

The Self Access Recording System (S.A.R.S.)

By

Erick McKitterick

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

June 2004

The Self Access Recording System (S.A.R.S.)

By

Erick McKitterick

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 2004 Erick McKitterick

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

<u>Erick McKitterick</u>	<u>06-03-2004</u>
Erick McKitterick	Date
<u>Hazem Said</u>	<u>6/4/04</u>
Dr. Hazem Said, Faculty Advisor	Date
<u>James F. Sullivan</u>	<u>3 June, 2004</u>
James F. Sullivan, Department Head	Date

Acknowledgements

I would like to give special thanks to Scott Frieszel, technical manager the Network Consulting Team, for giving me the opportunity to do this project. His support in this project has been extremely helpful and the feedback given throughout the long process has been invaluable. Additionally I would like to give a special thanks to Pat Lipinski, Network Consulting Team co-worker. Pat's help and input into S.A.R.S. allowed me to finish this complex project and helped support me in times of need. I would also like to take this chance to thank Reynolds and Reynolds who gave me the opportunity to complete this task for work, and for all their support throughout the implementation of S.A.R.S. Finally, I would like to give special thanks to my advisors Dr. Sam Geonetta and Dr. Hazem Said for all their support and patience in helping work around my hectic work schedule.



Table of Contents

Section	Page
Acknowledgements	iii
Table of Contents	iv
List of Figures	v
Abstract	vi
1. Statement of the Problem	- 1 -
2. Description of the Solution.....	- 2 -
2.1 User Profile	- 3 -
2.1.1 S.A.R.S. Coordinator.....	- 3 -
2.1.2 S.A.R.S. Upper Level Support	- 4 -
2.1.3 S.A.R.S. Guest.....	- 4 -
2.2 Design Protocols	- 4 -
2.2.1 S.A.R.S. Microsoft Access Database	- 5 -
2.2.2 S.A.R.S. Networking.....	- 5 -
2.2.3 S.A.R.S. Programming	- 5 -
2.2.3.a S.A.R.S. Programming Notes	- 5 -
3. Deliverables.....	- 8 -
4. Design and Development	- 10 -
4.1 Timeline	- 10 -
4.1.1 Senior Design I Accomplishments	- 10 -
4.1.2 Senior Design II Accomplishments	- 10 -
4.1.3 Senior Design III Accomplishments.....	- 11 -
4.2 Budget	- 12 -
4.3 Software	- 13 -
4.3.1 Software Server:.....	- 13 -
4.3.2 Software Server:.....	- 14 -
5. Proof of Design	- 15 -
5.1 Logon and Authentication	- 15 -
5.2 Main Menu / Daily Ticket info.....	- 16 -
5.2.1 "Hot Tickets Scrolling Marquee"	- 17 -
5.2.2 Search Capability	- 18 -
5.2.3 Run Reports Function.....	- 19 -
5.3 My Tickets	- 20 -
5.4 Ticket Screen.....	- 21 -
6. Testing Procedures	- 24 -
7. Conclusions and Recommendations.....	- 25 -
7.1 Conclusions	- 25 -
7.2 Recommendations	- 26 -
Appendix A. Research Information.....	- 27 -
Appendix B. Project Timeline.....	- 28 -
Appendix C. Code Example.....	- 29 -
C.1 Code Example from Display Ticket.asp	- 29 -

Abstract

The Self Access Recording System (S.A.R.S.) is web based interactive application to help streamline the process of the Network Consulting Team ticket tracking methods. This Active Server Page or ASP based application will help increase the productivity and process of taking on calls for the Network Consulting Team. S.A.R.S. also helps with auto notification of the proper departments such as the Design Team, and proper billing departments for automated billing. S.A.R.S. additionally has the ability to run reports to track how many tickets an individual consultant has taken and which tickets he or she has taken. By streamlining this process this gives the Network Consulting Team the ability to take on additional calls a day as well as help track calls and the status of customers when an employee is out due to sickness. The application utilizing ASP is very easy to use easy to maintain as the Administrator privileges allow the addition of the proper entries into the Access database back-end. The front-end is a very thin application written so it can be read or modified by anyone. The database was designed using Access for the database structuring back-end and by using Notepad Pro to write the database front-end. The application's functionally uses a very user friendly format and can be customized and tailored to the users preference by the utilization of cookies.

The Self Access Recording System (S.A.R.S.)

1. Statement of the Problem

This project was developed and originated in April 2003 after conversations with senior design professors (3, 10), when a project arose at Reynolds and Reynolds. After speaking with my manager (2) we decided to bring this idea up to the advisors to see if I could use it as a senior design project. After speaking with the advisors and getting the proper paperwork signed the project began in Fall of 2003.

Even though it was a known problem, the old way of completing this process is quite obscure. First, the coordinators must call the customer and schedule an appointment. Upon contacting the customer, the coordinators will get contact information and dealership numbers from an outdated non-writable database called the VAX. This system originally was supposed to be "The Core" where all customer data was collected. Over the years, this database has become completely unusable to track customer's information. Since then technology has changed and so have the customer's networks. Customers now use numerous subnets and have multiple Internet connections. These links are used to generate redundancy and is information that must be kept.

After the coordinator schedules the call he/she walks over to the Network Consulting Team and places the folder in the correct position in the queue. Each day three folders are assigned to either Team A or Team B. From here the team leaders will distribute the calls to the appropriate team member. After this is done the actual call occurs and will take the consultant anywhere from three hours to three weeks to complete depending upon the response of the customer and the current network state. Upon

completion the Consultant must print out the drawing and make the changes. When this is done the Network Consulting Team must fill out a billing paper and then make copies of all the paperwork created. After the copies are made, one copy is put back in the customer's "green folder" (given to the coordinators) and the other copy is given to the Network Design Team (NDT). Upon completion of the Coordinators and NDT's design update, the consultant must then decide who needs to be e-mailed and when so the Self Access process can be completed.

Additional internal problems occur when an employee is absent when a customer calls back. The Network Consulting Team does not currently have a way of pulling "past history" of an open ticket. Because this cannot currently be accomplished, work is often repeated on an issue that has already been addressed. Additional problems stem from the obscurity of the process from the time it takes to walk to the different locations to complete a task.

2. Description of the Solution

The Self Access Recording System is a Web based front-end, accessing and modifying an Access database backend. This site has been built for functionality to help streamline the process. Things such as automatic report generation, automatic e-mails indicating completion of tickets, ticket history, and ease of use will be incorporated into the system. The Web based front-end will help to accommodate a central system which everyone can access from anywhere within the Reynolds and Reynolds network. Since Reynolds and Reynolds is a Microsoft partner and every computer on the network comes standard with Internet Explorer 6.0, the database will be written to allow the use of "Frames" for a streaming marquee to alleviate "Hot Tickets".

Additional functionality consists of features including automated billing and scheduling access. This database allows NCT to quickly generate troubleshooting strategies when working on other Consultant's tickets. The focus of SARS is functionally and usability versus so called "looks". Another aspect of SARS is to incorporate wireless networks into the database solution to make SARS more mobile. This will allow users to walk around with their laptops and still access the database in a secure manner.

This will be accomplished using ASP based or server side pages. The use of ASP was chosen for the following reasons.

- ASP or Active Server Pages allow the processing to be done at the server end and sends back only raw HTML to the browser.
- ASP allows for a quick and efficient link to connect to the Access database backend.
- ASP easily allows web users to modify or search the database.
- ASP hides the code from the user.

2.1 User Profile

Users of the Self Access Recording System (S.A.R.S.) consist of 3 levels of users. These levels consist of S.A.R.S. Coordinator, S.A.R.S. Upper Level Support, and S.A.R.S. Guest. The three levels of users will help to control who has write abilities to the database.

2.1.1 S.A.R.S. Coordinator

Employees included in the Coordinator group have full and exclusive rights to ticket creation, modification, billing, and reopening of tickets. This allows more control

for the Coordinators to properly schedule calls and make sure there are no conflicts or double bookings for scheduled times to call customers. There are currently two coordinator accounts assigned to the proper people within the network.

2.1.2 S.A.R.S. Upper Level Support

Employees included in the Upper Level Support group include the actual consultants doing the work for the customer. This class of user tracks IP information, customer information, and firewall information. Thus the user has write access to the appropriate fields as well as write access to the notes field to allow them to track the progress of a current ticket. The notes section doubles as a previous ticket history allowing any consultant to see work done on past customers to help streamline the process. These users are not be able to create or re-open tickets to help prevent “double billing” of the customer.

2.1.3 S.A.R.S. Guest

All employees not having Coordinator or Upper Level Support status has Guest access to S.A.R.S. This allows users a basic read only right into the database. Using a guest account will allow a user to see any ticket info and history notes on the customer he or she is checking. The guest account also allows for any employee to automatically email a consultant if a guest user has a question.

2.2 Design Protocols

In this project I used three main areas of focus obtained from my education in the Information Engineering Technology Program. These areas included database design, construction, as well as application programming. In addition to programming and

database design I will be building a web server to host this project to be accessible from The University of Cincinnati for purposes of presentations to my senior design professors.

2.2.1 S.A.R.S. Microsoft Access Database

The database is used to primarily store information needed in the S.A.R.S. web environment. This database was built from scratch and will involve the use of security aspects to make sure the correct users will access the appropriate areas of the project.

2.2.2 S.A.R.S. Networking

For the purpose of presentation I set up a Windows 2000 server located at my home. This server serves the demonstration of S.A.R.S. to the Senior Design class. The users for S.A.R.S. are authenticated through this server and will not be granted access without the proper permissions.

2.2.3 S.A.R.S. Programming

The programming portion of S.A.R.S. will be written using notepad pro and is easy to read for future updates. Since the application is being written from scratch, this allows the database to be more flexible and thus ready for future changes needed to the application front-end. The code is simple and clean and used to communicate with the database. This is accomplished with the use of ASP pages to allow the best efficiency for this system. This allows the entire user interface to be generated using HTML code. The S.A.R.S. Flowchart can be found below (See Figure 1).

2.2.3.a S.A.R.S. Programming Notes

Additionally the application utilizes cookies to allow each user to customize his or her own web interface. This allows users with sensitive eyes to change the color of text and background to the user's personal preference. A scrolling marquee will be located at the top of the page to allow consultants and coordinators to communicate instantly to help alleviate "hot tickets".

S.A.R.S.

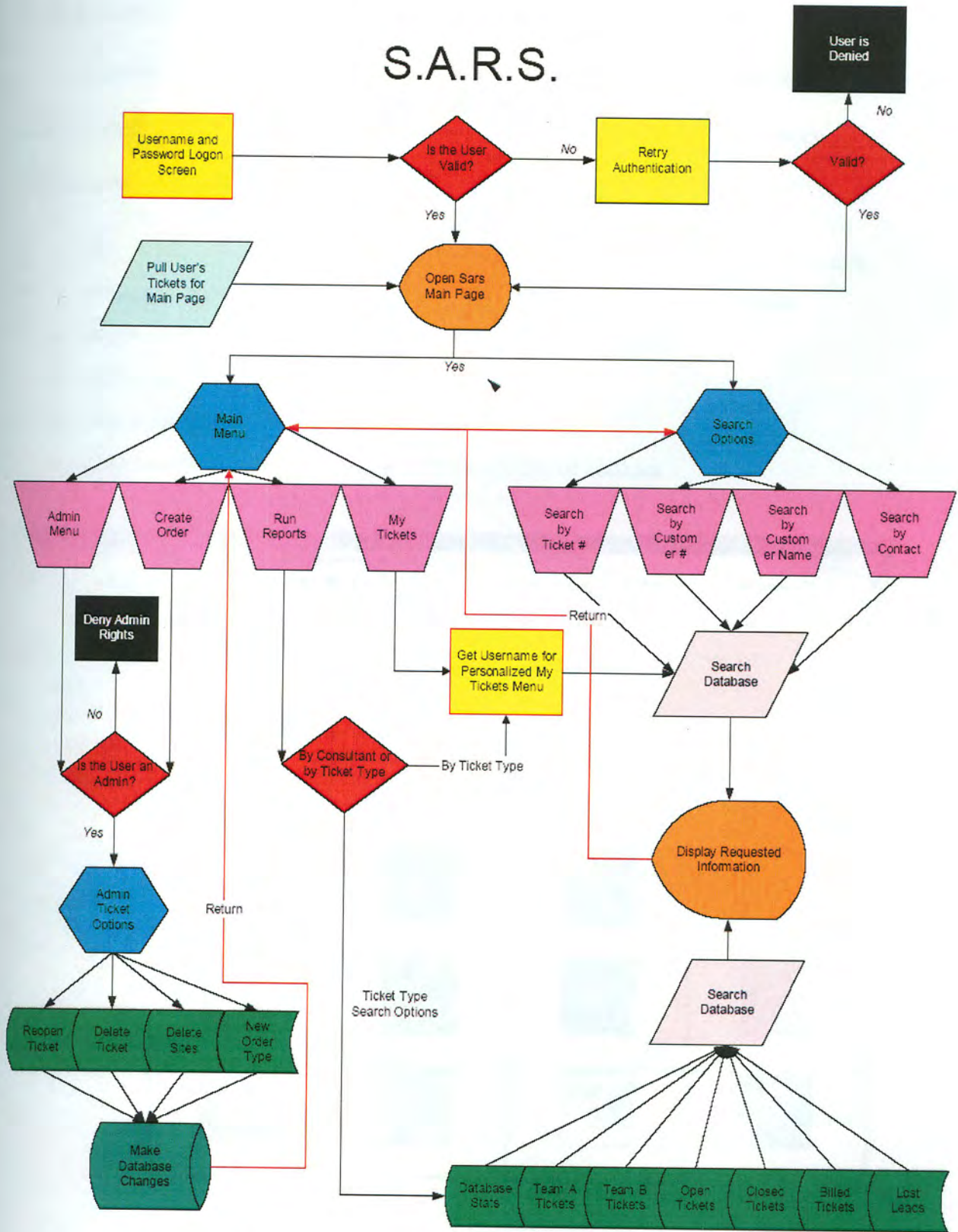


Figure 1. S.A.R.S. Flowchart

3. Deliverables

In order for S.A.R.S. be as successful these deliverables were created in order to help keep on track with what S.A.R.S. was designed to achieve. These goals were followed through ought the design and programming

- a. A web based ticket tracking system which manages and tracks customer's calls.
- b. Provider a dynamic user interface written in HTML utilizing ASP pages.
 - Allow users to customize backgrounds (See Figure 2 Below)
 - Allow users to customize font colors
 - Allow users to customize table formats
 - Customization will be accomplished via the use of cookies

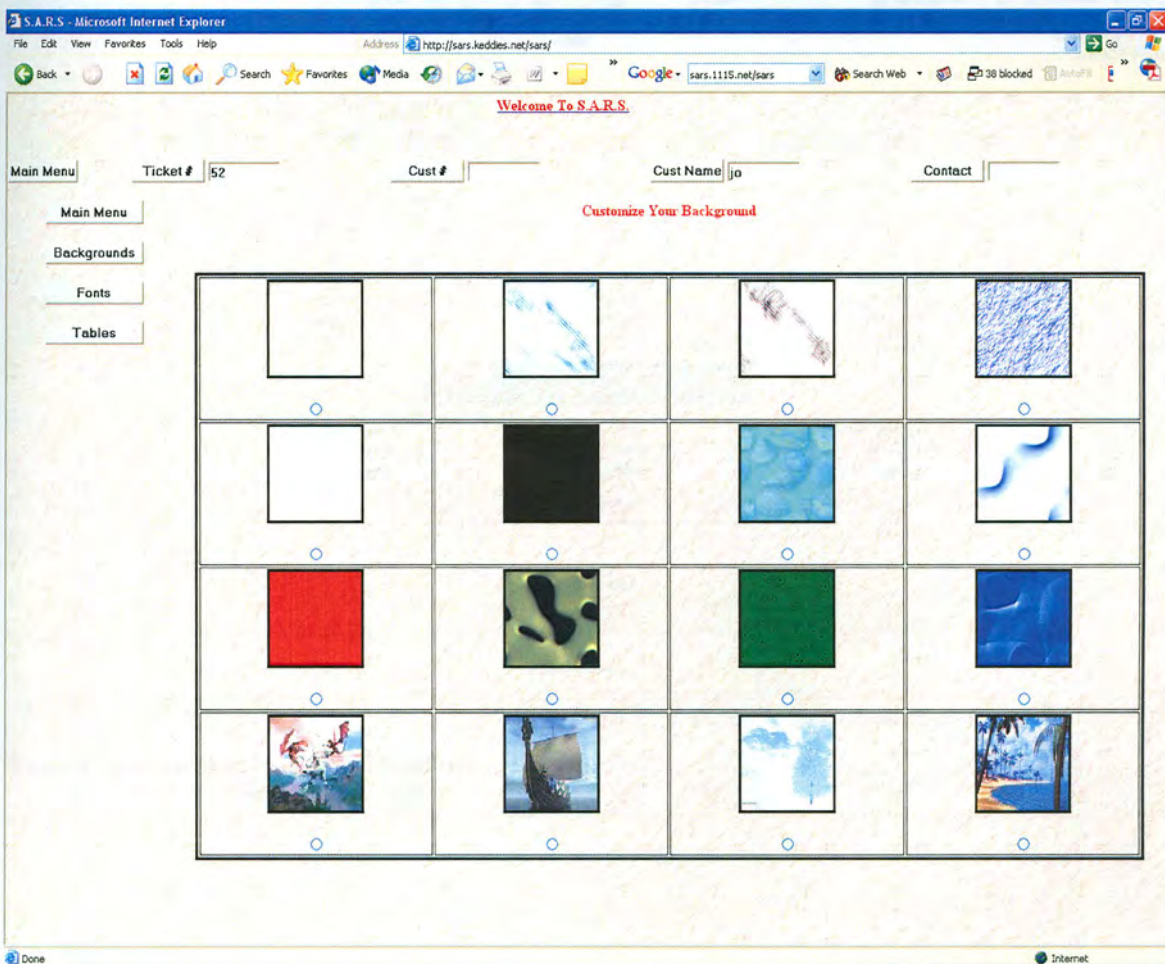


Figure 2. Backgrounds Selection Screenshot

- c. S.A.R.S. uses Active Service Pages to allow communicate between the application front-end and the database back-end.
- d. S.A.R.S. will login and use both local account security on the IIS server as well as security in the database.
- e. Additionally S.A.R.S. users will be able to complete the following tasks:
- Create new tickets
 - Modify existing tickets
 - Pull customer data from remote locations and sites
 - Track user information in a centralized database
 - Completed tickets will automatically generate an email sent to the people needing to be notified

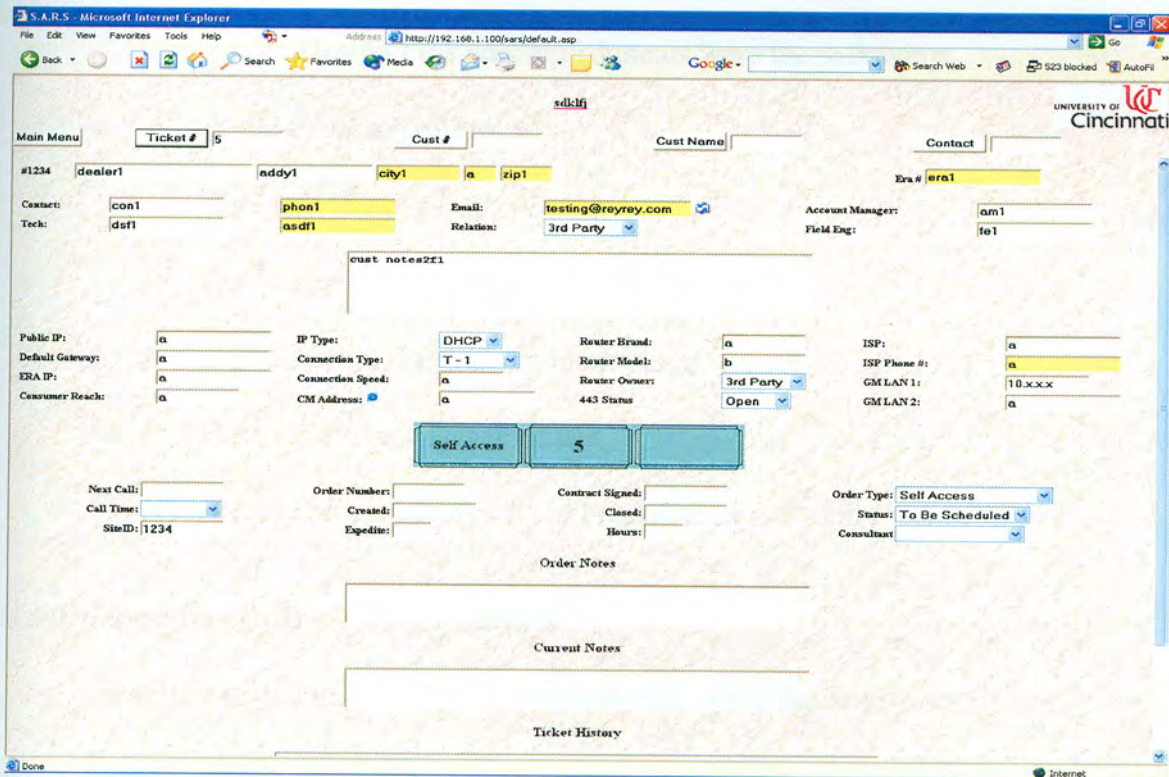


Figure 3. Screenshot from Ticket Screen

4. Design and Development

This section is dedicated to the design and development phase. Sections included in this will include Timeline, Budget, and Software.

4.1 Timeline

The project involved several challenges, learning mistakes/opportunities, and accomplishments. Below the milestones and achievements accomplished during the Senior Design sequence. The spreadsheet version can be found below (See Figure 4).

4.1.1 Senior Design I Accomplishments

During Senior Design I the accomplishments were the following:

- Initial Research of Project
- Research of Access Server Extensions
- Get Senior Design Approval
- Begin Access Database Programming
- Begin Web Page Development
- Senior Design I Power Point Presentation to Faculty

4.1.2 Senior Design II Accomplishments

During Senior Design II the accomplishments were the following:

- Finish up HTML Coding
- Link ASP pages into database
- Progress reports submitted to advisor
- Draft of Product Description
- Design Freeze Draft

- Senior Design II Power Point Presentation to Faculty

4.1.3 Senior Design III Accomplishments

During Senior Design III the accomplishments were the following:

- Test / Debug / & Revise Alpha Product
- Progress Reports Submitted to Faculty
- Test / Revise Beta Product
- Put S.A.R.S into Production
- Senior Design III Power Point Presentation to Faculty

Task	Start Date	End Date	Completed
Senior Design 1			
Research Project	4/1/2003	10/1/2003	Yes
Research Access Server Extensions	6/7/2003	8/30/2003	Yes
Get Senior Design Approval	9/22/2003	12/4/2004	Yes
Submit Paperwork for Senior Design 1	10/1/2003	12/4/2003	Yes
Begin Access Database Programming	10/15/2003	3/1/2003	
Begin Web Page Development	10/15/2003	3/1/2003	
Power Point Presentation to Faculty	12/4/2003	12/4/2003	Yes
Senior Design 2			
Finish up HTML Coding	1/1/2004	4/1/2004	
Link ASP pages into database	1/1/2004	4/1/2004	
Progress report 1	1/8/2004	2/5/2004	Yes
Progress report 2	1/15/2004	2/26/2004	
Draft of Product Description	1/22/2004	1/29/2004	Yes
Design Freeze Draft	2/1/2004	2/19/2004	Yes
Presentation to Faculty	3/4/2004	3/4/2004	
Senior Design 3			
Test Alpha Product	4/1/2004	4/8/2004	
Progress Report 1	4/1/2004	???	
Progress Report 2	4/1/2004	???	
Final Presentation to Faculty	4/1/2004	5/30/2004	
Debug Alpha Product	4/8/2004	4/15/2004	
Revise Alpha Product	4/15/2004	4/22/2004	
Test Beta Product	4/22/2004	4/29/2004	
Revise Beta Product	4/29/2004	5/6/2004	
Put S.A.R.S. into production	5/13/2004	5/13/2004	

Figure 4. Detailed Timeline for S.A.R.S.

4.2 Budget

Currently Reynolds and Reynolds is only planning on using S.A.R.S. for internal use so initial costs will be very small. However, the following hardware budget is the estimated "To Market" costs of selling a SARS box to a customer to help them track their own customer information. All prices are estimated at the time of the quote and may not be accurate upon completion of the project (See Figure 5 & 6 below).

Qty	Description	Unit Price	TOTAL
1	Windows 2000 Server Dual Pentium 4 2.0 GHz Processor 2048 MB DDR RAM Onboard Video Onboard 100/1000 NIC Additional 100/100 NIC	950	\$950.00
3	Maxtor 30 GB SCSI Hard Drives Promise Raid Controller / 128MB RAM	N/C	N/C
		SubTotal	
		Shipping	
		Tax Rate(s)	
Comments: Please note this is a DOA warehouse quote. This is not a production quote.			
		TOTAL	\$950.00

Figure 5. Proposed Internal Budget Invoice

Qty	Description	Unit Price	TOTAL
1	Windows 2000 Server Dual Pentium 4 2.0 GHz Processor 2048 MB DDR RAM Onboard Video Onboard 100/1000 NIC Additional 100/100 NIC	4,995.00	\$4,995.00
3	Maxtor 30 GB SCSI Hard Drives	600.00	1800.00
1	Promise Raid Controller / 128MB RAM	1200.00	1200.00
1	Estimated SQL Licence Fee	5500.00	5500.00
		SubTotal	13495.00
		Shipping	
		Tax Rate(s)	
Comments	Please note this is a quote taken from our warehouse on 4/1/2003		
		TOTAL	\$13,495.00

Figure 6. Proposed to Market Budget

4.3 Software

The software in this project was originally picked for flexibility. The initial design of S.A.R.S. will utilize Access as the main Database engine and after it is implemented will be converted to MS SQL Server in Q4 of 2005

4.3.1 Software Server:

Microsoft Windows 2000 Server:

This server is the key to integrating the database with the existing Reynolds and Reynolds network. This will be done by authenticating to the Reynolds and Reynolds domain controller already existing on the local area network.

MS SQL Server:

We have chosen MS SQL Server because of compatibility, dependability and cost. Currently being a Microsoft partner we can use MS SQL Server at practically no cost. However, it must be made aware if we bring this into a customer environment or productize this database additional licensing fees will be assessed. SQL Server costs in excess of \$5,000.00 and will have an impact only if this product is standardized. We also chose SQL because at the current rate of growth we feel MS Access Server's space limitation should be addressed now instead of later.

4.3.2 Software Server:

Microsoft Windows 2000/XP/XP Media Center Edition:

Microsoft will be the main platform supported by our database. Reynolds and Reynolds is not supporting Apple or Linux at this time but may incorporate it in the future. Reynolds and Reynolds additionally does not have a fixed cost for Microsoft Windows software as these are licensed via a lease of the software on a yearly basis and is included with the support model. Once again bringing this database to client production will occur additional costs of \$80.00 to \$200.00 per machine for this operating system.

Microsoft Internet Explorer 6.0/Netscape/Frame Capable Web Browser:

NCT's need for a scrolling marquee and from a designing aspect has forced me to implement frames into the design of S.A.R.S. A few years ago when bandwidth was a concern for our potential customers a non-frames version would be required. However, after doing research on our needs and wants we have decided to use frames. A non-frame browser will still be able to use the database, but will not receive all functionality.

Software Design Notes:

All code used for the "Web" or front-end of the database will be written completely from scratch. No html editors will be used in the making of SARS, which will lessen the "bloat" caused by typical web editors. Since SARS will be completely made up of "raw code", the web design will be much smaller in size, which in turn will result in much lower bandwidth requirements.

5. Proof of Design

This section is dedicated to show in detail how the deliverables were met for this project. Also covered in this section are the challenges and stumbling blocks encountered during this year long project.

5.1 Logon and Authentication

Logon will be simple as it will authenticate into the local servers user table and will assign appropriate rights to the user though a combination of database security and Windows Server security policies (See Figure 7).

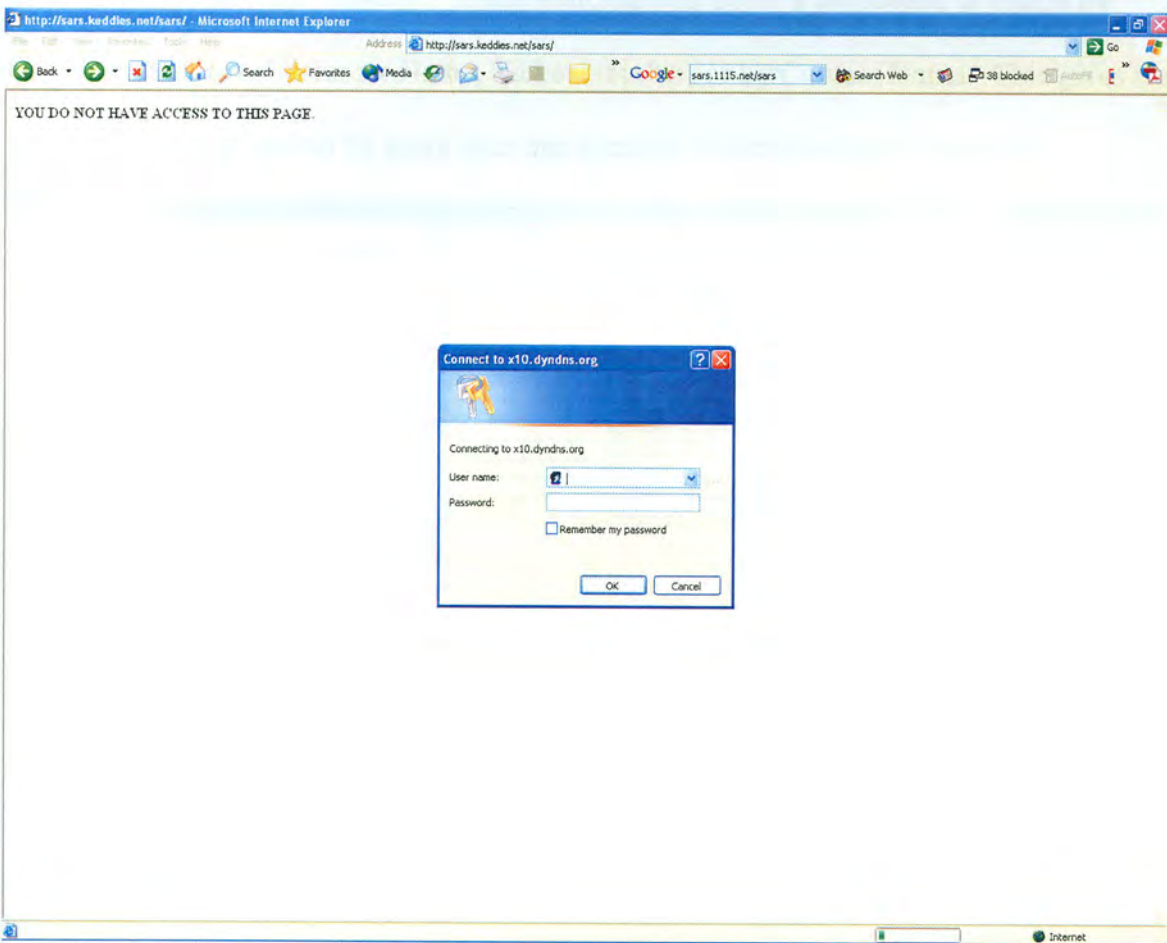


Figure 7. S.A.R.S. Remote Logon Screen

5.2 Main Menu / Daily Ticket info

The main menu will display the tickets scheduled for the current day. If the user is one of the power users it will show the last entry on the notes section. Additionally this page will allow for a drop down box for any new calls not yet assigned. This allows the users to “Quick Change” the consultant assigned to the ticket. Also included in this screen is the “Hot Tickets Scrolling Marquee” This section is located at the top of S.A.R.S. which allows users to alert consultants of tickets which need to be addressed A.S.A.P. Addition functionally of the main page and other pages is the search bar located below the “Hot Tickets”. These searches will allow a consultant to search by Customer Number, Customer Name, Ticket Number, or by Contact Name. This gives the users the ability to search by more than one piece of information (See Figure 8).

Microsoft Internet Explorer window showing the S.A.R.S. Main Menu. The browser address bar displays `http://sars.keddies.net/sars/`. The page title is "Welcome To S.A.R.S.". The main content area features a search bar with fields for Ticket # (52), Cust #, Cust Name (jo), and Contact. Below the search bar is a table of tickets:

Ticket #	Cust #	Cust Name	Consultant	Service	Time	Status
93	123124	Jo Schmo Dealer	B - Team	Self Access	9:00 AM	To Be Scheduled
94	654	Test 1	A - Team	Contact Management	11:00 AM	To Be Scheduled
95	987	dsafa	A - Team	Hourly Consulting	3:00 PM	To Be Scheduled
96	54198198	65	Eric McKitterick	Hourly Consulting	4:00 PM	In Progress

Below the table, it states: "There are 4 Tickets for 6/1/2004". On the left side, there is a "Main Menu" with links for My Tickets, Customize, Reports, Edit Marquee, Create Order, and Admin Menu. The user "sarsadmin" is logged in.

Figure 8. S.A.R.S. Main Menu Screenshot

5.2.1 “Hot Tickets Scrolling Marquee”

This feature is pretty self explanatory. The “Hot Tickets” will be displayed on every screen which allows for communication between consultants without the use of email or phone. It additionally allows coordinators to display expedite calls which come in for our team. After a coordinator gets back the signed hourly contract along with authorization for the expedite fee they post the call in this section. After one is posted it allows to link to the appropriate ticket for the first available consultant (See Figure 9).

Microsoft Internet Explorer
Address: http://sars.keoddes.net/sars/

Back Forward Stop Home Search Favorites Media Print AutoFill

Google sars.1115.net/sars Search Web 38 blocked AutoFill

Welcome To S.A.R.S.

Main Menu Ticket # 52 Cust # Cust Name jo Contact

#115288 Brown Motor Sales Co Inc. 5625 West Central Toledo OH 43615 Era # 419-534-6791

Contact: Rick Hanson 419-531-0151 Email: Account Manager: Wines, Tim
Tech: None None Relation: DealerShip Field Eng:

Update

Current Site: 115288

Public IP: 205.162.221.247 IP Type: Router Brand: Cisco ISP: Buckeye Cable Vi
Default Gateway: 10.2.191.1 Connection Type: Router Model: PIX ISP Phone #: 419-724-9800
ERA IP: 10.2.190.194 Connection Speed: Router Owner: Customer GM LAN 1:
Consumer Reach: CM Address: 443 Status: Closed GM LAN 2:

[ADD SITE 115288](#)

All Tickets For This Customer

#	Site #	Order Type	Consultant	Status	Closed
52	115288	Self Access	Sars Admin	Closed	2/25/2003

Self Access 52 Sars Admin

Next Call: 2/25/2003 Order Number: 453631 Contract Signed: 1/14/2003 Order Type: Self Access
Call Time: 1:00 PM Created: 1/1/2004 Closed: Hours: 1 Status: Closed
Site ID: 115288 Expedite: Hours: 1 Consultant: Sars Admin

Order Notes

Done 7 of 24 - Clipboard Item collected.

Figure 9. Scrolling Marquee Screenshot

5.2.2 Search Capability

This feature allows for quick searching of the S.A.R.S. database to find customer info. This can search by any method described in the bar (See Figure 10).

The main method of searching are:

- Ticket Number
- Customer Number
- Customer Name
- Contact Info – Person's name at dealership

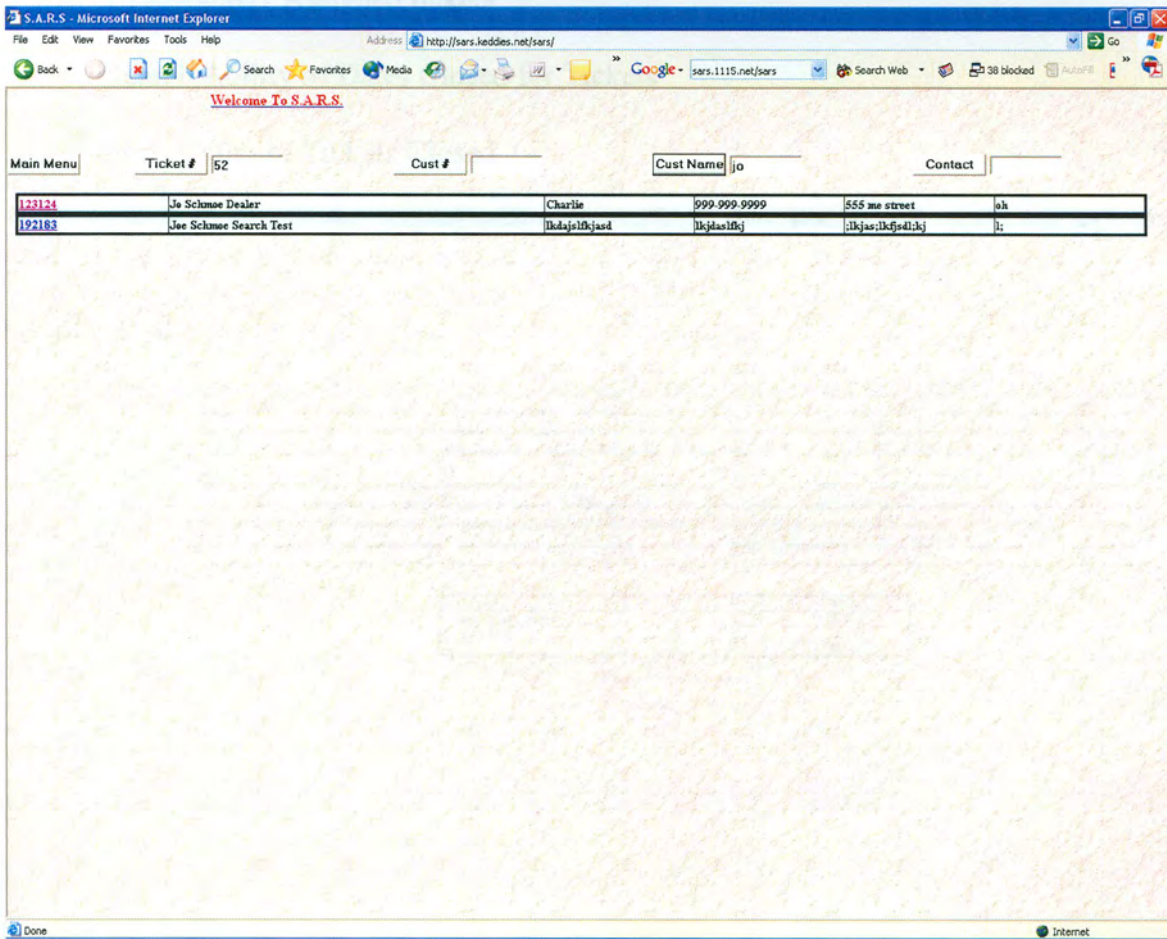


Figure 10. Search Screen Results

5.2.3 Run Reports Function

This feature allows for quick access to the closed tickets and reporting functions.

It allows a manager or power user to access these features (See Figure 11).

The main method for “Run Reports”:

- Gross Revenue Generated
- Closed Tickets by Team or Individual
- Types of Calls Closed
- Unassigned / Assigned tickets
- Search Ability of Dates Closed for Tickets
- Sort Ability of Tickets Closed

Welcome To S.A.R.S.

Main Menu Ticket # | 52 Cust # | Cust Name | jo Contact |

A - Team
B - Team
Closed

Tickets Closed From 10/1/2003 to 6/1/2004

▼Ticket #	▼Customer #	▼Dealer	▼Consultant	▼Order	▼Closed
51	412229	Performance Motors	Patrick Lipinski	Contact Management	10/8/2003
80	7679109	Gossett Volkswagon	Sars Adman	Contact Management	10/3/2003
83	19718	Budget Leasing (DBA Roger Beasley Volvo)	Patrick Lipinski	Self Access	11/7/2003
84	21299	South City Ford, Inc.	Eric McKitterick	Self Access	10/15/2003
85	24040	Hackettstown Ford, Inc.	Eric McKitterick	Ford Secure	11/25/2003
86	7995	York Motors	Patrick Lipinski	Contact Management	10/27/2003
87	2042	Bermont Motors	Patrick Lipinski	Self Access	3/4/2004

Self Access	3	\$3525
Contact Management	3	\$3525
Ford Secure	1	\$1175
Hourly	0	\$0 / \$0

There are 7 Closed Tickets
Totalling : \$8225

Figure 11. Run Reports Screenshot

5.3 My Tickets

The My Tickets button allows a user to quickly see only those tickets which are assigned to him or her. This allows a consultant to remove the clutter of the other tickets when not necessary. It allows the consultant to see when the next scheduled call is and see exactly how many open tickets he or she has in the queue (See Figure 12).

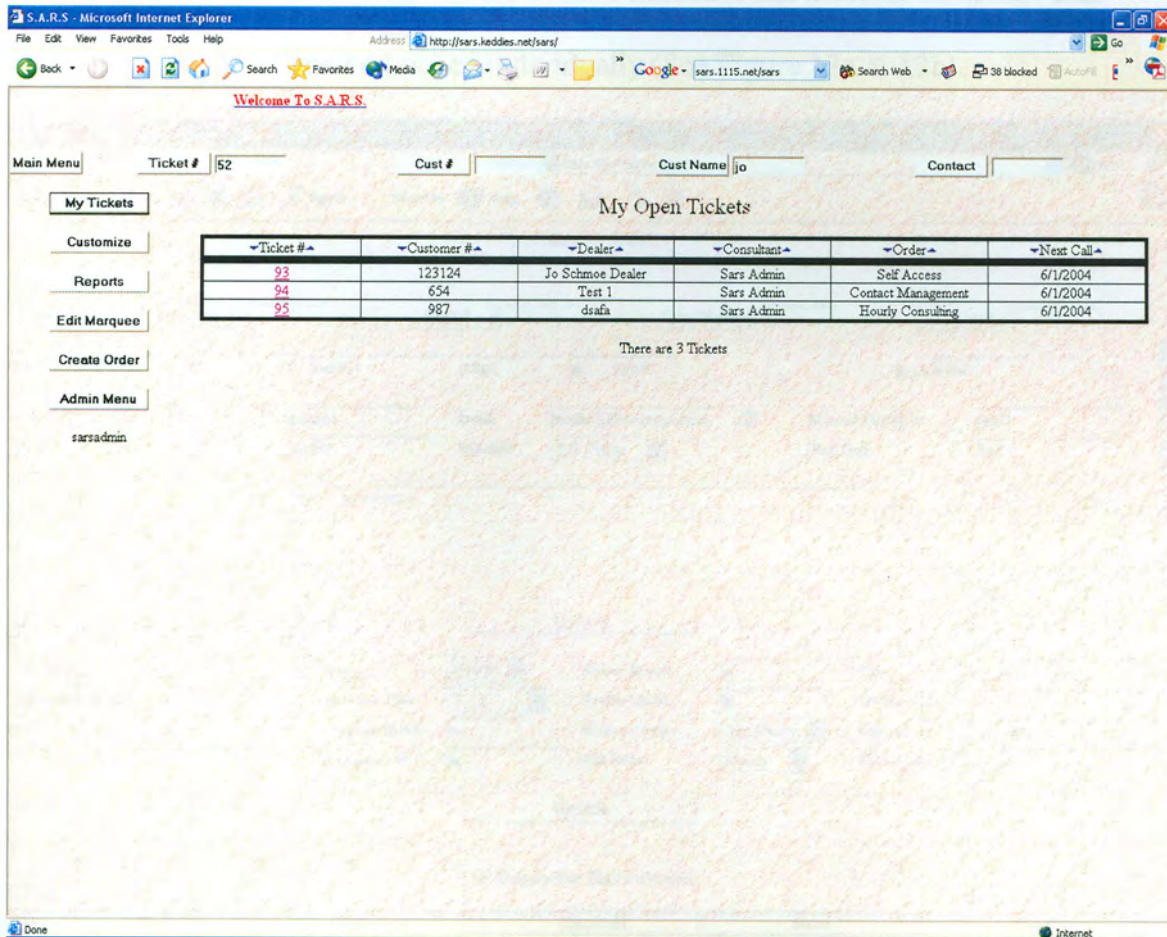


Figure 12. My Tickets Screenshot

5.4 Ticket Screen

As shown in the database things such as IP information, dial-in numbers, and customer contact information can be store all in one organized place. This site also allows for the creation of “Sites” in the event a franchise has multiple buildings and locations each “Site” can be tracked individually (See Figure 13a & 13b).

The screenshot shows a web browser window titled "S.A.R.S. - Microsoft Internet Explorer" with the address "http://darkserver/sars/". The page content includes a "Welcome To SARS" header and a "Main Menu" with a "Ticket #" dropdown set to "4". Below this are input fields for "Cust #", "Cust Name", and "Contact".

The main form area contains several sections:

- Customer Information:** Fields for "#1234", "dealer1", "addy1", "city1", "a", "zip1", and "Era # era1".
- Contact Information:** Fields for "Contact: con1", "phon1", "Email: testing@reyrey.com", "Account Manager: am1", "Tech: dsf1", "asdf1", "Relation: 3rd Party", and "Field Eng: fe1".
- Notes:** A text area labeled "cust notes2f1".
- Update Button:** A button labeled "Update" below the notes area.
- IP and Network Settings:** Fields for "Public IP: a", "IP Type: DHCP", "Router Brand: a", "ISP: a", "Default Gateway: a", "Connection Type: T-1", "Router Model: b", "ISP Phone #: a", "ERA IP: a", "Connection Speed: a", "Router Owner: 3rd Party", "GMLAN 1: 10.x.x.x", "Consumer Reach: a", "CM Address: a", "443 Status: Open", and "GMLAN 2: a".
- Update Button:** A second "Update" button below the IP settings.
- Tickets Table:** A table titled "All Tickets For This Customer" with columns for "#", "Site #", "Order Type", "Consultant", "Status", and "Closed".

#	Site #	Order Type	Consultant	Status	Closed
3	1234	Self Access	A - Team	To Be Scheduled	1/1/2004

The browser status bar at the bottom shows "Done" and "Local intranet".

Figure 13a. Ticket Screenshot

S.A.R.S - Microsoft Internet Explorer
 Address: http://darkserver/sars/

File Edit View Favorites Tools Help

Back Search Favorites Media

Welcome To SARS

Main Menu Ticket # 4 Cust # Cust Name Contact

6	1234	SelfAccess	B - Team	To Be Scheduled	1/1/2004
---	------	------------	----------	-----------------	----------

Contact Management 4

Next Call: 1/1/2003
 Call Time: 2:00 PM
 SiteID: 1234

Order Number: _____
 Created: _____
 Expedite: _____

Contract Signed: _____
 Closed: _____
 Hours: _____

Order Type: Contact Management
 Status: To Be Scheduled
 Consultant: _____

Order Notes

Ordernotes

Current Notes

Ticket History

On 2/19/2004 at 1:58:09 AM
 The Next Call Time was changed From 1:00:00 PM TO 2:00 PM
 By: administrator

Done Local intranet

Figure 14b. Notes Screenshot.

This is the heart and sole of S.A.R.S. This allows the user to enter or get the data needed from each customer. This information is vital to the responsiveness of the consultant and very important to future calls to the customer. The information is tracked in the S.A.R.S. database tables (See Figure 14).

S.A.R.S. Database Tables						
Consultants	Customers	Marquee	Ordertypes	Sites	Teams	Tickets
ConsultantID	CustomerNumber	text	OrderID	SiteID	TeamNumber	CustomerNumber
Consultant	DealerShipName	url	OrderDescription	CustomerNumber	TeamName	SiteID
TeamName	Address			PublicIPAddress		OrderID
Admin	City			DefaultGateway		OrderNotes
userid	State			EraIP		BillingNotes
email	Zip			CRIP		Journal
	EraPhoneNumber			IPType		ConsultantID
	MainContact			ConType		DateCreated
	MainPhoneNumber			ConSpeed		DateClosed
	TechContact			ISP		Status
	TechPhoneNumber			ISPPhoneNumber		Hours
	TechRelation			GMLAN1		NextCallDate
	CustomerNotes			GMLAN2		NextCallTime
	CustomerEmail			CRPublic		ExpediteFee
	FaxNumber			Open443		ContractSigned
	AccountManager			RouterMake		
	FieldEngineer			RouterModel		
				RouterOwner		
				SiteAddress		
				SiteCity		
				SiteState		
				SiteZip		

Figure 15. S.A.R.S. Database Structure

6. Testing Procedures

Testing was an integral part of the development process for S.A.R.S. Each part of S.A.R.S. is linked in one way or another to yet another critical function of S.A.R.S. The testing procedure was mostly done by an extreme programming method. This was done by actually putting S.A.R.S. into a staged Alpha or Beta stage and then as problems arose or new functionality was needed the code was rewritten. By taking this approach it added some challenges in the actual design process of S.A.R.S. By basically testing on live data it allowed for S.A.R.S. to take on definition quickly. However, on the other hand, when problems arose they had to be addressed quickly to allow for the Network Consulting Team to continue their jobs.

The pieces of S.A.R.S. I found most challenging was the actual automatic e-mailing and other built in reporting features of S.A.R.S. These features seemed to be the smallest yet ended up being the most complex and in some cases being the most troublesome to implement. The usage of E-murl helped to great lengths. This freeware program allowed me to setup an internal server within S.A.R.S. complete what I considered one of the easiest tasks. However, after attempting to actually write the code to interface with this program.

Additional testing was done by the manager Scott Frieszel. He overlooked the entire process and creation of S.A.R.S. The input he gave was invaluable and was used in many situations to help streamline the process of creating S.A.R.S. which in turn has helped to make the Network Consulting Team run more smoothly.

Additional unit testing in a staged environment helped to eliminate minor syntax errors. Planning before, during, and after the design process and coding process reduced programming time and development delays caused by general debugging and error checking.

Since programmers normally have a skewed view of how well a program works or how it should work. By not using these traditional ways of programming and actually being an end-user as well I believed helped in great lengths as to how efficient S.A.R.S. was created.

System testing was completed on the following Microsoft operating systems: Windows 2000 and Windows XP. Since Reynolds and Reynolds is a Microsoft Partner and we exclusively use Microsoft Internet Explorer 6.0 or higher no other browsers were fully tested. However, we found both Netscape 7.0 and Mozilla did work but with some functionally loss.

7. Conclusions and Recommendations

7.1 Conclusions

This project was created in response to Reynolds and Reynolds Network Consulting Team's need and desire for a way to streamline their process for taking on calls and for tracking trouble tickets. The solution was created using Active Server Page technology which was thin and easily accessible for all employees needing to access the data. Because S.A.R.S. was created and used no web page editors in the creation of this application S.A.R.S. can efficiently work even over dialup VPN access. The application was designed with functionality in mind, however, each user can customize his or her own interface to adjust it to their own personal settings.

To prepare the project, I used Microsoft Access, Microsoft Server 2000, and Note Pad Pro. The project was completed over the three quarter Senior Design sequence and during this time I received invaluable assistance from the senior design professors. The budgets used in this scenario were built to be internal use at Reynolds and Reynolds and may not reflect accurate numbers in so called "Real World" cases. The project fulfilled all the Design Freeze deliverables, and testing was done to ensure the product's usability, and functionality.

7.2 Recommendations

I recommend to anyone taking on Senior Design to truly dig down in his or her first quarter and get a full understanding of what they are getting into. I never thought this project would get so big or complex and the hours spent completing this is countless. Improvements on my design I believe could be done using things such as Crystal Reports or writing the application in a non "Web Based" environment. Additional functionality could be achieved using Adobe to automatically make "PDF" files for greater flexibility of the application. Also since I believe Access is slow in nature the use of SQL Server would be nice to help speed the response times of the application. In closing, even though this is one of the hardest challenges I have faced in my lifetime, I consider this an excellent learning experience and would like to once again thank Reynolds and Reynolds as well as The University of Cincinnati for everyone's help.

Appendix A. Research Information

During my research of developing S.A.R.S. I found many things I wasn't quite familiar with. I found the Google search engine to be quite valuable when I found myself in a jam. Also during my research I found many books today are either outdated or are flat out wrong in their code examples. One would think a book published by Microsoft would be accurate and up-to-date, however, I found several cases in Microsoft ASP books and Access books where the examples did not work(4). Therefore I found Google to be a great supplement when I ran into these problems. Additional research sources can be found in the research section of this document.

Appendix B. Project Timeline

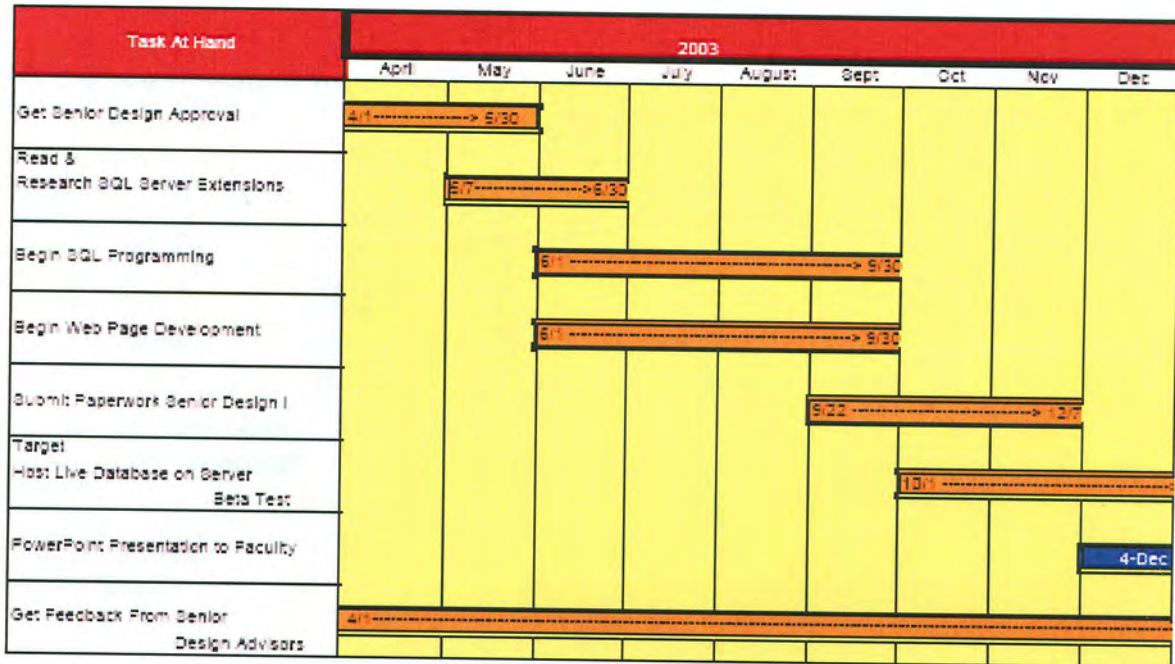


Figure 16. 2003 Schedule

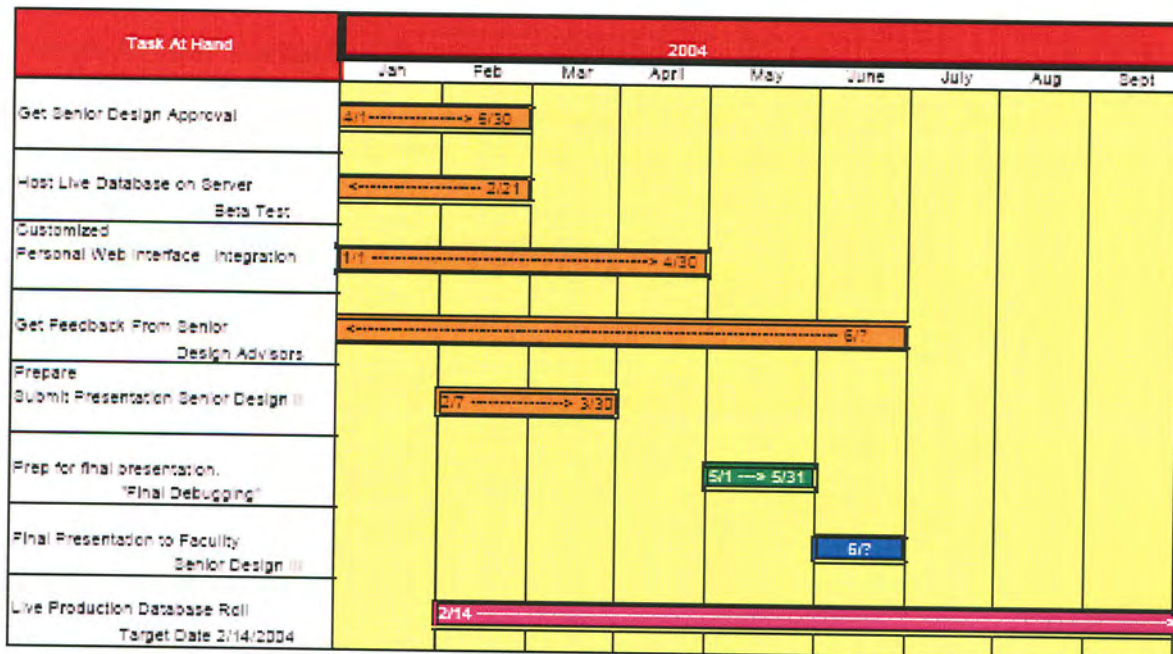


Figure 17. 2004 Schedule

Appendix C. Code Example

C.1 Code Example from Display Ticket.asp

```
<% @ Language=VBScript %>
<!--#include file="db_connect.asp"-->
<%
' =====Setup Display Vars=====
background      =      request.cookies("background")
font            =      request.cookies("font")
backgroundcolor =      request.cookies("backgroundcolor")

if background = "" THEN
background = "paperz.gif"
END IF

if font = "" THEN
font = "000000"
END IF

if backgroundcolor = "" THEN
backgroundcolor = "#FFFFFF"
END IF
' =====

' =====Get User Name=====

userid          =      Request.ServerVariables("auth_USER")
ServerID        =      Request.ServerVariables("SERVER_NAME")

If userid = "" THEN
        Response.Status = "401 access denied"
        Response.Write("<B>You DO NOT HAVE ACCESS TO THIS PAGE.</B>")
Response.END
END IF

' Format username to all lowercase, and remove any possible server name
userid          =      (lcase(userid))
x              =      Instr(1,userid,"\",1)
userid         =      Mid(userid,x)
userid         =      Replace(userid,"\", "")

%>
<html>
<title>Ticket Display</title>
<base target="main">
<body background = "backgrounds/<%Response.Write(background)%>">
<link rel="stylesheet" type="text/css" href="style.css" />
<font color="<%Response.Write(font)%>">
```

<%

'----- Pull Ticket Number Or CustomerNumber from URL-----

TicketNumber = Request.QueryString("TicketNumber")

' If there is not ticket number in the url check to see if it was from a form

IF TicketNumber = "" THEN

TicketNumber = Request.Form("TicketNumber")

END IF

' end form check

CustomerNumber = Request.QueryString("CustomerNumber")

If CustomerNumber = "" THEN

CustomerNumber = Request.Form("CustomerNumber")

END IF

IF Ticketnumber = "" AND CustomerNumber = "" THEN

Response.Write "<<<----- Please Enter a Valid Ticket Number"

Response.END

END IF

'-----Validate TicketNumber / CustomerNumber -----

IF IsNumeric (TicketNumber) = "False" THEN

Response.WRITE "<<<----- Please Search Again. Invalid Ticket Number Or Customer Does not Exist"

Response.END

END IF

IF IsNumeric (CustomerNumber) = "False" THEN

Response.WRITE "<<<----- Please Search Again. Invalid Ticket Number Or Customer Does not Exist"

Response.END

END IF

'-----End Validation of TicketNumber / CustomerNumber -----

'-----Decide, are we looking at a ticket or Just a customer-----

IF TicketNumber <> "" THEN

sql = "SELECT * From Tickets,Customers,sites,consultants,ordertypes " & _
"WHERE TicketNumber = " &TicketNumber & _
" AND Tickets.CustomerNumber = Customers.CustomerNumber" & _

```

" AND Sites.SiteID = Tickets.SiteID "& _
" AND Tickets.OrderID = OrderTypes.OrderID" & _
" AND Tickets.ConsultantID = Consultants.ConsultantID"

```

```
ELSE
```

```

sql      =      "Select * From Customers,sites Where Customers.CustomerNumber = "
&CustomerNumber & _
          " AND Customers.CustomerNumber      = Sites.SiteID"

```

```
END IF
```

```

'-----Find customer and ticket info-----
call db_connect(conn)

```

```
set rs = server.CreateObject("ADODB.RECORDSET")
```

```
with rs
```

```

    .cursortype = 1
    .locktype = 1
    .activeconnection = conn
    .open sql

```

```
end with
```

```
' Check to see if this ticket really exists
```

```
IF rs.eof THEN
```

```
Response.WRite "<B><><>----- Please Search Again. Invalid Ticket Number Or Customer Does not  
Exist</B><BR>"
```

```
Response.END
```

```
END IF
```

```
' And the fun part, Pull all the Vars
```

```
' While this could be done inline
```

```
' I think its cleaner to code them twice
```

```
' Customer info
```

```

CustomerNumber      =      rs("CustomerNumber") ' Why not
DealerShipname      =      rs("DealerShipName")
Address              =      rs("Address")
City                 =      rs("City")
State                =      rs("State")
Zip                  =      rs("Zip")
EraPhoneNumber       =      rs("EraPhoneNumber")
FaxNumber            =      rs("FaxNumber")
CustomerEmail       =      rs("CustomerEmail")
MainContact          =      rs("MainContact")

```

```

MainPhoneNumber          =      rs("MainPhoneNumber")
TechContact              =      rs("TechContact")
TechPhoneNumber          =      rs("TechPhoneNumber")
TechRelation            =      rs("TechRelation")
TechContact              =      rs("TechContact")
TechRelation            =      rs("TechRelation")
CustomerNotes           =      rs("CustomerNotes")
AccountManager          =      rs("AccountManager")
FieldEngineer           =      rs("FieldEngineer")
' End customer Info

```

' Site Info IP

```

SiteID                  =      rs("SiteID")
PublicIPAddress         =      rs("PublicIPAddress")
DefaultGateway         =      rs("DefaultGateway")
EraIP                  =      rs("EraIP")
CRIP                   =      rs("CRIP")
IPType                 =      rs("IPType")
ConType                =      rs("ConType")
ConSpeed               =      rs("ConSpeed")
ISP                    =      rs("ISP")
ISPPhoneNumber        =      rs("ISPPhoneNumber")
GMLAN1                 =      rs("GMLAN1")
GMLAN2                 =      rs("GMLAN2")
CRPublic               =      rs("CRPublic")
Open443                =      rs("Open443")
RouterMake             =      rs("RouterMake")
RouterModel            =      rs("RouterModel")
RouterOwner            =      rs("RouterOwner")

```

'End Site IP / Networking Info

' Ticket Info

'-----Check to make sure we are looking at a ticket-----

IF TicketNumber <> "" THEN

```

TicketNumber           =      rs("TicketNumber") ' Why not
SiteID                 =      rs("SiteID")
OrderDescription       =      rs("OrderDescription")
OrderNotes             =      rs("OrderNotes")
BillingNotes           =      rs("BillingNotes")
Journal               =      rs("Journal")
ConsultantID          =      rs("ConsultantID")
Consultant             =      rs("Consultant")
OrderID               =      rs("OrderID")
DateCreated           =      rs("DateCreated")
Hours                 =      rs("Hours")
NextCallDate          =      rs("NextCallDate")
NextCallTime          =      rs("NextCallTime")
OrderNumber           =      rs("OrderNumber")

```

References

1. Cisco Press. Cisco Systems San Jose, CA "Cisco Product Guide" Quick Reference Guide September 2002, 1-3
2. Friezell, Scott. Network Consulting Team, Reynolds and Reynolds Personal interview. April 31, 2003
3. Geonetta, Sam. Multimedia Advisor, College of Applied Sciences Personal interview. April 10, 2003.
4. Microsoft Corp. "Internet Explorer Guide and Reference Website". <http://www.microsoft.com/windows/ie/default.asp> September 20, 2003
5. Moseley, Robert, "Software Firewalls". Maximum P.C. July 2003. 18-19
6. Mysql Website Administrator "Mysql Database Website" <http://www.mysql.com/products/mysql/index.html> October 1, 2003
7. Reynolds and Reynolds Corp. Inside Sales Dept. April 1, 2003
8. Roberts, Paul. "Cisco warns of vulnerability in IOS". *NetworkWorld*. July 21, 2003. 18
9. Said, Hazem, Senior Design Advisor, College of Applied Sciences Personal interviews. Fall 2003, Winter 2004, Spring 2004
10. Stockman, Mark, Networking Advisor, College of Applied Sciences Personal interview. October 9, 2003.
11. Unknown, "Toms Hardware Guide" <http://www6.tomshardware.com/mobile/20030711/index.html> July 11, 2003
12. Wade, Matt, "Codewalkers database forum website". <http://codewalkers.com/>. April 30, 2003
13. World of Windows Networking Administrator "World of Windows Networking Website" <http://www.wown.com/> October 1, 2003