that the programs function properly. Understanding view state and session state and being able to store objects in them was one of the biggest problems I had. Especially when trying to maintain a dataset without querying the database on every post-back. Implementing Forms Authentication was relatively easy but very necessary in securing the web application. It was entering in the Web pages into the Web.Config file that did not need to be accessed after the administrator logs in that was often forgot. Regular Expression Validators may be the most useful and hardest to understand ready-made validators available in ASP.Net. These validators are fully customizable, limiting the user to not only use the approved characters but also length and even formatting, like a specific date format. Required field validators must also be used in conjunction with these, but it provides the utmost in securing user entry fields.

Now that I have the initial learning curve under my belt I know that I could complete a project on a much shorter timeline. And perhaps even add further functionality and spend more time developing the user interface. In the future I would try to develop more complex

uses of Object Oriented Programming to speed develop even further, by cutting down on the amount of coding I would have to do.

After choosing GoDaddy.com for my domain name, hosting and SSL certificate provider because of financial restrictions, I found out that I had a few difficulties when trying to implement this project on their server. First, connection to the database server is limited to GoDaddy's web interface, and I was forced to recreate the project database via T-SQL scripts. Second, the documentation provided in GoDaddy's help files pertaining to connection strings to the SQL Server database did not function properly, and technical support could not help me with this issue as scripting and data access support is not a free service to their customers. So I had to experiment with different keywords until I found something that worked. It turns out that a provider or driver value is not only not required but not recognized by the database server. Third, GoDaddy allows the choice of ASP.Net framework versions with their Windows hosting service. However, by default this option was set to version 2.0. I developed my application using version 1.1 and a third-party class that I used to create a radio button column inside of datagrids failed due to the lack of a trust level declared inside of the class. Changing the framework version to 1.1 remedied this error.

I have dealt with two hosting companies so far, and although GoDaddy.com offers some value choice solutions, the lack of control on the server can be frustrating. Working with MaximumAsp.com was a much more enjoyable experience, although much more expensive and targeted mainly at businesses. MaximumAsp also has 24/7 customer support with live MCSE certified technicians and ensures 4 nines of up time. My recommendations for someone who is searching for a hosting company is to seriously do the research. Check out their guaranteed up time, amount of monthly data transfers, maximum database size, and

most importantly customer service.

References

1. American Psychological Association. "Ethical Principles of Psychologists and Code of Conduct." <http://www.apa.org/ethics>, 2002.
2. Badre, Albert N. *Shaping Web Usability: Interaction Design in Context*. Indianapolis, IN: Pearson Education, Inc, 2002.
3. Duthie, G. Andrew. *Microsoft ASP.Net Programming with Microsoft Visual C#.Net Step By Step*. Redmond, Washington: Microsoft Press, 2003.
4. Goodman, Danny. *Dynamic HTML: The Definitive Reference*. Sebastopol, CA: O'Reilly & Associates, Inc, 1998.
5. Hanover College. "Psychological Research on the Net." <http://psych.hanover.edu/research/exponnet.html>, December 12, 2004.
6. Key Survey. "Online Survey Software." <http://www.keysurvey.com>, 2004.
7. Laboratory for Automation Psychology. "Online Survey Design Guide." http://lap.umd.edu/survey_design/defs.html, 2004.
8. Microsoft. "Transfer Data to Excel by Using SQL Server Data Transformation Services." <http://support.microsoft.com/default.aspx?scid=kb;en-us;Q319951>, December 12, 2003.
9. Policht, Marcin. "Exploring Windows 2003 Security: IIS 6.0." http://www.serverwatch.com/tutorials/article.php/10825_3294371_2, December 31, 2003.
10. PsychData, LLC. "PsychData Online Surveys." <http://psychdata.com>, 2004.
11. Surveymonkey.com. "Powerful Tool For Creating Web Surveys." <http://surveymonkey.com>, 2004
12. Young, Candice. Bachelors of Science in Psychology from Miami University of Ohio, Masters in Health Services Administration from Xavier University. Personal Interview. November 17, 2004.