

Job Tracking and Timesheet Entry Application

By

Michael B. Jones

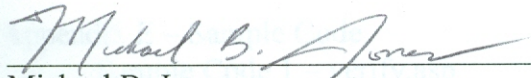
Submitted to

the Faculty of the Information Engineering Technology Program
in Partial Fulfillment of the Requirements
for

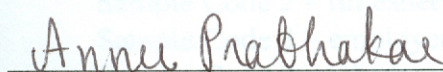
the Degree of Bachelor of Science
in Information Engineering Technology

© Copyright 2001 Michael B. Jones

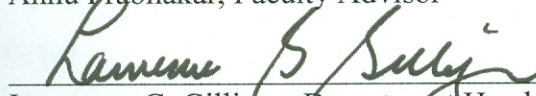
The author grants to the Information Engineering Technology Program permission to reproduce and distribute copies of this document in whole or in part.


Michael B. Jones

5/29/2001
Date


Annu Prabhakar, Faculty Advisor

05/29/01
Date


Lawrence G. Gilligan, Department Head

5/29/01
Date

Table of Contents

Section	Page
Table of Contents	i
List of Illustrations	ii
Abstract	iii
1. Statement of the Problem	1
2. Description of the Solution	3
2.1 Overview of Features	3
2.2 User Profile	4
3. Objectives of the Project	5
4. System Design and Development	7
4.1 Development Specifics	8
4.2 Budget	11
4.3 Timeline	12
5. Proof of Design	12
6. Conclusions and Recommendations	28
Appendix A – Application Flowchart	30
Appendix B – Connection Drawing	31
Appendix C – Database Design	32
Appendix D – Database Diagram	34
Appendix E – Sample Code	35
Sample Code 1 – verify.asp	35
Sample Code 2 – timesheet.asp	37
Sample Code 3 – employee.asp	46
Sample Code 4 – jobreport.cls	51
Notes	57
References	58

List of Illustrations

Figure	Page
Figure 1. The Welcome and Login screen.	12
Figure 2. The Home page of the application.	13
Figure 3. The new timesheet interface.	14
Figure 4. A successful timesheet submission.	15
Figure 5. Selecting the timesheet to view.	16
Figure 6. Viewing the past timesheet.	17
Figure 7. Selecting an Employee report.	18
Figure 8. Employee Detail of the Employee report.	19
Figure 9. Job Report for an employee.	20
Figure 10. Final screen of the Employee Job report.	21
Figure 11. The Employee Billable report.	22
Figure 12. The Administration console.	23
Figure 13. Modifying a job.	24
Figure 14. Employee list.	25
Figure 15. Viewing timesheets.	26
Figure 16. Changing the color scheme.	27

Abstract

This report covers the “what and why” of the deliverables for the Job Tracking and Timesheet Entry Application. The purpose of this project is to make the necessary business routines of timesheets entry and job tracking more user-friendly, more streamlined, and less wasteful. This application is developed for a small business in Cincinnati, OH, Hammond Communications Group. The users of the old system believed it to be rigid, inaccessible, and wasteful of time. It was there unhappiness with the old system that prompted the development of this application.

Included here is the rationale for using each of the technologies chosen, including Visual Basic, ActiveX, COM, ADO, OLE DB, SQL Server, and Active Server Pages. Additionally, a complete description of the final product is presented.

Job Tracking and Timesheet Entry Application

1. Statement of the Problem

The system for tracking job information and handling time reports now in place at Hammond Communications Group is neither reliable nor expedient. Timesheets are filled out on paper and faxed from Cincinnati to the accounting office in Lexington where they are manually entered into a program running on a UNIX computer system. The chance for data entry errors is high, depending on the abilities of the user. Job tracking is handled by manually combining a number of reports into one or more spreadsheets. This is a tedious and time-consuming process. The current system has been in place at the company for many years. It is now time to move from this system to one that benefits the user and does not cause so much aggravation.

Employees must fill out time reports daily. This is a multi-step process:

1. The employee must make a copy of the master timesheet if they don't already have a copy.
2. The basic information at the top must be filled in, including name, employee number, department, and date.
3. The time matrix is then filled in. This includes job code, job phase code and description, and hours worked.
4. The employee makes a hard copy of the timesheet for his or her records.
5. The timesheet is faxed to the accounting office to be entered into another computer system.

This is a laborious process to complete every day. In speaking to 14 employees that ranged from upper management to production crew, I found that it takes an average of

10-15 minutes every morning to complete this. They have a strong desire to find a better way.

One alternative that was proposed in the past was going to an Excel spreadsheet time report that would then be e-mailed to the accounting office. This would seem to be much less wasteful since the employees would no longer have to fax hard copies. It turns out that the accounting office would nullify this assumed benefit by having to print hard copies of the spreadsheets anyway. One reason for this is because they eventually have to manually enter the data into the UNIX system. Also, they do not want the prospect of trying to handle the thousands of Excel files that would accumulate in just one year of approximately 70 employees sending daily files. Accounting would rather deal with thousands of hard copies than multitudes of digital files that cannot be easily retrieved once archived.

Job management is a problem with which nearly all businesses must deal. Having a tool that can instantly report budget, hours worked, employees assigned, dollars under/over budget, resources in use, etc. is invaluable to a manager dealing with multiple projects. Currently, job tracking at Hammond is mostly a paper-based process. In today's job environment, this is a cumbersome and obsolete means for tracking projects. A typical report entails combing a number of files for relevant information and combining it into one report. Of the individuals I spoke to, this generally takes between 20 minutes to over an hour depending on how far back the project goes. A more centralized, automated process could greatly reduce the time currently required to track projects. I predict a report that used to take 30 minutes to compile manually will take only a few minutes with this new system.

Since a good deal of job tracking information can be gleaned from employee time reports, it makes sense to tie these two crucial tasks into one enterprise-wide system. The two are disparate in the old system, but the new system brings these two together under one umbrella to create a useful business tool.

2. Description of the Solution

To increase reliability and productivity, and to decrease the frustration of manually sorting time reports, an intranet-based application for the handling of timesheet input and job tracking was developed. The system has the following features:

- User authentication
- User-friendly graphical interface
- “Smart”-forms, to make data entry easier
- User help, prompts, and input validation
- Automatic archiving ability
- Centralized administrative backend
- Reporting tools.

2.1 Overview of Features

1.) User authentication – each employee has a unique username/password combination that he or she uses to log into the system. Besides giving the user an identity, this provides security for the application so that no user may view the private information of another.

2.) User-friendly graphical interface – to make the application as inviting as possible, an attractive, logical user interface was developed for the front end.

Intuitive navigation is stressed. All buttons and links are clearly and sensibly labeled.

3.) “Smart”-forms – this refers to the ability of a form to automatically fill in certain fields depending on input from the user. For example, if the user enters a job phase code of “9040”, the system will automatically fill in the description of this phase code as “Sales/Marketing Support”.

4.) User help, prompts, and input validation – to augment user-friendliness, JavaScript code guides users with validation warnings and help messages. In addition, every page includes a link to a help screen for that specific page.

5.) Automatic archiving ability – to facilitate retrieval of past timesheets, every user submission is automatically archived. The user has the ability to pull and review any timesheet entered, whether it be from last week or last year.

6.) Centralized administrative backend – through a separate administrative console, managers and accountants are able to use the system to run reports, change phase codes, add/modify/delete job information, add/modify/delete job phases, etc.

7.) Reporting tools – from within a privileged area, managers are able to track job phases, hour worked, budgets, billable hours, and other important job information.

2.2 User Profile

The users of this system range from production crew (camera operators, audio engineers) to IT professionals (digital artists, programmers). A certain level of computer proficiency is expected, but the system does not actually require much computer or Web savvy except for those who require it. Because of the nature of the application, the less

computer proficient employees of the company are only concerned with the more basic functions. The individuals who use the more complex functions of the application are more proficient with the computer by nature of their positions.

3. Objectives of the Project

The system can be accessed only from within the Hammond Communications Group intranet. This ensures the security of the data from hackers outside of the firewall. Employees use a Web browser to access a Web page on the intranet and are presented with a login form. They enter their username and password to proceed. Once they are authenticated they are given a menu based upon their level of clearance. If the login fails, they are prompted to try again. Upon the third failed try, they are no longer allowed to attempt to login without shutting down their browser. Cookies are used remember the user's name and username upon repeat visits, saving the trouble of having to type in a username every time. The password, however, remains hidden and is always required.

The main menu contains a number of choices. Depending on the users' login information, they receive access to different parts of the system. There are three levels of authentication: basic users, advanced users, and system administrators. Basic users have the ability to enter and review their own time reports only. Advanced users are able to enter and review their own time reports, review other employees' time reports and run a number of job tracking reports. System administrators have access to all areas including the ability to add/modify/delete job information, phase codes, users, etc.

The basic user may login to the system and choose to enter a new time report. Because he or she is already authenticated with the system, their name, employee number, and department are filled into the new timesheet form. The date is automatically

filled-in but may be changed. The user selects a job number from a drop-down menu listing that is populated from the table of current jobs. He or she selects a job phase code from a drop-down menu listing that is populated from the table of phase codes. He or she selects the number of hours worked on this phase for this job. Selecting the button labeled "Next" populates the rest of the data and moves to the next row. Once the user has completed entering data for that day, a submission of the completed form writes the values into the database. From within the database, the user can retrieve the information for review whenever he or she likes.

Besides entering their own time reports, advanced users are able to monitor other users' time. This tool is for managers to determine how much time their employees are working on specific projects. A number of reporting tools are available to the advanced user. The manager can select a job from a drop-down menu listing that is populated from the table of current jobs and run a report that returns hours worked and by whom for that project. The system reports how under or over the budget the project currently is based on these hours. Another report indicates how "billable" the employee is based upon billable hours versus total hours reported. Other information can be tracked as well from within the "Comments" field.

The system administrators maintain the system by keeping the various database tables up-to-date. They have administrative tools to add/modify/delete data from all of the database tables – employee list with passwords, active and inactive jobs, job phase codes, and phase code descriptions. The administrator can also maintain the Announcements section to displays news and notes. They can choose to display the announcement immediately or at any predetermined time in the future.

Another important tool for administrator use is the timesheet console. When the administrator chooses a date, the system displays all the employees who submitted a timesheet for the date in one column, along with those employees who did not submit a timesheet in another column. Clicking on an employee's name will either display that user's timesheet or send him or her a reminder e-mail to submit a timesheet for the date, depending on which column the name appears.

When any user (basic, advanced, or administrator) is finished, he or she clicks the ever-present link in the menu bar to logout from the system. This ensures that no unauthorized individuals can get into the system by simply clicking the browser's "Back" button.

4. System Design and Development

The application operates on a Microsoft Windows 2000 server running Internet Information Server (IIS) 5 Web server. A Relational Database Management System (RDBMS) is needed to handle the database tables and stored procedures. For this, Microsoft SQL Server 7.0 (5) is employed. The Web server and database server exist as two separate physical computers. This substantially increases performance.

There are a number of benefits of running on the Microsoft platform. These include:

- Microsoft is the industry's best known name and most widely used platform
- The technology is ubiquitous
- There are many skilled developers for this platform
- The cost/performance ratio is top-notch.

The user will need a computer with access to the intranet and a browser, preferably Internet Explorer 5 or newer. All the employees of Hammond meet these requirements.

The system is comprised of 125 files, containing approximately 6500 lines of code. The database contains nine tables and 25 stored procedures. There are approximately 60 help screens.

4.1 Development Specifics

The system was developed using the most current Web and database technologies to thwart obsolescence. In general, Visual Basic 6 was used to write ActiveX components. These Dynamic Link Libraries (DLLs) are deployed on the intranet server as COM objects to access stored procedures within our database system. The database is accessed via ActiveX Data Objects (ADO) and OLE DB establishes the connections. Microsoft's Active Server Page (ASP) technology provides the framework for the system.

The following is a more thorough discussion of the technologies mentioned above.

Visual Basic 6 – Microsoft Visual Basic development system is the most productive tool for creating fast business solutions for Windows and the Web. A comprehensive, rapid application development environment helps developers quickly create and deploy client/server applications, plus easily program for the Internet using Visual Basic programming tools and techniques. (4) Provided benefit: industry standard architecture and Rapid Application Development (RAD).

ActiveX components – A unit of executable code, such as an .exe, .dll, or .ocx file, which follows the ActiveX specification for providing objects. ActiveX technology allows programmers to assemble these reusable software components into applications and services. Developers can create components that encapsulate business transactions, and combine these with generic components. Reusing tested, standardized code in this fashion is called component software development. (1) Provided benefit: securely encapsulates business logic and allows for reusability.

COM Objects – COM is a platform-independent, distributed, object-oriented system for creating binary software components that can interact. COM is the foundation technology for Microsoft's OLE and ActiveX technologies. COM objects can be created with a variety of programming languages, especially Visual Basic and Visual C++. These provide programming mechanisms that simplify the implementation of COM objects. These objects can be within a single process, in other processes, even on remote machines. (6) Provided benefit: increase in performance within remote applications.

ActiveX Data Objects – ADO is Microsoft's strategic, high-level interface to all sorts of data. ADO provides consistent, high-performance access to data, whether one is creating a front-end database client or middle-tier business object using an application, tool, or language. ADO is the single data interface a developer needs to know for 1- to n-tier client/server and Web-based data-driven solution development. (7) Provided benefit: latest and greatest interface for data access.

OLE DB – ADO is designed as an easy-to-use application level interface to Microsoft's newest and most powerful data access paradigm, OLE DB. OLE DB provides high-performance access to any data source, including relational and non-relational databases, e-mail and file systems, text and graphics, custom business objects, and others. ADO is implemented with a small footprint, minimal network traffic in key Internet scenarios, and a minimal number of layers between the front-end and data source-all to provide a lightweight, high-performance interface. ADO is easy to use because it is called using a familiar metaphor - the OLE Automation interface, available from just about any tool and language on the market today. (2) Provided benefit: provides substantial performance benefit over ODBC connections.

Active Server Pages – ASP is the server-side execution environment in IIS that enables ActiveX server components to run on the server. By combining scripts and components, developers can create dynamic content and powerful Web-based applications. Web pages can be customized for each user on-the-fly, based upon their actions or requests. ASP provides a very open development environment, with support for both Microsoft VBScript and JScript along with PerlScript. (3) Provided benefits: RAD and industry standard.

4.2 Budget

The system requires hardware and software elements. The software can be broken down into the applications needed for deployment and the applications needed for development.

Intranet Server	
CPU	Athlon 1 GHz
Memory (RAM)	384 MB
Hard disk space	18 GB
Optical	48X CD-ROM
Removable	100 MB Zip
Bandwidth	10/100 Mb Ethernet
Approximate cost	\$3,000

Database server	
CPU	Athlon 1 GHz
Memory (RAM)	512 MB
Hard disk space	36 GB Total RAID 5
Optical	48X CD-ROM
Removable	100MB Zip
Bandwidth	10/100Mb Ethernet
Approximate cost	\$3,500

Required software: Deployment

Software	Name	Cost ¹
Web server operating system	Microsoft Windows 2000 Server	\$799.00
Web server	Microsoft IIS 5	Free
Database server operating system	Microsoft Windows 2000 Server	\$799.00
Database server	Microsoft SQL Server 7.0	\$1,259.00

Required software: Development

Software	Name	Cost ²
RAD tools	Microsoft Visual Studio 6 Enterprise	\$1339.00
Graphics	Adobe Photoshop 6.0	\$561.00

Budget Total: \$11,257.00

4.3 Timeline

The following is a guideline for project milestones.

Project Stage	Date
Gathering of initial system requirements	November 15, 2000
Presentation of proposal	December 7
Gathering of complete design content	January 1, 2001
System design and database schema design	January 1 – February 15
Programming Quick Prototype	February 1 – March 4
Draft Proof of Concept Report	February 9 – February 15
Develop of Proof of Concept Presentation	March 1 – March 7
Demonstration of Prototype	March 8
Expand prototype into full application	March 8 – May 7
Preparation of documentation	May 1 – May 21
Testing/Debugging on development system	May 7 – May 15
Transition to production servers	May 15-16
Production testing	May 16-24
Final deployment	May 25
Final Presentation	June 7

5. Proof of Design

The goal of the project was to develop a system that increases reliability and productivity, and decreases the frustration of manually sorting time reports. The following pages show several screens from the application. These screenshots, along with their explanations, attest to the completion of these project objectives.

Please refer to Appendix E to peruse sample code used to accomplish the project objectives.

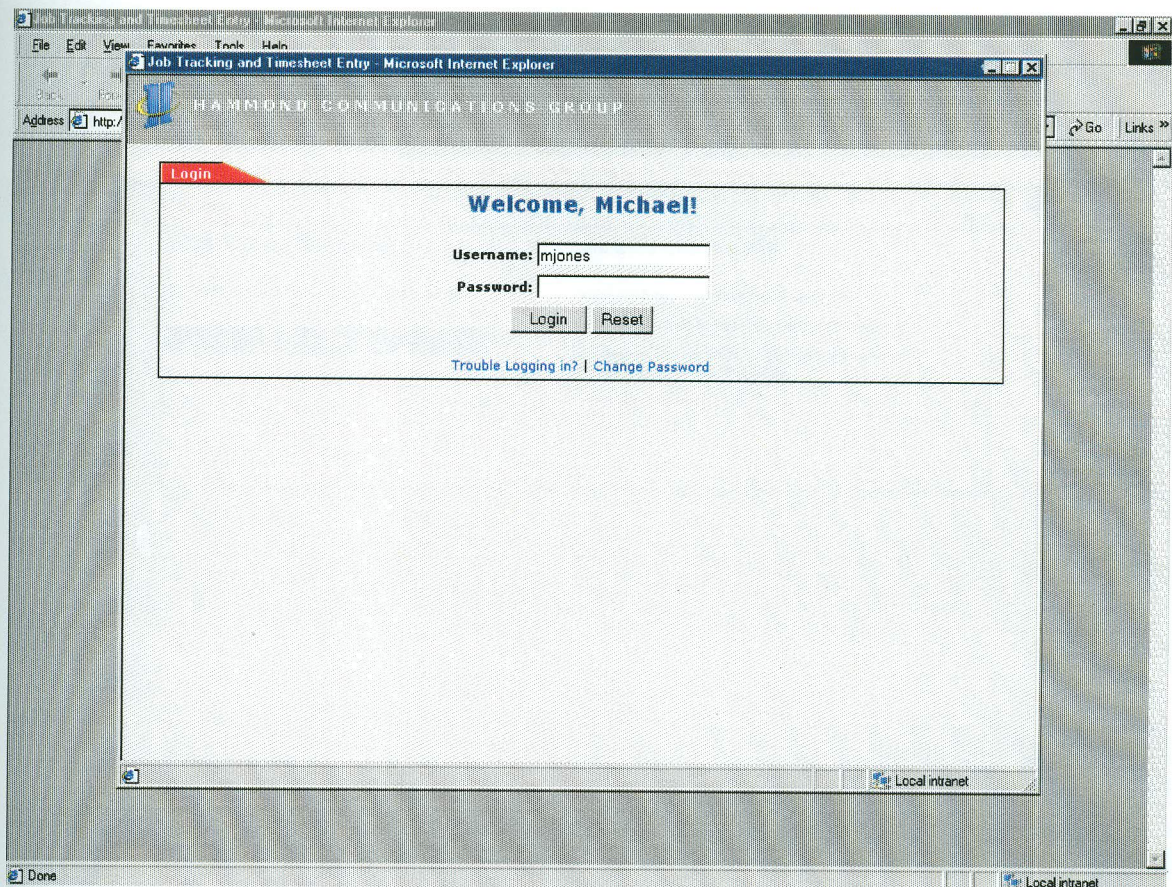


Figure 1. The Welcome and Login screen.

Figure 1 shows the login screen for the Job Tracking and Timesheet Entry application. All users – basic, advanced, and administrator – log in through this interface. Notice the use of cookies in correctly identifying the author with a “Welcome, Michael!” message. Additionally, the cookie has filled in the username with “mjones”, saving the user the trouble of entering this every time.

This interface also has links for login help and to change password.

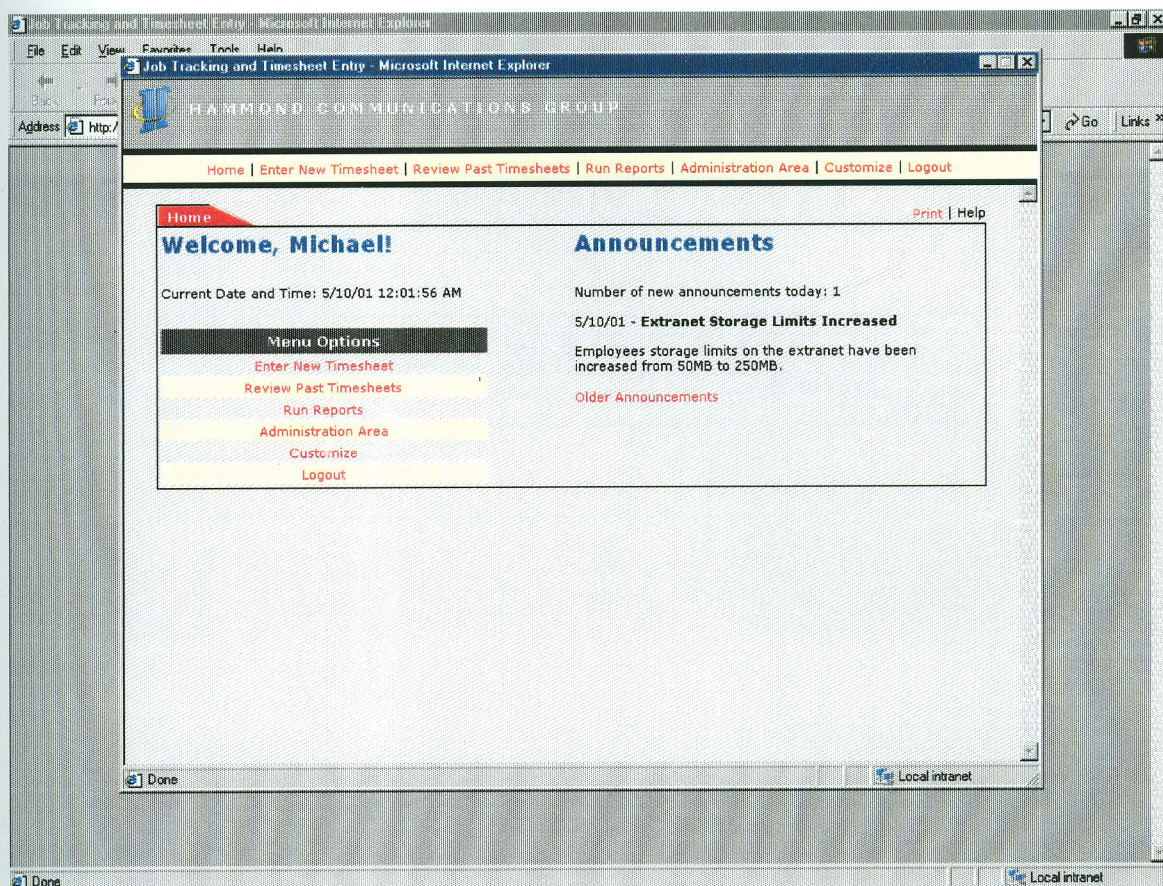


Figure 2. The Home page of the application.

Figure 2 shows the home page of the application once the user has successfully logged in. The navigation menu is located in two locations – near the top, horizontally, and on the left, vertically. Because this user has an access status of 3 (administrator), he or she has menu links to “Run Reports” and “Administration Area”. Users with access status of 1 (basic) would not see these additional links. Advanced users (access level 2) would have “Run Reports”, but not “Administration Area”.

Notice the “Announcements” on the right side. Administrators have the ability to post notes, reminders, and news in this section from within the Administration Area.

Job Tracking and Timesheet Entry - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address http://

Go Links

Home | Enter New Timesheet | Review Past Timesheets | Run Reports | Administration Area | Customize | Logout

New Timesheet Print Help

Employee Name: Michael Jones
Employee #: 48
Department: Interactive Media

Timesheet date: 5/9/01 (change date)

Job Number	Name	Client	Phase Code	Phase	Hours
24510	Web site	Bob's Bar	4020	Programming - C++, VB, Database	2.0
22014	Audio Tracks	Marker's Mark	4025	Programming - HTML, Flash	6.5
Total Hours:					8.5

0.5 Next

Finished Start Over

Done Local intranet

Figure 3. The new timesheet interface.

Figure 3 shows the interface for entering a new timesheet. This timesheet has been filled out and is ready to be submitted for processing. The user would click the “Finished” button for final processing, or may reset the screen by clicking the “Start Over” button.

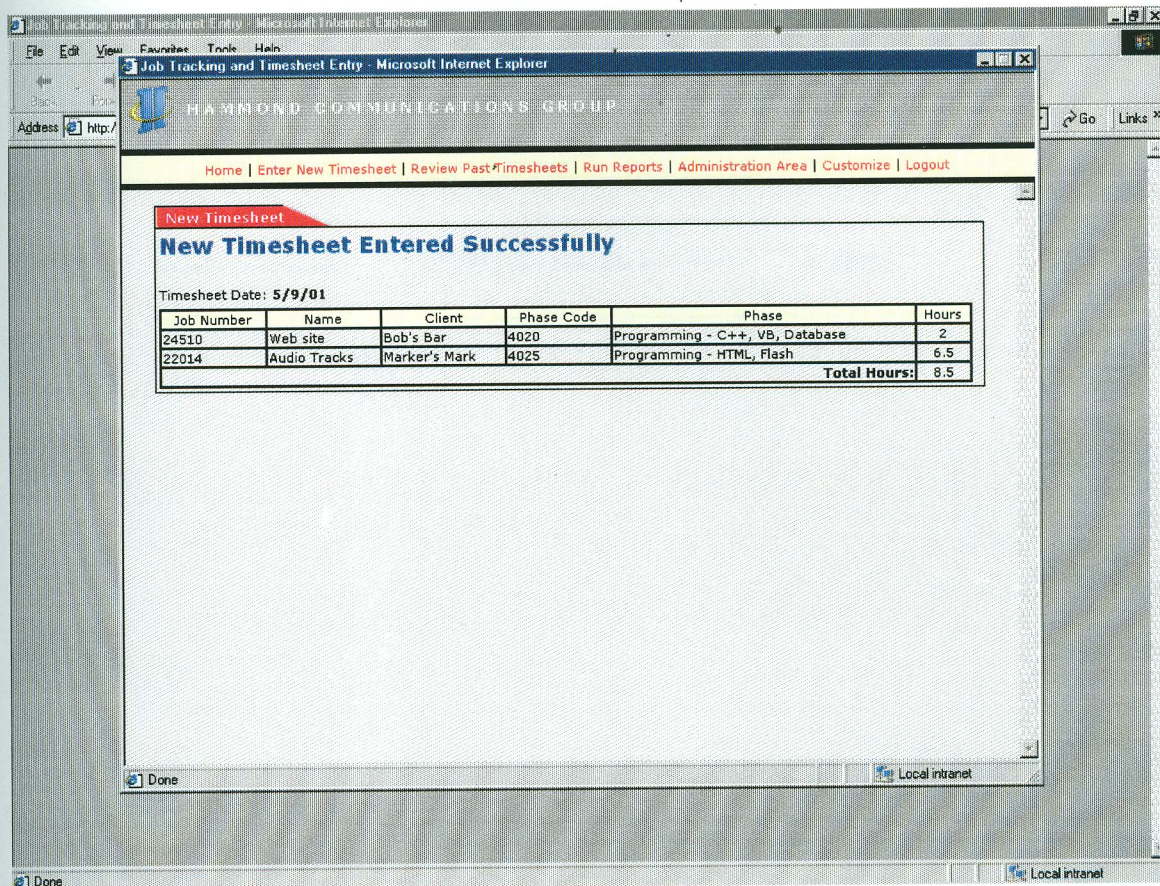


Figure 4. A successful timesheet submission.

When the user clicks the "Finished" button, he or she is presented with a screen similar to Figure 4. This tells the user that the submission was successful and shows him or her what was entered.

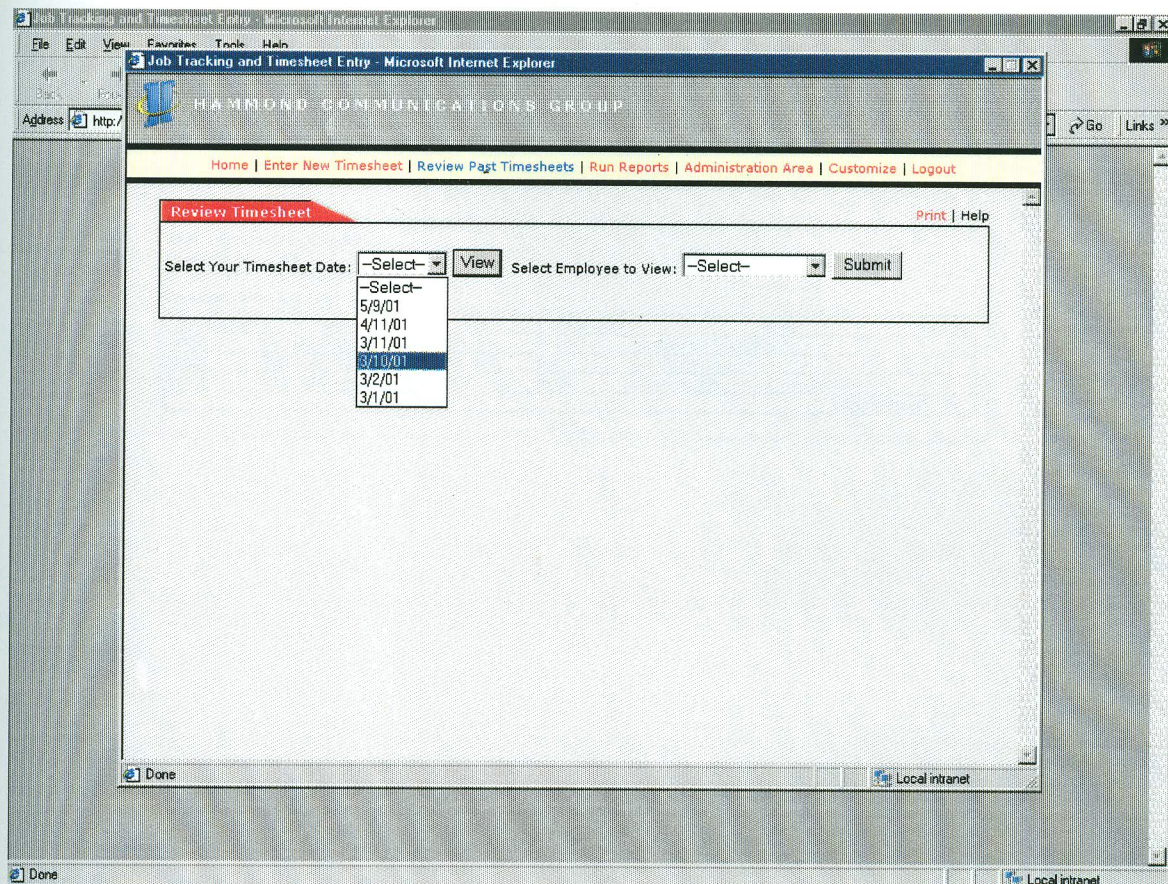


Figure 5. Selecting the timesheet to view.

When a user selects “Review Past Timesheets”, the interface in Figure 5 is displayed. The user selects the date of the timesheet he or she wishes to review and clicks the “View” button.

Notice the other dropdown menu on the right side. Since this user is logged on as an administrator, he or she can view any employee’s timesheet as well as his or her own.

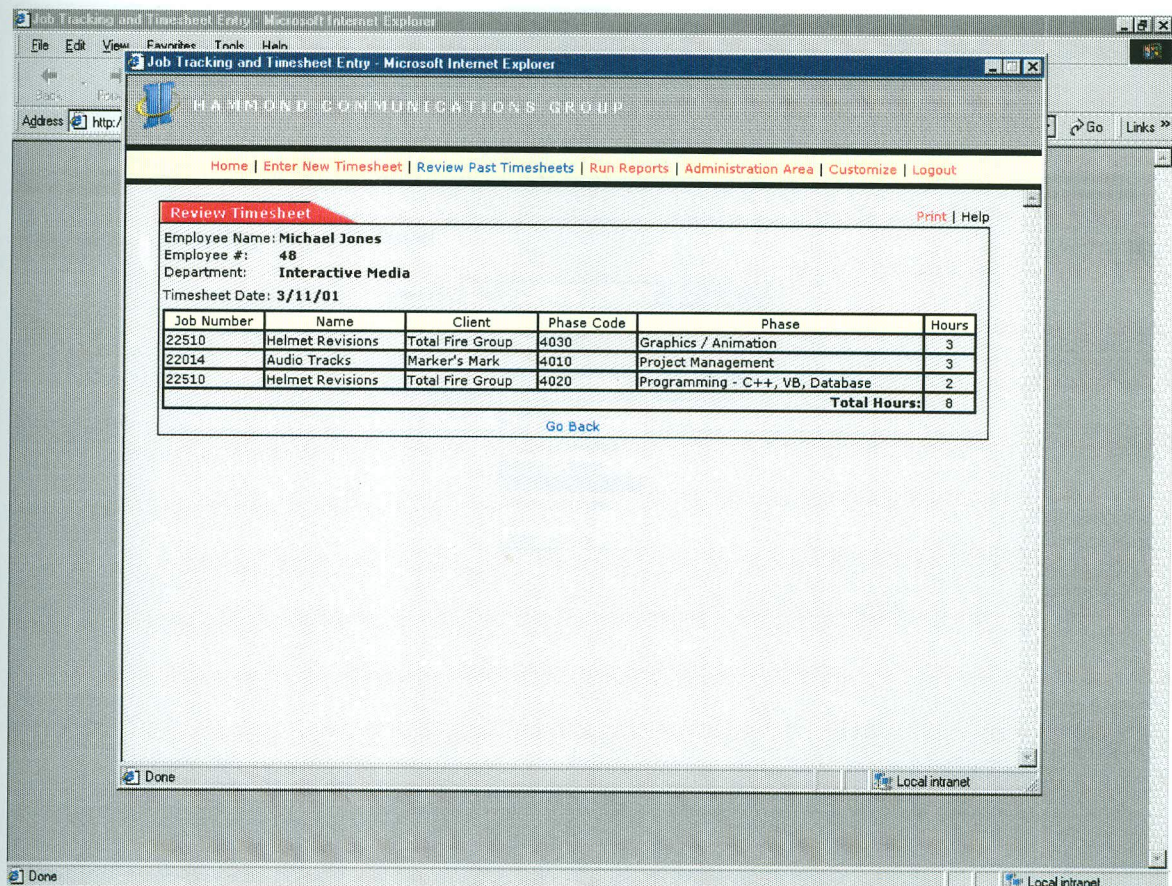


Figure 6. Viewing the past timesheet.

Figure 6 shows the result of the user selecting his or her timesheet from 3/11/2001 to review. Notice the “Print | Help” links in the upper right corner of the lower frame. These links are available on every page for instant access to printing this page or for a help about this particular screen.

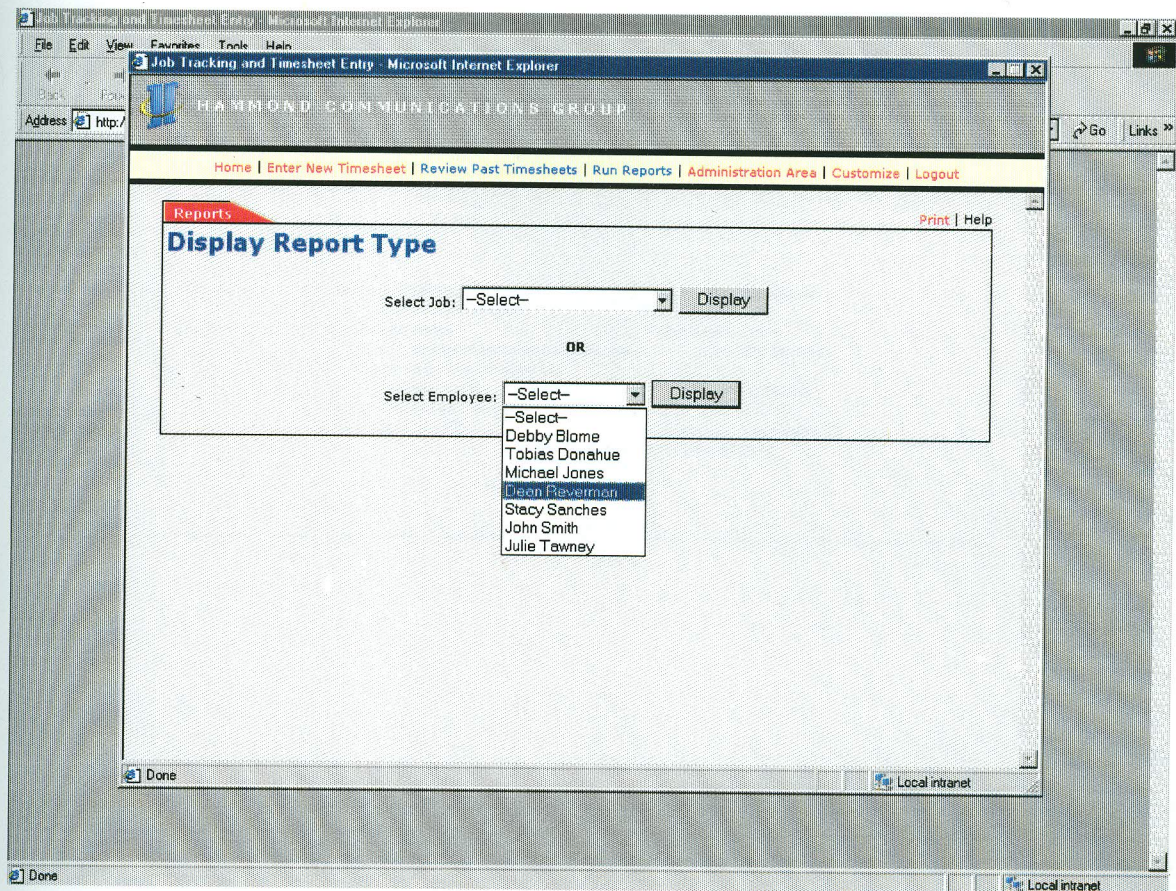


Figure 7. Selecting an Employee report.

Figure 7 shows the main page of the "Run Reports" section. Users with access to this page can run either job or employee reports.

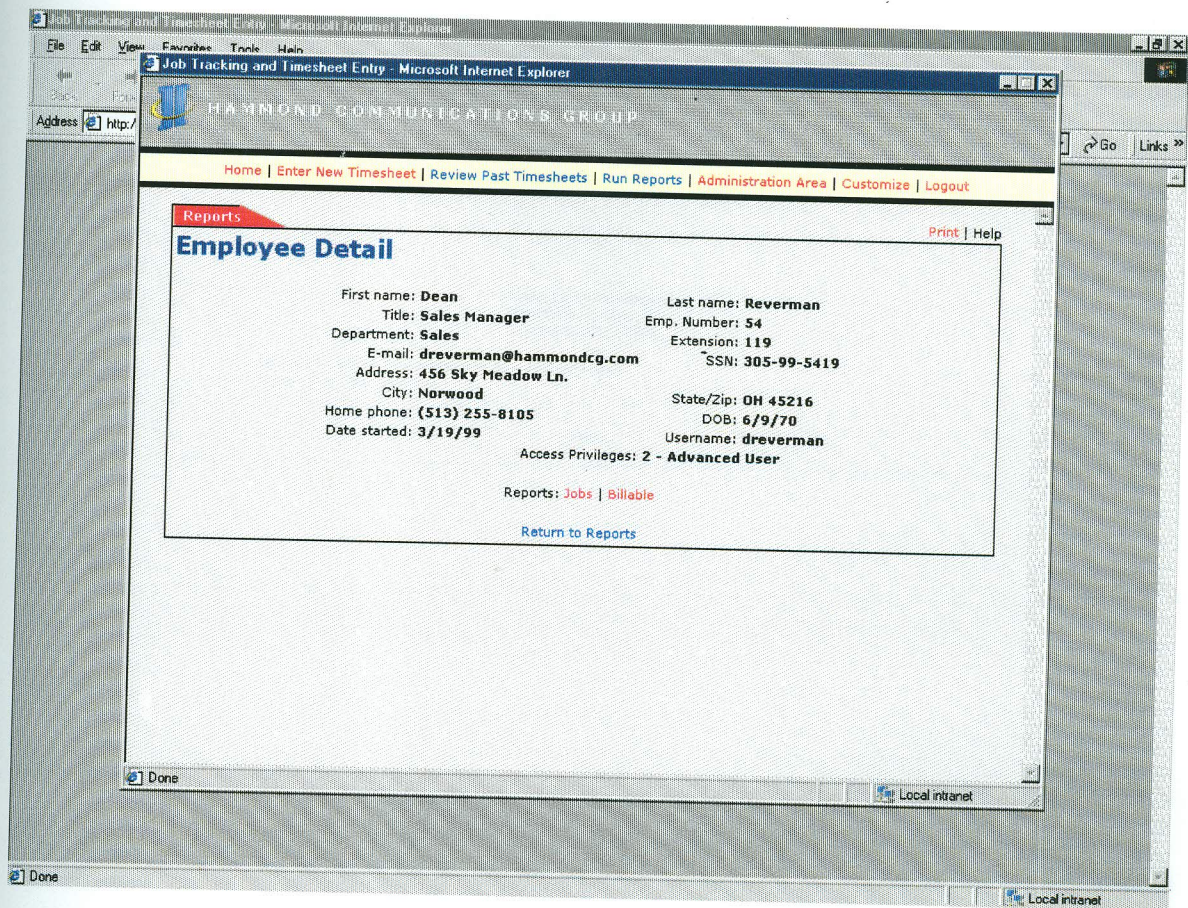


Figure 8. Employee Detail of the Employee report.

Figure 8 shows the first page of the employee report for Dean Reverman. The two links near the bottom of the screen link to the more specific reports for “Jobs” and “Billable”.

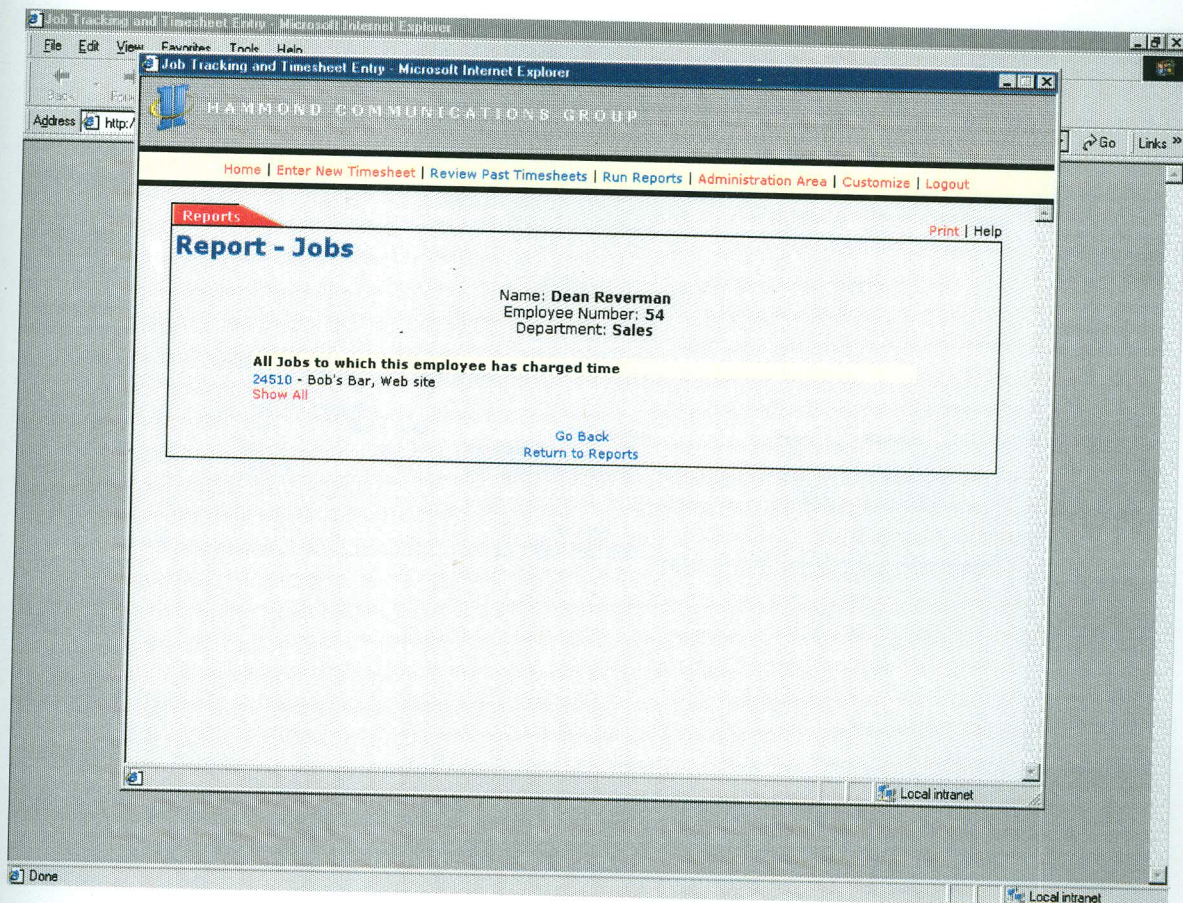


Figure 9. Job Report page for an employee.

When the user selects the "Jobs" report from the Employee Detail page (Fig. 8), he or she is shown a list of all the jobs to which the employee has charged time. This employee has time charged to one job: Bob's Bar.

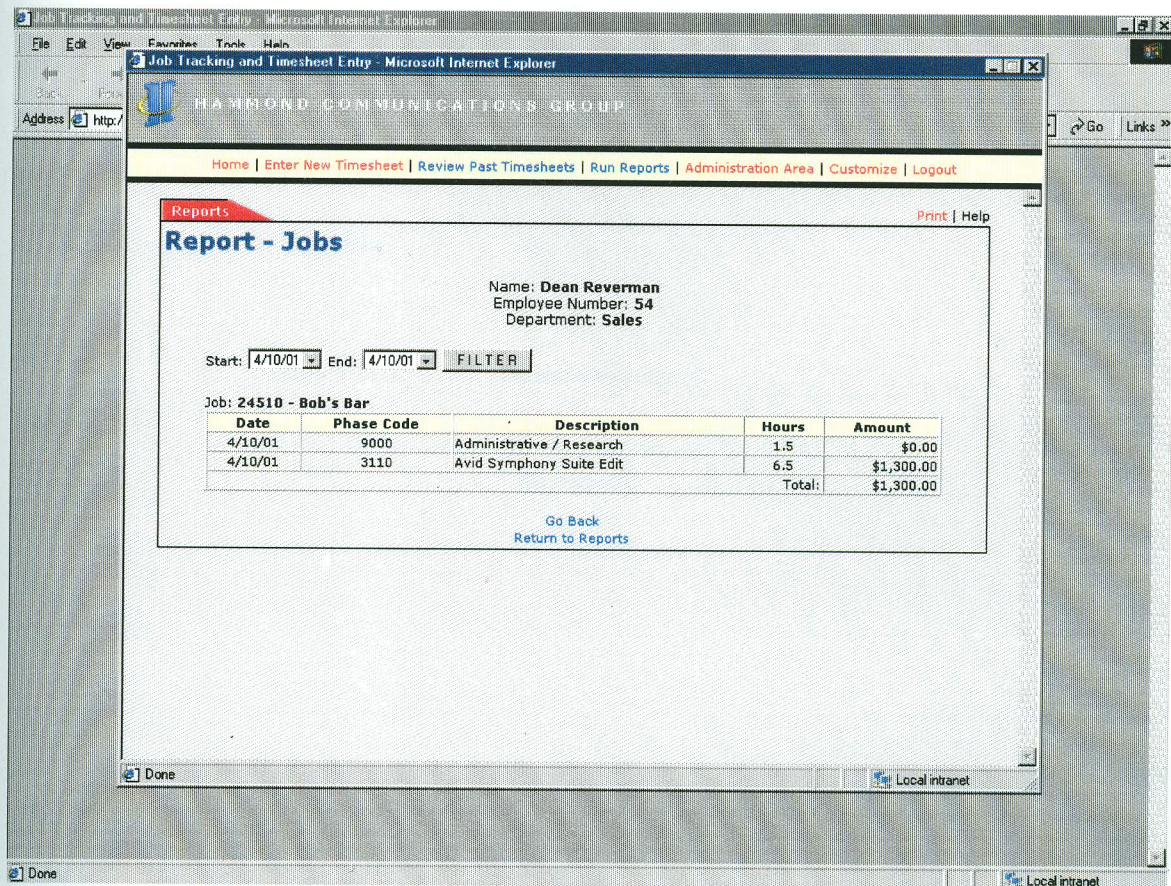


Figure 10. Final screen of the Employee Job report.

Figure 10 shows a breakdown of all the work the employee completed for the particular job. Notice that the user can filter the data displayed in the table simply by selecting the start and end dates to display and clicking "Filter".

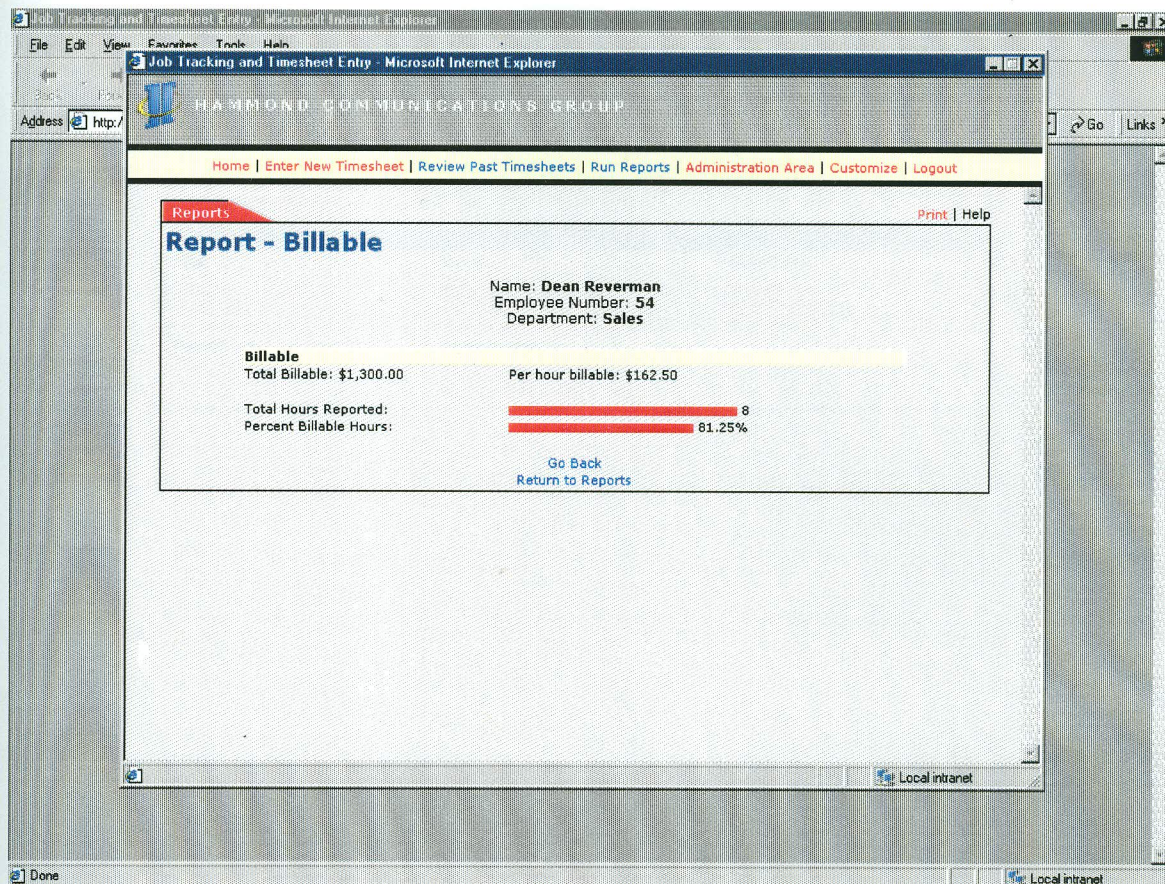


Figure 11. The Employee Billable report.

Figure 11 shows the Employee Billable report. This report displays the total amount billed by the employee, how that amount breaks down by hour, his or her total hours reported, and his or her percentage of billable hours.

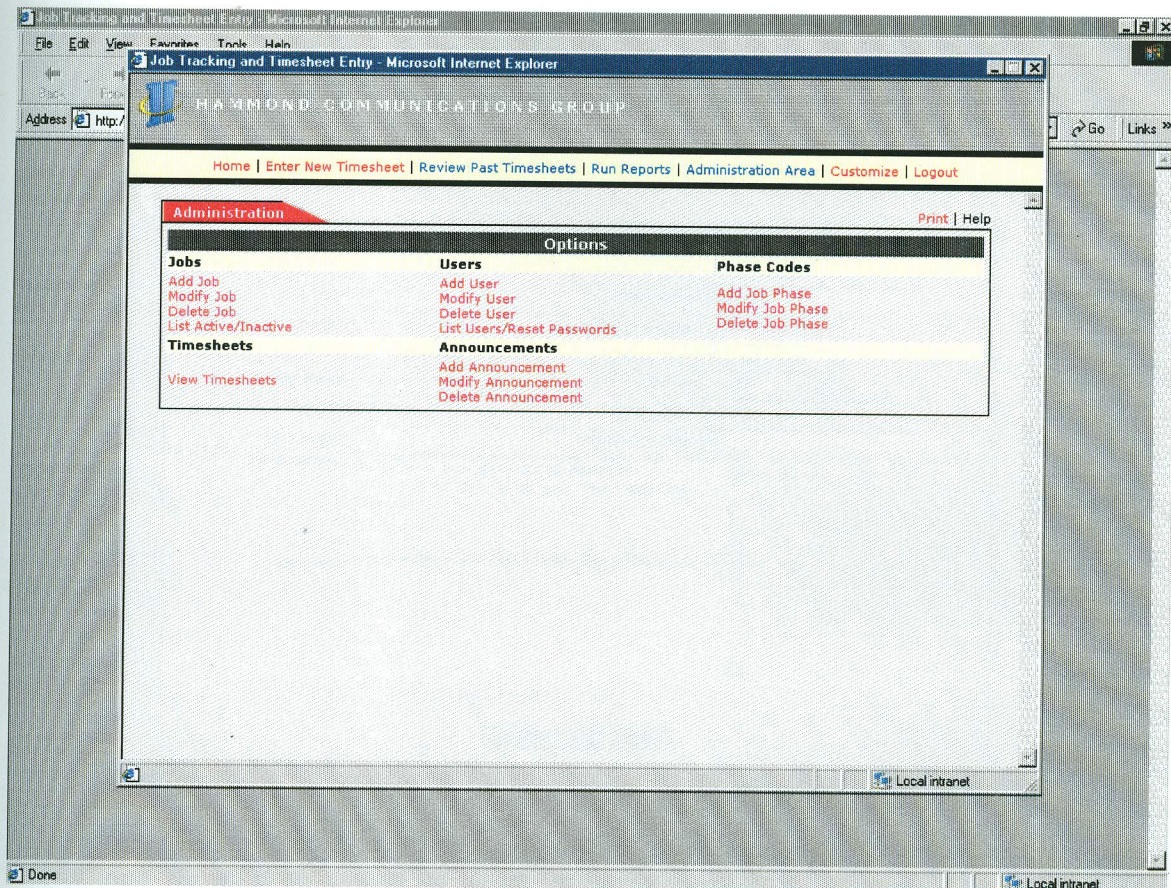


Figure 12. The Administrative console.

Figure 12 shows the main page of the Administration Area. From this menu, privileged users can add/modify/delete jobs, users, job phase codes, and announcements. They can also view all users, view timesheets, and reset passwords.

Job Tracking and Timesheet Entry - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address http://

Go Links

Home | Enter New Timesheet | Review Past Timesheets | Run Reports | Administration Area | Customize | Logout

Administration Print Help

Modify a Job

Job Number: 21594 Client Name: Stonerside

Job Name: Web site creation Account Executive: Dean Reverman

Project Lead: Michael Jones Art Director: Michael Jones

Technical Lead: Michael Jones Status: 1 - WIP

Budget: 32200 Deadline: 5/25/01

Invoiced to Date: 0 Percent Complete: 80

Job Description: Design a new site from scratch.
Similar to Marablu and Thoroughbred Corp.

Comments:

☐ Check box to archive job

Modify Job Reset

Done Local intranet

Figure 13. Modifying a job.

One of the more common tasks of the Administrator is modifying job information, shown in Figure 13. Administrators make their changes in this form and click the “Modify Job” button when they are finished. The screens for modifying users and job phases are very similar to Figure 13, the only difference being the form fields.

The screens to add jobs, users, and phases are also very similar to Figure 13, however, the add screens are blank by default.

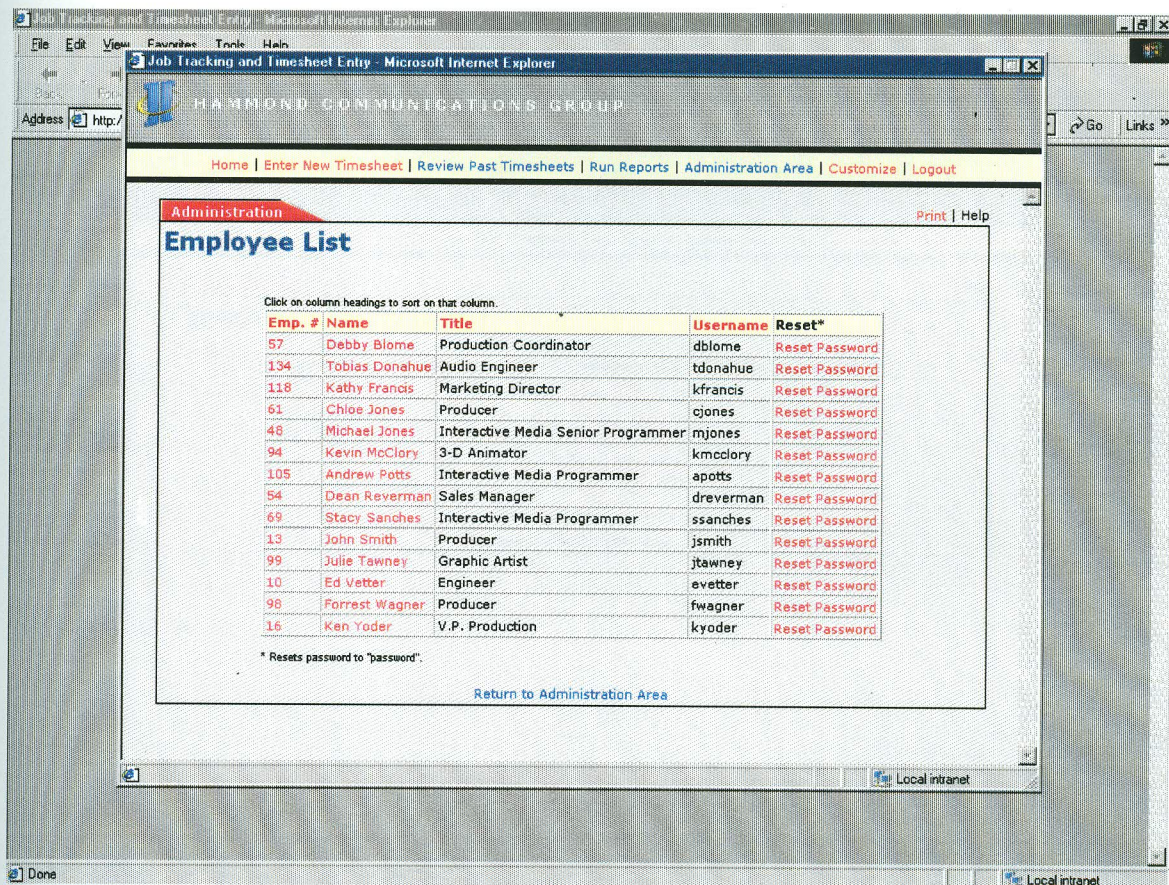


Figure 14. Employee list.

Figure 14 is the Employee List. Administrators click on the column headers to sort the list on that column. Clicking the employee number displays that employee's complete information. Clicking on a name opens the e-mail program to send that user a message, provided there is a default e-mail client on the Administrator's computer. Finally, clicking the far right column resets the user's password to "password" should he or she ever forget it. With this utility, the administrator never needs to know the user's chosen password.

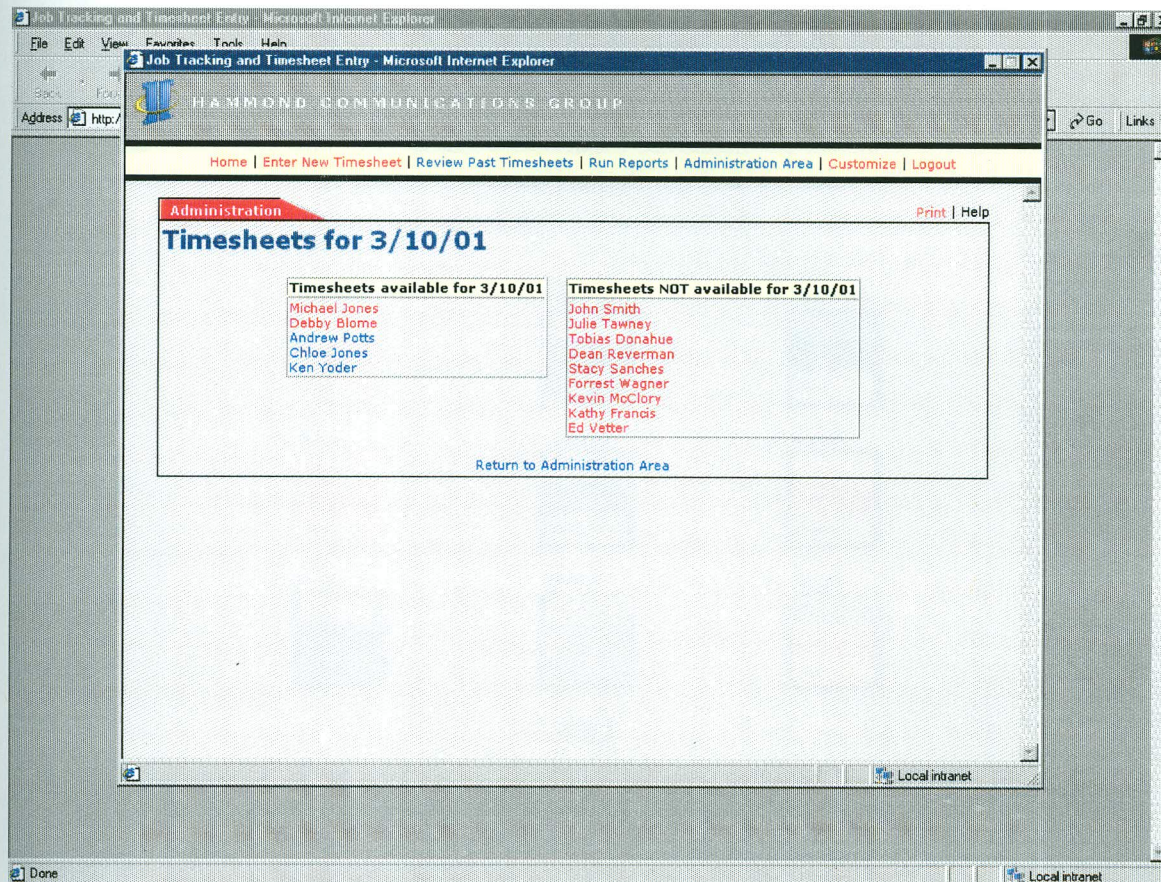


Figure 15. Viewing timesheets.

Figure 15 shows the administrator's timesheet console. This screen shows in the left column those employees who have submitted a timesheet for the given date, and those who haven't in the right column. Clicking on a name in the left column displays the timesheet. Clicking on a name in the right column automatically sends that employee an e-mail reminding him or her to submit a timesheet for the chosen date.

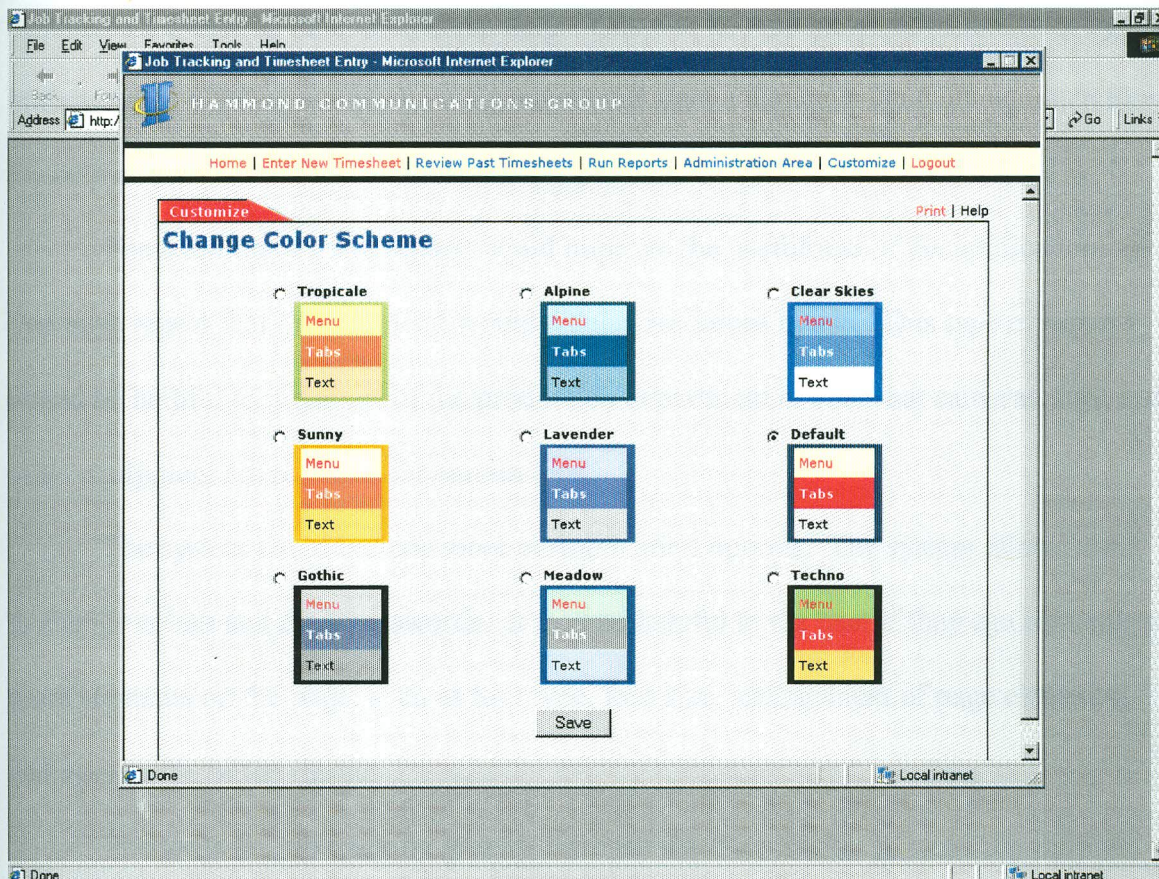


Figure 16. Changing the color scheme.

Should the user ever get bored with the default color scheme of the application, he or she can choose between nine different color schemes from the screen shown in Figure 16.

6. Conclusions and Recommendations

The completed system is an invaluable tool for administrators, managers, and regular employees alike. Because of the reduced time it will take to enter time reports, employees are more productive and less frustrated. However, there are improvements that can be made to the system.

I recommend that the help section of the system move from the current HTML version to a dynamic, database-driven help. This will provide more advanced help by introducing the ability to search on keywords and have a dynamic topic index, much like the help in many Microsoft applications.

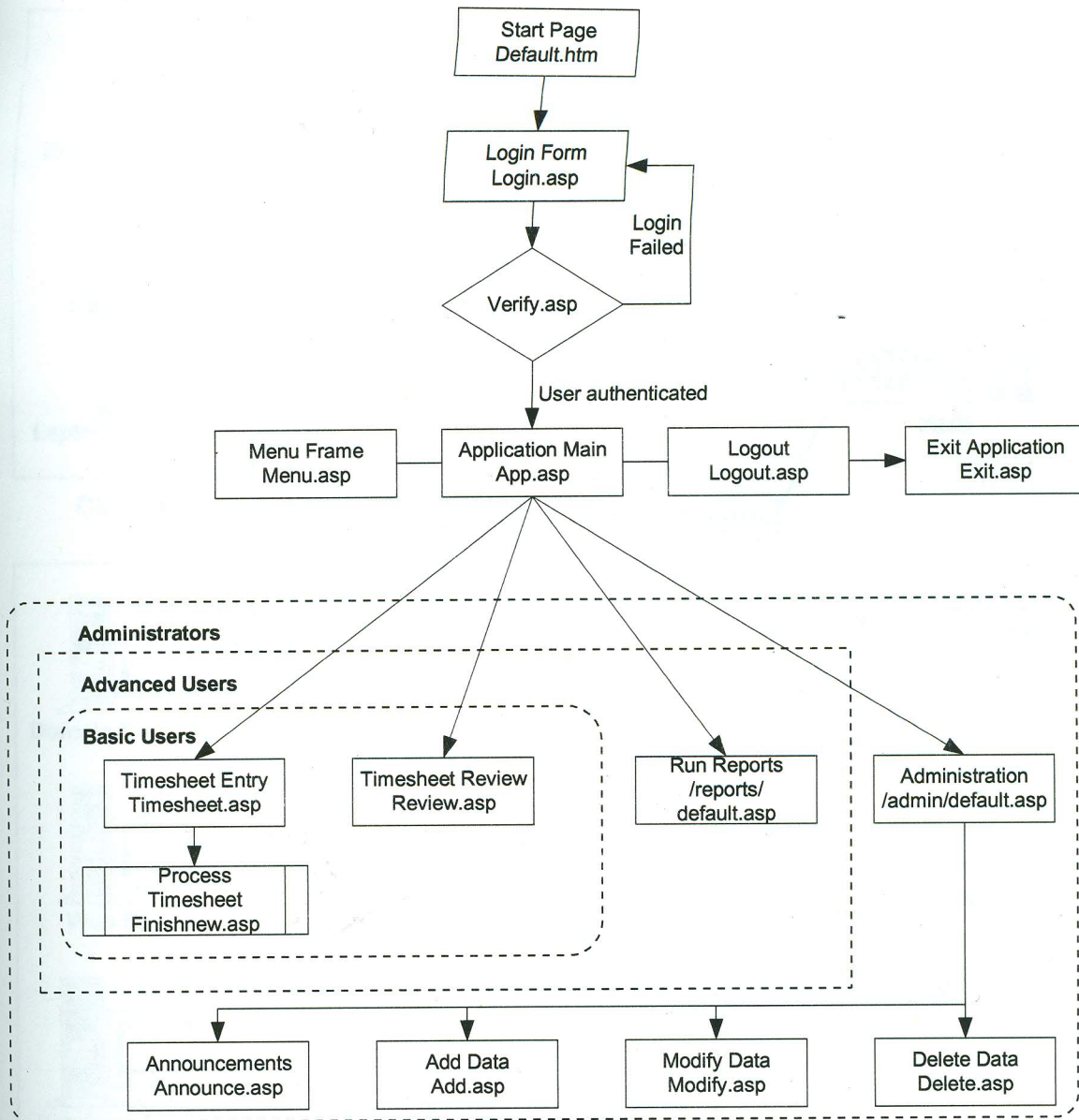
The addition of more reports could improve the usefulness of the application. One such report that I recommend would display resources in use. This type of report would be helpful to members of the production department to see what cameras, rigs, and other equipment are reserved for certain jobs.

The system could include more customization options. The system allows users to change colors and their passwords. It may be helpful to the user if they can customize more elements on the page, such as font type, font size, arrangement of page elements, and orientation of the menus.

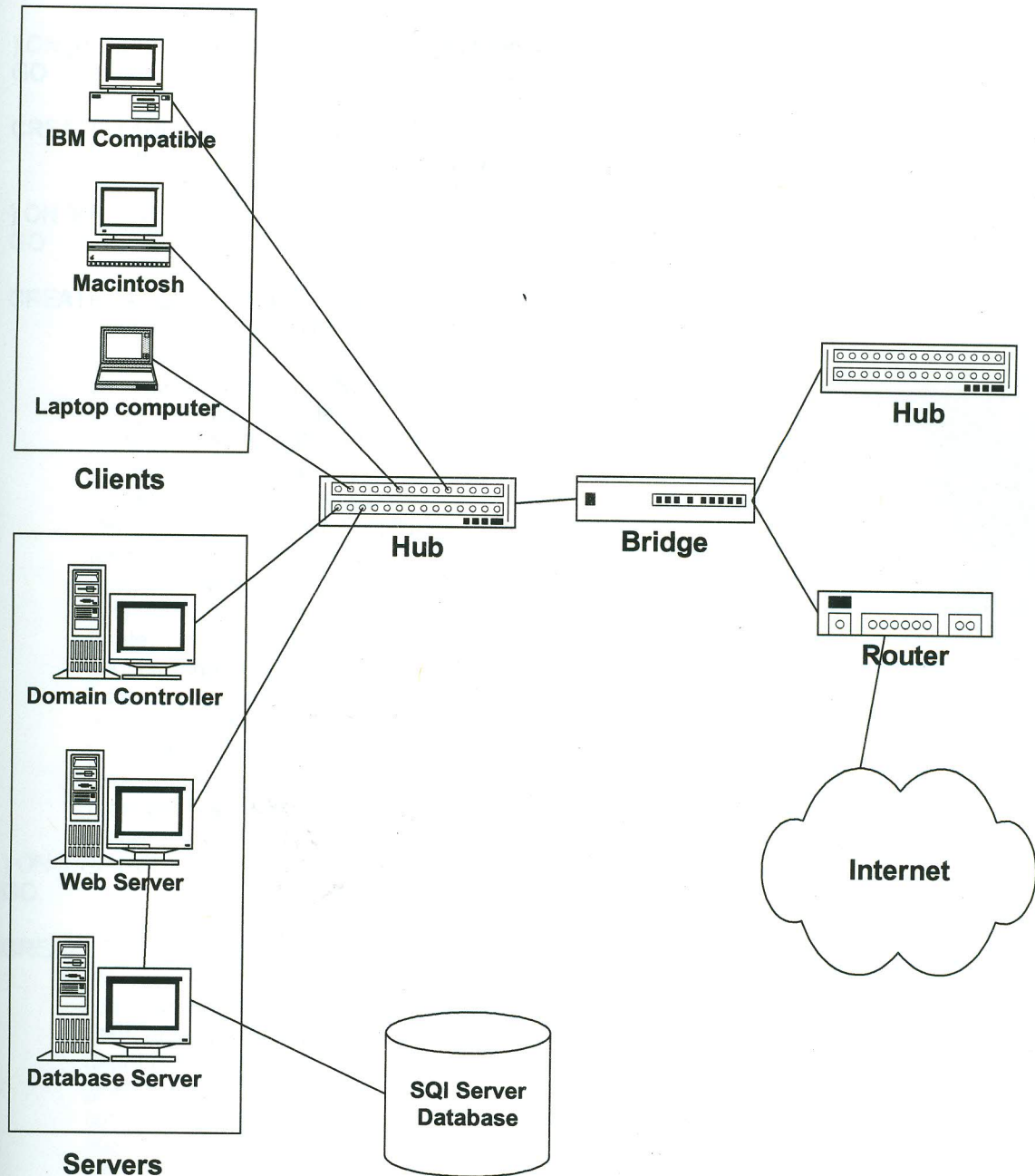
The tools for job tracking greatly diminish the time it takes to currently retrieve project information. The system is a model for future efficiencies that the corporate intranet coupled with Web-based applications can provide. Although the system was developed especially for Hammond Communications Group, the design is modular and can easily be modified to work for any business that employs daily time sheets and wishes to better track jobs.

Appendix A

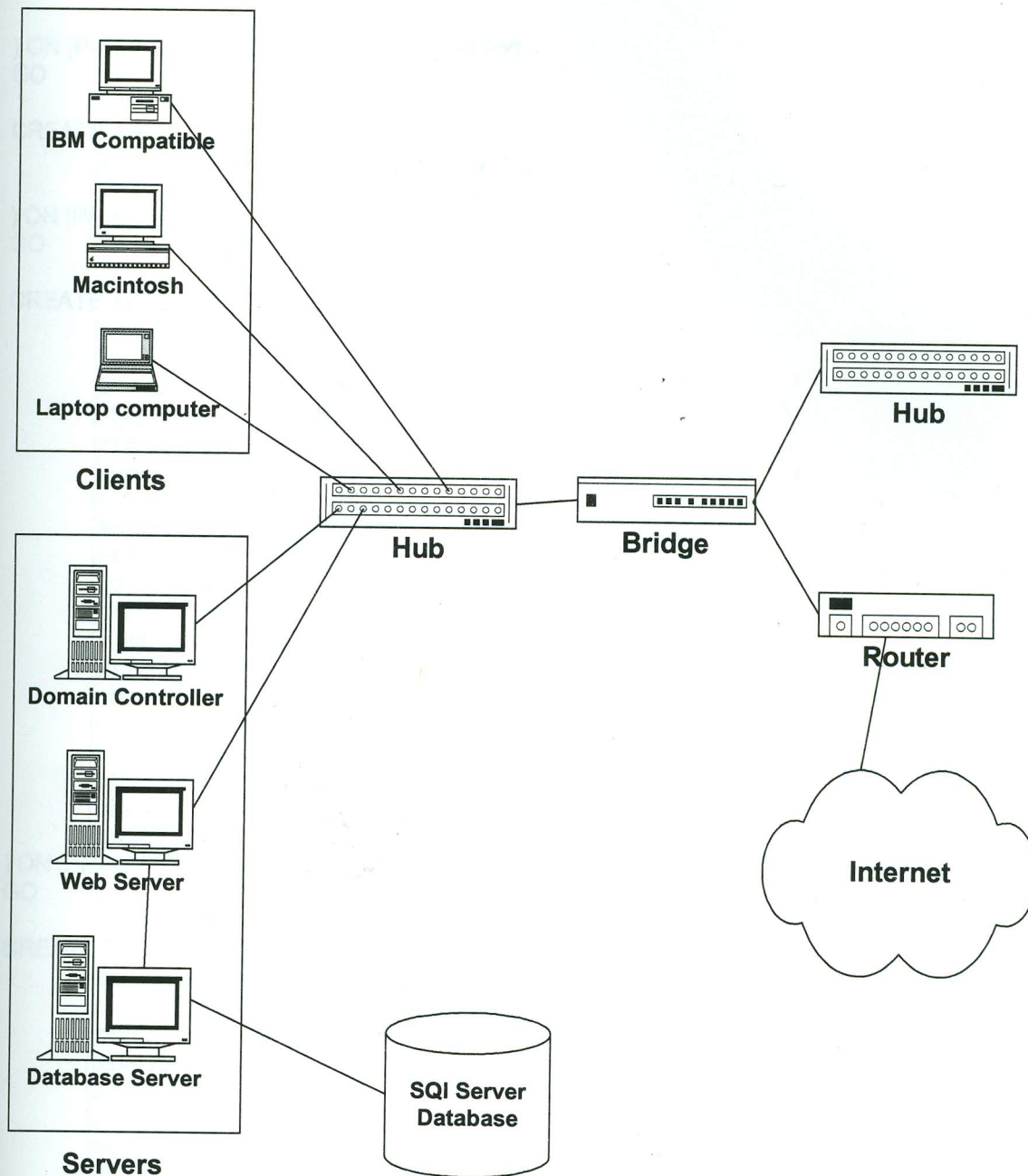
Application Flowchart



Appendix B Connection Drawing



Appendix B Connection Drawing



References

- 1.) Active-X Web site. "Active-X.COM – Your ActiveX Component Library".
[Http://www.active-x.com/](http://www.active-x.com/). No date given.
- 2.) adOpenStatic.com Web site. "Why Use OLEDB?".
[Http://www.adopenstatic.com/faq/whyOLEDB.asp](http://www.adopenstatic.com/faq/whyOLEDB.asp). No date given.
- 3.) Microsoft Developer Network. "Web Workshop – ASP Overview".
[Http://msdn.microsoft.com/workshop/server/asp/ASPover.asp](http://msdn.microsoft.com/workshop/server/asp/ASPover.asp). May 28, 1997
- 4.) Microsoft Web site. "Microsoft Visual Basic 6.0 Enterprise Edition".
[Http://www.microsoft.com/catalog/display.asp?subid=22&site=730](http://www.microsoft.com/catalog/display.asp?subid=22&site=730). November 28, 2000.
- 5.) SQL Server Magazine. "Why is SQL Server better/worse than Oracle?"
[Http://www.sqlmag.com/Articles/Index.cfm?ArticleID=14217](http://www.sqlmag.com/Articles/Index.cfm?ArticleID=14217). July 28, 1999.
- 6.) Suresh Raj, Gopalan. "A Component Engineering Cornucopia".
[Http://www.execpc.com/~gopalan/](http://www.execpc.com/~gopalan/). May 5, 1999.
- 7.) Whatis?com Web site. "Whatis?com".
[Http://WhatIs.techtarget.com/definition/0,289893,sid9_gci213761,00.html](http://WhatIs.techtarget.com/definition/0,289893,sid9_gci213761,00.html).
August 17, 2000.