Web-Based Decision Support Systems in Conference Registration

Christopher Todd Isler
Fall 2004
Summa Cum Laude
Bachelor of Science in Industrial and Systems Engineering

Abstract

Many college students attend at least one student-run conference at another college. Often, the host school has very limited resources and relies on emails and mailings for communication with the conference attendees. Conference registration is usually done by email, which is prone to human error and incomplete information. When the University of Florida was selected to host the Institute of Industrial Engineers 2004 Region 3 Conference, the host committee wanted a better way to collect registration information and use it to prepare for the conference. A web-based decision support system was developed, and it helped the conference team assign workshops, order t-shirts, and collect registration dues. The system also allowed attendees to easily register and ensured that their information was accurately collected. This paper details the registration system and its features from the perspective of an attendee and a host committee member.

Table of Contents

Table of Figures

1 Introduction and Problem Formulation

In January 2003, the University of Florida was selected to host the 2004 Institute of Industrial Engineers (IIE) Region 3 Conference. The conference involved gathering over 150 students from 16 schools across the Southeast to one location for three days of workshops, team-building activities, networking sessions, and speakers. Immediately it was clear that coordinating over 150 students would pose logistical problems, especially if all of the information came from emails. Past conferences collected student registration via email. Email, however, was prone to error. Having registration done via email could lead to lost information, inadequate information, and large amounts of manual labor keeping track of attendees.

The conference team brainstormed ways to make registration less reliant on email coordination. The solution the team decided upon called for a web-based conference registration system that collected attendee contact information and any other conference-related preferences (such as t-shirt size). Additionally, the conference team wanted the system to tell them which schools registered the most students, what workshops were the most popular, and the t-shirt size breakdown for all attendees. These requirements lead to the development of IIE's web-based decision support system for conference registration. This paper describes what technology was used to create the decision support system (DSS), the information it gives users, the code structure of the DSS, its ultimate use in the conference activities, and suggested improvements.

2 Solution Formulation

2.1 Use of Technology and Resources

It was immediately clear that the conference team had to decide on how it would host a dynamic website, what programming language it would use, and what kind of database it would employ. At the time the conference team started developing its website, the Industrial & System Engineering Department set up a web server with Microsoft Internet Information Server (IIS). Microsoft IIS provides support for Advanced Server Pages (ASP) 3.0. The ASP server environment allowed the conference team to easily interact with a database while providing for a dynamic registration process. Finally, the team had to decide on a database to use. The team decided to use Microsoft Access because the IIE computer already had Access loaded and it was easily understood by all team members. Using the ISE Department IIS web server, the ASP scripting environment, and Microsoft Access, the conference team possessed all the necessary resources to create a dynamic conference registration site that also included decision support elements.

2.2 Website Requirements

The team later sat down to define the requirements for the website. The team identified the following requirements:

- Take registration input from students over the internet. Registration input included the following:
    - Name

    - School
    - Address
    - Phone Number
    - Email
    - Day of Arrival (evening of first day or morning of second day)
    - Shirt Size
    - Staying on or off campus
    - Classification (Undergraduate, Graduate, Faculty, Other)
    - Day of Arrival
    - Top six workshop preferences
- Tell the students how much their registration fee was (this depended on the time of registration and if they were a UF student)
- Allow the attendees to change their workshop preferences as new workshops were added to the conference agenda
- Allow the conference team to access a list of people who were registered for the conference and see details for each attendee
- Allow the conference team to see which workshops were most popular and how many shirts of each size were needed.
- Show the conference team a breakdown of how many students were attending from each school
- Track which students had paid their registration fees

2.3 Interfaces

The conference team concluded that there were two users of this website: the conference team and the conference attendees. Thus, two interfaces were needed; one for each user. The conference attendee interface would be open to everyone who wanted to attend the conference, while the conference team interface would be password-protected.

2.3.1 Conference Attendee Interface

The conference attendee interface needed to be very simple to use because we could not assume a high level of computer knowledge by the end user. A flowchart for the user interface is shown below. Each step or block in the flowchart is further defined in following text.

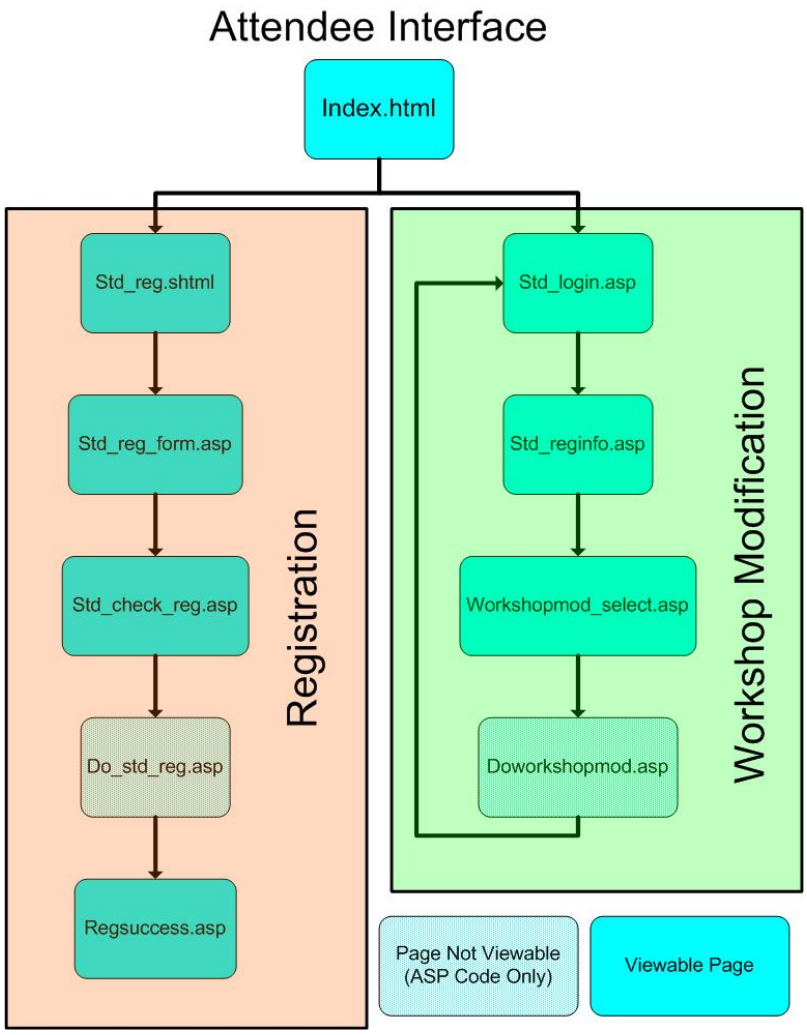

Figure 1 – Attendee Interface

2.3.1.1 Conference Attendee Interface – Registration

Registration is the primary reason for the website. The conference team used five pages (only four visible) to take a user's input, allow the user to confirm the input, update the database, and tell the user the next steps they must take.

- **Std_reg.shtml** – This page serves as an overview of the registration process for the user. It tells the user to know when they are arriving, their shirt size, and contact information.

- **Std_reg_form.asp** – This page is the actual form the users fill out their contact information. The form uses textboxes to request the user's name, address, phone number, and email address. Combo boxes are used to request the user's school, gender, classification (undergraduate, faculty, etc), shirt size, day of arrival, and IIE membership status. It also asks the user to input his preferred workshops at the conference (top 6). The user may change his workshop preferences after initial registration. The combo boxes allow the conference team to control the user's inputs. By having a limited number of inputs, it allowed the conference team to gain valuable information from queries later. All of the inputs are part of the same form and are passed to the next page using the post command.
- **Std_check_reg.asp** – This page takes the filled-in fields from the std_reg_form.asp page and passes those values to session variables. It checks that all of the fields were completed (not left blank), and displays the inputs to the user for confirmation. Should the user leave one of the fields blank, the page will redirect the user back to the form page with a session variable containing an error message to be displayed to the user. For example, if the user leaves the address field blank, the system will display "Your address is required" in large red letters.
- **Do_std_reg.asp** – This page is invisible to the user. The purpose of this page is to take the user's input and run an SQL insert query on the conference database to input the user's information. Before it runs the insert query, the code checks to see if the user has already registered by querying the database for the user's email address. If the email address is already listed, the user is sent back to the form page and is informed that he is already registered. After doing the error checking,

the user's information is inserted into the database (into "tblReg") using an insert SQL query and the session variables, and is redirected to a confirmation page.

- **Regsuccess.asp** – This is the final page for the user read before the registration process is complete. The page congratulates the user on a successful registration and tells the user the next steps to take. The conference team asked that all registration be mailed in via check. The registration fee depended upon the time of registration and school. Users who registered before February 6 or who were from the University of Florida and registered before February 13 were charged $40 for registration. Those registering after those dates were charged $50. Using if statements, the system checked the current date and user's school and told the user how much to pay. Additionally, the confirmation page instructed users to send in their resumes to the IIE email account for the resume CD the conference team would give to visiting companies.

2.3.1.2 Conference Attendee Interface – Workshop Modification

The conference team's theme for the event was "fitting into the corporate puzzle". The team wanted to bring in a wide array of companies and industries to allow students to figure out where they fit in the "puzzle". Many companies were eager to give workshops, but the conference team knew it would be securing workshops up until a few weeks before the conference. Unfortunately, registration would run parallel to securing workshops, and workshops were limited in their capacity. The team wanted to allow attendees to pick their workshop preferences during registration, but those who registered later would see a larger selection of workshops. To address this issue, the team allowed everyone who registered to change their workshop preferences at a later time. Those who

registered the earliest would receive higher priority for their workshop choices. The site thus had to allow users to see their workshops and alter their preferences. The website used four pages (three viewable) to accomplish this task.

- **Std_login.asp** – The first task was to identify who wants to change their preferences. Because the registration did not ask for a username or password, the unique identifier for each attendee was their email address. This page simply presents one textbox and a submit button for the user to see his current workshop preferences.
- **Std_reginfo.asp** – This page presents the user all of the information the database has for him. It shows contact information, school information, and most importantly, workshop preferences. The user can review his current workshops and decide whether to change them. Should the user enter an invalid email address, the page will redirect the user back to the logon page and present a message telling the user that the email was invalid. The page originally used session variables to pass information from one page to the next. However, some attendees experienced problems with session variables, so the workshop modification section uses hidden forms to pass data using the POST command.
- **Workshopmod_select.asp** – This is the page that allows the user to select a different preference order of workshops. The page shows six combo boxes that are initially set to the current workshop selection and lists all possible workshops. The workshops are pulled from the "tblWorkshops" table that lists all of the available workshops. The user may change workshop preferences and then confirm the selection.

- **Doworkshopmod.asp** – This page is the main portion of the workshop modification section, even though the user never sees this page. The page uses an update SQL query to update the six workshop preferences where the email address matches. The user is directed to the login page with a message indicating that the workshop update was successful.

2.3.2 Conference Team Interface

The interface designed for the conference team (administration interface) is where the registration system helps in the decision-making processes for the conference. The goal of the administration interface was to give the team a page to easily see statistics on the conference including items such as t-shirt sizes, workshop preferences, and school attendance. The figure below shows the structure of the administrative interface. Following are the details for each step.

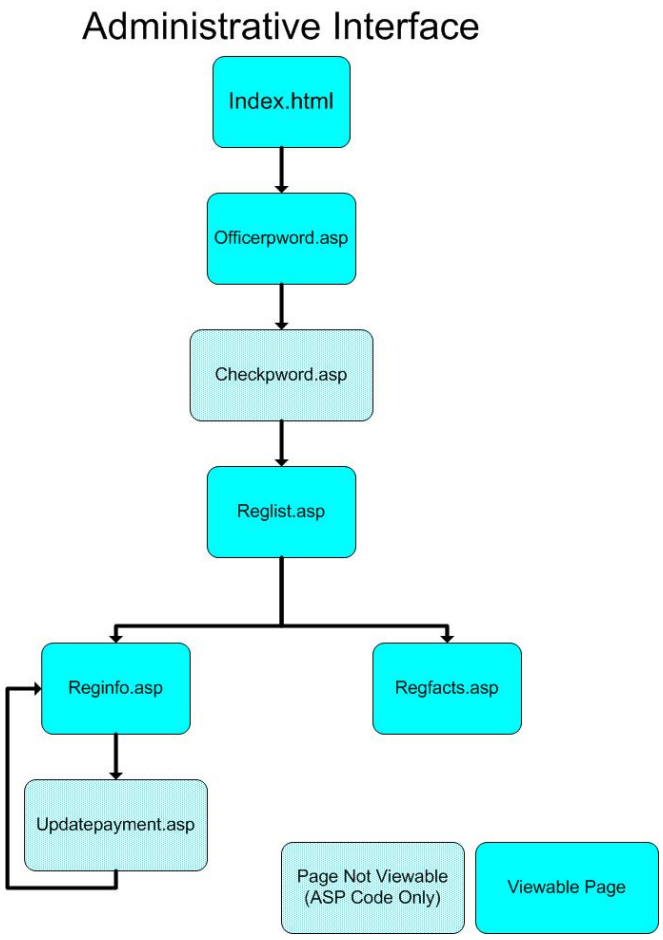

Figure 2 – Administrative Interface

- **Officerpword.asp** – From the index page, officers can click on the "Officer Resources" link which takes the user to a simple page that contains a textfield and a submit button. This is the officerpword page that captures the password needed to access the information in the administrative section of the system.
- **Checkpword.asp** – This page is invisible to the user. It compares the officer password to the real password. If the password is correct, the system gives the user a session variable named "officer" with a True value. Otherwise, the user is

redirected back to the Officerpword.asp page with a message explaining to the user that the password was invalid.

- **Reglist.asp** – This page is the core of the administrative section. At the top of the page is a table that shows the registered schools and how many attendees have registered from each school. It gives a grand total of the number of attendees registered, too. This table is pulled from an MS Access query via SQL. The Access query counts the number of records for each school in the table tblReg. Below the school table is a list of all registered attendees. The table shows the attendee's registration ID, first name, last name, school, email address, registration date, and payment status. The registration ID is the entry number that MS Access automatically gives each new entry. From this page, the user can sort the registered user table by clicking on its headers. This allows the user to sort by any field just as one would in an Excel spreadsheet. The user can also click on an attendee's registration ID and see more information about that attendee (such as arrival date, workshop preferences, etc), or the user can click on "More Facts" and see more information about all of the attendees.
- **Reginfo.asp** – This page shows the user all of the information the attendee provided at registration and also shows the attendee's current workshop preferences by querying the tblReg table from the Access database. If the attendee has not paid registration dues, a button will appear that allows the officer to mark the attendee as "paid".
- **Updatepayment.asp** – This is a simple page that is invisible to the user. If an officer wants to change an attendee's payment status from "not paid" to "paid",

the office will click on the button displayed in reglist.asp, and this page will take the user's registration ID and run an SQL update query to change the user's payment status from False to True. It then redirects the user to the reginfo.asp page with a successful update message.

- **Regfacts.asp** – This page, along with reglist.asp, are the two pages that help the conference team make informed decisions. This page shows the user three tables.
    - *Workshop Preferences* – This table was the most helpful to the conference team. It shows the user which workshops have the most demand. It lists the workshops and the total number of requests for each workshop by the attendees. It also gives a weighted total of the workshop preferences. For example, if someone listed Workshop1 as their top choice, and Workshop2 as their last choice, Workshop1 would be weighted six times as much as Workshop2. This table allowed the conference team to change workshop schedules and add additional workshop times to fit as many people into the popular workshops as possible. This information is pulled from an Access query in the database.
    - *T-Shirt Sizes* – This is a very simple table that uses an Access query to count the number of attendees that require each size. This table became useful when the team needed to order shirts before everyone had registered. The team was able to use this table to come up with a percentage breakdown for each shirt size and order 250 shirts based on the table's size breakdown.

- *Payment Table* – This is another simple table that showed the conference team the number of paid and unpaid attendees. It uses an Access query that counts the number of True entries in Paid field of the tblReg table. This helped the team determine how much additional money it would have to deposit into the bank and how many people to email before the conference began.

2.4 Benefits

The conference team found the online registration system to be an immense help. The system went beyond simply taking a user's information and storing it in a text file; it allowed the user to change preferences and the conference team to use the data without having to do any manual data manipulation.

The workshop data provided the most useful help in making decisions. The conference team was required to set a workshop schedule such that as many students could go to their workshops as possible. The workshop Disney gave was by far the most popular workshop. Initially, the team had only scheduled Disney for one workshop timeslot. Within a few days of opening registration, it became clear that the one Disney workshop would fill up too quickly, and the team was able to talk to Disney and secure a duplicate workshop later in the day. This allowed for everyone who put the Disney workshop as their number one choice to go to it.

The webpage also allowed everyone in the conference team to see the same registration data. By showing the registration data online, everyone in the team knew when a new school registered and were always up to date on the latest registration developments. For example, the school coordination chair was able to email schools as they started

registering without having to manually pull names from an Access database. This improved the team's school communication and received, and was reflected in the conference comment cards.

2.5 *Improvements*

The conference for which this system was developed was relatively small. If the conference were larger, there are several additions to the DSS that would help a conference team better manage the events.

One request from conference attendees was for multiple offerings of meals at dinner and lunch. The team discovered that many people were vegetarians or were allergic to certain types of food. During the registration process, the guests could have the option to custom-order their meals. At the end of the registration process, the DSS could output a report that the team could pass along to the caterer for the number of meals of each type (e.g. chicken, beef, vegetarian).

Greater functionality could also be added to the workshop assignment process. When the team was taking workshop preferences, it did not have firm dates and times for each workshop. The team wanted to gauge the popularity of each workshop and arrange the workshops accordingly. If more people attended, the system could save man-hours by automatically assigning attendees to workshops based on preferences, time constraints (cannot have two workshops at the same time), and space availability. With this conference, however, it was not useful to write the code needed to have the system automatically assign workshops. Instead, the conference team assigned workshops manually.

3 Conclusion

The conference decision support system was highly regarded among conference attendees and the conference team. Listed below are some comments about the online registration system from conference attendees:

- "Very effective - liked the idea that we had a choice in the workshops"
- "Registration was efficient and helpful by being available online"
- "Registration was incredible - quick and easy"

The conference team came to rely on the website as a daily tool for keeping up to date on conference registration and workshop scheduling. The online registration system eliminated registration errors and created an easier process for both the attendees and the conference team. The online registration process helped lay the foundation for a well-organized conference and required little manual manipulation of registration data. The final system can be found here: http://www2.ise.ufl.edu/iie/conf2004/index.shtml