

# Payroll Made Easy: Developing a Web Based System for Student Employee Payroll

William S. Thieke, Ph.D.

Le Moyne College  
1419 Salt Springs Rd.  
Syracuse, NY 13214  
315-445-4599

thiekews@mail.lemoyne.edu

## ABSTRACT

Administration of a student payroll in technology services, or any other area is a necessary task. It must be done accurately, and it must be on time.

Le Moyne College has developed a payroll system for student staff using a custom developed web interface and Filemaker Pro 5.0. Students log in via a browser, using an ID card reader. The ID number is verified against an employee database and hours are logged based on the time and date when the record was created on the server. Logging out follows similar procedures. The system is also capable of querying the payroll database to see records of hours worked during self-defined time periods.

This paper is intended for all those who are tired of spending hours processing payroll...time that could be better spent doing more productive tasks. It will demonstrate the functionality of the web interface and explore the FileMaker Pro database and CDML (Claris Data Markup Language) behind the web interface.

## Keywords

Payroll, automation, dynamic web pages, database publication to the web, Filemaker Pro.

## 1. INTRODUCTION

The college has approximately 50 students working in 4 labs and several IT offices scattered around the campus. Payroll was processed by sign in sheets at each work location, calculated manually, and transferred to authorized time sheets for submission to the Payroll Office. Typically, payroll submission for the 50 students took about 3.5 hours to complete. It seemed appropriate to develop a more efficient payroll system.

The Payroll Administrator and Information Technology (IT) staff established criteria for the redesigned payroll process. They

determined that the new process should: 1) Be automated, using the lab computers to calculate the time in and out for each shift; 2) Be able to determine the location of the login; 3) Be relatively "tamper proof"; and 4) Automatically generate the required payroll form to submit each pay period.

The changeover began by developing a database for the Information Technology student staff to use to track payroll. Filemaker Pro 5.0 was chosen as the database of choice, as it was the most versatile database already available on our campus and was compatible with both Macs and PCs. For a period of time, data was entered manually into this database from the sign-in sheets by the payroll administrator and used to process and calculate the payroll to a format sheet that was designed to mirror the college's payroll form. This process was time consuming and transcription errors were a common occurrence. Next, IT student staff was given access to the main database and a template was developed. Students entered hours directly into the payroll database. They were permitted to write to the database on the login only; After the original login, the entry automatically changed to read-only, to maintain the integrity of the database. These changes cut the process down to about an hour for each pay period.

There were still several problems with this new process. It depended on an honor system. Students enter the time manually when they sign on to the database. This time may or may not be the time they "actually" logged in. Additionally, we were unable to verify that students were actually in their work location when they logged in. This "honor" login system worked well for the majority of the students, but there were several who abused it. Finally, several students were doing intern work at odd hours on special projects from their rooms. The existing system forced them to login after their actual shift, or to come to one of the labs, sign in and then return to their work site.

## 2. DEVELOPING THE IDEA

In an attempt to resolve some of the problems described above, the IT department decided to move the student payroll system to the internet. This was uniformly accessible across campus and would provide access to the system for interns working for IT from their rooms. We were able to take advantage of Filemaker's built-in web publishing capabilities to make the payroll database "web-accessible". Filemaker uses Claris Dynamic Markup Language (CDML) to enable its databases to interact dynamically with the internet. CDML is based loosely on Tango and Lasso

scripting languages. There are two choices for publishing to the web with Filemaker: Use its built-in interface, which utilizes a generic interface to interact with published databases and enables “easy” publishing of web-enabled databases; or use custom web publishing to develop the payroll program. Because Filemaker’s built-in interface lacked some of the bells and whistles that the college wanted to include and was less secure than the custom solution, the latter was chosen.

The entire process is based on four Filemaker Databases: 1) The Staff database, which includes a complete record on each student, including all necessary personal and payroll information as well as a brief employee history; 2) the Login database, which captures login information as students log in; 3) the On-Duty database, which holds login records of students during their active shifts; and 4) the Payroll database, which stores full login/ log out records for all shifts worked. In addition to the Filemaker databases, HTML code was developed using a combination of Macromedia’s Dreamweaver 3.0 and hand coding. CDML code was developed using the Filemaker’s Online Database Publishing reference guide that comes bundled with Filemaker and the web-based example files that are included in the Filemaker install. The entire process runs on a Pentium II, 350 Megahertz PC running Windows 98, Filemaker Pro V5.0, and the built-in Filemaker web-server. In addition to the payroll databases, this machine hosts an inventory database for Information Technology, and a Student Activities transcript database with a web-front end for the Student Life office.

### 3. THE CURRENT PROCESS

The entire process works as follows. When students arrive at their workstation, they go to a bookmarked web site, (see figure 1) choose their options, and use an ID card reader to read a 17-character code placed on their Student ID card by the campus ID card system. Only six digits of this number are unique to the student. The rest are site codes, card issue numbers and other campus information. Using their ID card resolves several security and authentication issues. When students swipe their card through the scanner, the payroll administrator knows they are at their workstation and using the system correctly. We are also able to get around the problem of students saying that they “lost” their ID card, keeping the “old” ID card at their primary work location and having their peers swipe them in. When students are issued a new card by the ID office, Information Technology is automatically notified and their card-issue number on the staff database is changed, enabling only the new card to authenticate to the system.

When students swipe their card and hit enter, the browser captures the 17 character code, extracts both the student ID number and card issue number and compares this number using a lookup to the Staff database (see Figure 2) that only the Payroll administrator has access to. If the ID number on the processed card matches an entry in the staff database, a new record is generated in the Login database. The record is time and date stamped based on the web server’s time and date (which students cannot modify). A lookup is performed to the staff database,



Figure 1. Information Technology payroll login page.

capturing pertinent login information (i.e. name, payroll ID, Social security number, etc.) associated with the Student ID number. If the ID number is not found, a new page is generated that flashes a warning to the user reminding non-IT students that they don’t belong here, or telling the IT staff that there is a problem with their ID card.

Finally, via CDML scripting, the IP address of the machine where the student is logging in is captured to the login database, enabling verification that students are logging into the payroll system from their assigned work location. A new web page is dynamically generated which processes the student’s name, the time and date that they logged into the system (see figure 3). Here students are also able to indicate if they are subbing for another student, enabling us to track shift changes. Students then click through two more pages (which process and manipulate the record) and return to the main payroll page. Their active record now sits in the “On-Duty” database and has been deleted from the original Login database.

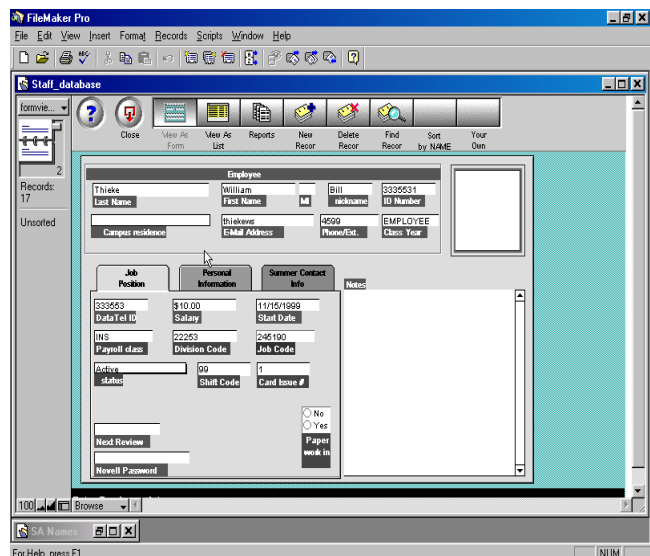


Figure 2. Staff Database Example.

The log out process is similar to logging in, but includes some error trappings to ensure that students are logging in and out properly. When students finish their shifts, they again swipe their ID card through the card reader. As before, the ID authenticates to the Staff database, which will reject any unauthorized ID card numbers. If authenticated, the web page processes a lookup to the On-duty database for all records containing that student's ID number. Once a record is found, a new web page is dynamically generated that flashes the student's name, login time and date, log out time and date and the total time worked on that shift to the nearest quarter hour (this was a requirement of the payroll office,

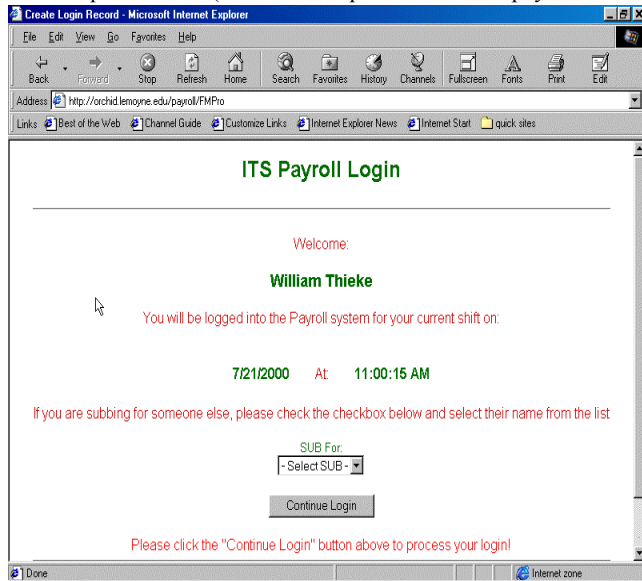


Figure 3. The login confirmation screen.

and required Filemaker to process a complicated formula to generate correctly). In addition, the logout IP address is also captured, verifying location of logout. The logout process then proceeds through two more web pages, which write logout information to the On-duty database, copy the current record to the Payroll database, and then delete the current record from the On-duty database.

One built-in error trapping in the logout process is designed to identify students who fail to log out, or those who log out incorrectly from a previous shift. When students forget to log out, and then log in for another shift at a later time, the login record for the first shift is still present in the On-duty database. Students who fail to log out properly may have two or more login records present in the database. Ideally, there should only be one login record for each logout processed in the on-duty database. By default, Filemaker would choose the first record matching the students ID number. This would result in shifts being processed with the login data from one shift, and logout data from another shift. To prevent this from happening, if Filemaker encounters more than one record in the On-duty database matching a students ID number when logging out, a warning web page is dynamically generated. This page indicates the problem and lists all shifts that exist in the On-duty database (see figures 4 and 5). It prompts the students to select the correct login shift (which corresponds to the current logout record) and processes the logout based on the selected shift. Students are also reminded to logout from the other

shift(s) that still exist in the On-duty database and to contact the payroll manager so these records can be adjusted with the correct logout time. If students fail to notify the payroll manager, these records are usually easy to identify. Standard shifts are 2-3 hours and most students only work one shift a day. When shifts are identified lasting 6-12 hours or that span several days, students are contacted and the entries are corrected manually. Manual entry by the payroll administrator defeats the accuracy of the login/out system, so persistent offenders are reminded to use the system correctly.

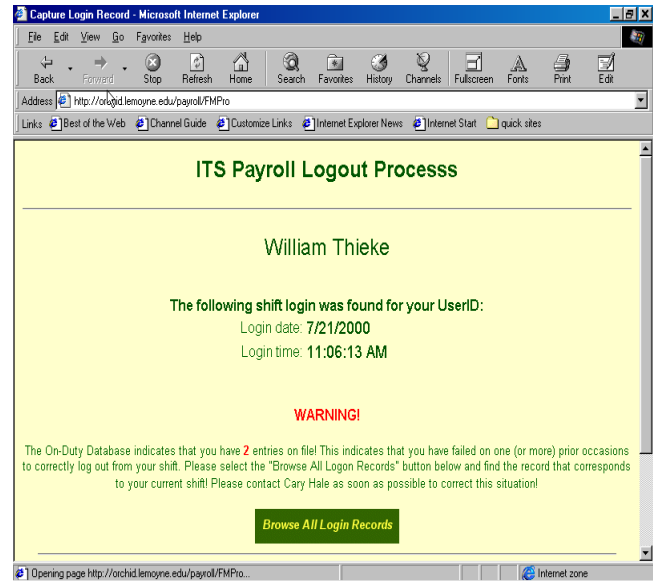


Figure 4. Multiple shift login warning screen.

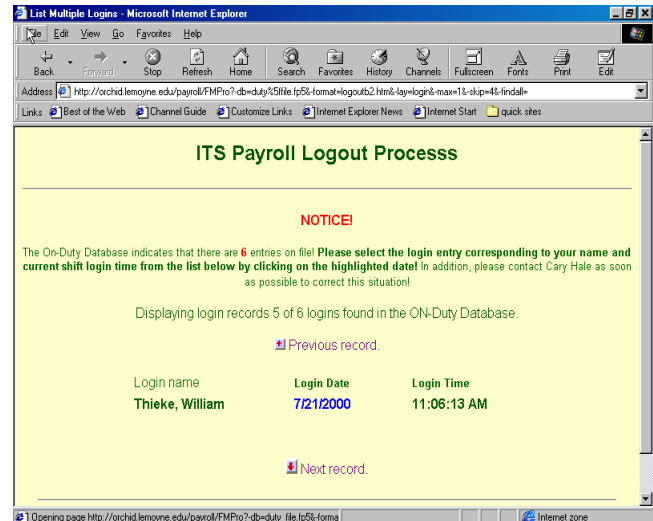


Figure 5. Multiple Login shift selection screen.

The Payroll report is generated bi-weekly, by extracting records from the payroll database that fall between the date ranges of the current pay period. These records are imported into a smaller database that sorts all data into a format that mirrors the Payroll

Office's preferred form. The form lists all student shifts for the current pay period, the total hours, the total pay for the pay period, as well as pertinent identification information on the students. This information is printed out and submitted to payroll. These bi-weekly electronic payroll reports are archived for reference on a secure server. In addition, the master payroll database is archived each semester and after the summer session. The archive process keeps the database at a manageable size and improves access times for students logging in to the database. These archives are also stored on a secure server and the data on the master payroll database represented by the archive is purged.

#### 4. STUDENT LOOK UPS

Another issue that we wanted to avoid were students who would constantly ask what hours they had worked during some previous pay period or those verifying that their hours for the current pay period were correct. This required going back into the payroll records and detailing them for students. To get around the issue, a look-up routine was developed as a third option for the students. As with login and logout, it requires students to authenticate to the Staff database prior to accessing payroll records. Once authenticated, students are prompted to input "from" and "to" date ranges based on the times (or payroll) periods they would like to view. When this information is submitted to the database, all shift records present in the Payroll database corresponding to the Student ID and falling between the dates specified are found. A new web page is dynamically generated that lists all the corresponding shifts, 10 records a page, detailing date, time in, time out, and hours for each shift (see figure 6). When the students finish viewing the information they need, they are given the option of performing another search or returning to the main login page.

### 5. HOW WE HAVE FARED

#### 5.1 Problems solved

The new web-based payroll system resolves most of the issues that were encountered prior to the development of the system. First, it enables the payroll administrator to verify students are at their workstations at the start and end of their shifts. Second, it provides accurate pay records based on computer-generated times. It also eliminates transcription errors and errors generated as a result of illegible handwriting on payroll sign in sheets. Third, it enables the payroll administrator to identify students who are chronically late or those that leave early for their shifts. This takes the burden off their peers for reporting students abusing the system. Fourth, the system is relatively tamper proof. Students do not have direct access to any of the payroll or staff databases. They can view them using Filemaker, but they are read-only to all except the payroll administrator. The Staff database is password protected, as it contains personal information about the students. The payroll database can only be accessed and written to by students using the web interface. This provides a "tamper-proof" interface for the payroll process. Finally, the payroll process now involves virtually no manual processing by IT staff. The entire process at the end of the pay period now takes about 15 minutes (most of that is verifying that students showed up for shifts that they were supposed to work, not entering data and processing it).

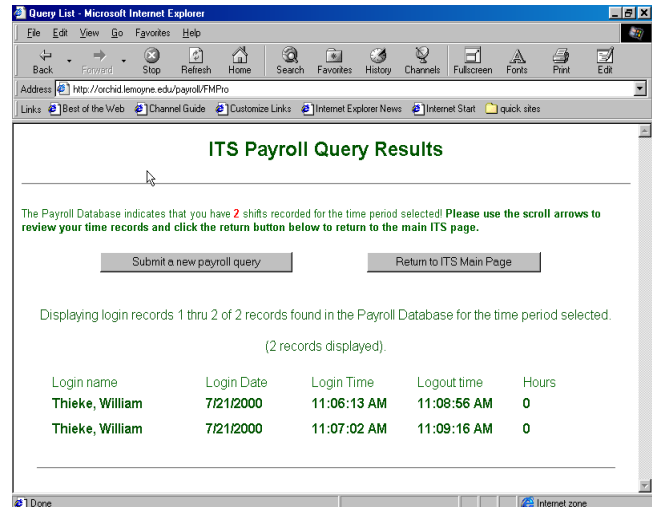


Figure 6. Payroll shift query results screen.

### 5.2 Lessons Learned

There is some maintenance involved in keeping the program running smoothly. Even though the process is almost fully automated, it requires manual entry by the payroll administrator on a regular basis. Students are constantly losing their ID cards, and we continually hire and lose students, requiring the staff database to be updated and maintained. Also, students change financial status (i.e. College work-study to fully funded or vice versa) over the course of the year requiring that payroll reflect these changes. Students neglecting to login at all will also require manual entry into the payroll database. Finally, as with virtually all computers, the machine hosting the payroll program periodically experiences problems requiring downing the server, restarting Filemaker and re-opening all of the active databases (Note: we have automated this process somewhat by having a master database that when opened, auto-runs a script that opens all of the master databases). This has only happened one time since the program became fully operational in November 1999.

One of the biggest problems encountered during this project was trying to get the CDML tags to function as described in the Filemaker documentation. Most of this was accomplished using the Filemaker help files and example programs found within Filemaker. The code from these example files was customized to fit our situation. Even though Filemaker touts itself as being able to easily publish databases to the web in minutes using the built-in interfaces, the custom interfaces are significantly more complex and not terribly intuitive. Many, many hours were spent reading and re-reading documentation and going over the HTML code in the example files trying to get the look-ups to function correctly and data to post from existing databases. Once we were able to figure out the syntax and how the HTML and databases worked together, the process got significantly easier.

### 6. FUTURE PLANS

In addition to making minor adjustments to the program to allow it to function more smoothly, one major project in the works is to fully automate the process so that we are able to submit the payroll electronically to the Payroll Office on payroll days. Our

staff has been working cooperatively with the Student Payroll Manager to detail the necessary submission information and format required for our data to be exported into the main campus payroll system. The payroll office has even indicated that they might like to see a fully integrated system like ours developed for some of the other larger student-employing departments on campus.

## **7. CONCLUSIONS**

The Web-based payroll system developed at Le Moyne has been demonstrated to have several advantages over the traditional “pen and paper” methods of processing payroll used by many institutions: 1) It provides an accurate accounting of student hours. Login and out times are computer generated, not left up to

the students to honestly log their hours; 2) Processing payroll is a snap... about 5 minutes from start to finish. Pay period hours are exported to a “report database” with a layout mimicking the College’s payroll form. Payroll totals are automatically calculated and account information on the students pulled in directly from an employee database; 3) It enables us to verify that students are actually at work where they say they are. The login/ logout sequence captures the IP address of the local machine, enabling verification of login location. By computerizing the payroll process, Le Moyne has freed up valuable staff time, previously spent processing payroll, and provided the college with an accurate and efficient method to record student hours worked for the IT department.