

# **eModes: Enhanced Music Organization Digital Ensemble System**

**Tony Wilsman**  
**Information Science and Technology Department**  
**Doane College**  
[awilsman@doane.edu](mailto:awilsman@doane.edu)

**Tim McArdle**  
**Information Science and Technology Department**  
**Doane College**  
[tmcardle@doane.edu](mailto:tmcardle@doane.edu)

## **Abstract**

The Enhanced Music Organization Digital Ensemble System (eModes) is targeted for use by a post-secondary educational institution in a musical performance setting. This is not however, the limit of its potential application. The system is intended to increase the efficiency of a music program by simplifying and streamlining the administration, rehearsal, and performance of multiple and diverse ensembles. The system digitizes the process of maintaining a music library, simplifies the administration of ensembles, and automates the tasks of assigning and displaying music.

eModes is built on existing web and database technologies. It utilizes Apache Web Server, MySQL, and the PHP scripting language, running on the Linux operating system. Music, performer profiles, and director profiles are stored in the database, and are retrieved using PHP scripted web pages.

## **Introduction**

The original concept of eModes was designed for a senior seminar project for the Doane College Information Science and Technology Department. This concept is to turn the entire music ensemble system into a digital networked system. The music industry is in full swing with the ever growing technological advances. The teaching and playing of music, however, is not up to date with today's latest technologies. Everywhere from elementary school music rooms through collegiate ensembles to professional music ensembles, metal or plastic stands and paper music are still used.

eModes will revolutionize the way that ensembles are taught, prepare, and play music. This version of eModes is specifically engineered with collegiate ensemble groups in mind. Directors will no longer need to rely on their students to bring their folders to class. Everything about the student, from what instrument they play and what voice part they sing to the high school they attended and their home address, will be stored in a database on a single server.

The student simply goes to practice and logs into their stand. All of the music which would normally be in their folder is there for their viewing. Any announcements for them from their director can be seen, including a schedule for that day's rehearsal time. Concerts will no longer have to wait in between songs for the players to pull out a different piece of music, for with only one click of a button the director can automatically send the next song to the student's screen.

With the growth of technology, the music industry is very adamant about its views on music sharing right now. This is particularly true for .mp3 pop culture music files, but is also very pertinent in the eModes system. Because of this fact, we wanted to make a system where only one computer would actually hold any of the digital music. The director's stand and the student's stand would not physically hold the music, but would be constantly downloading the music from the server to avoid piracy. Ideally, only the music meant for the particular institution could be used on their hardware.

Each institution or professional ensemble would have its own secure server tailored for the ensemble's needs. This server will hold digital copies of the music, along with the scores. This version of eModes is specific for a collegiate or university setting. There will also be an administrative side to the system for keeping track of users, music, ensembles, and parts. A database will be used to keep track of all of the information needed for the entire system.

### **eModes: Enhanced Music Organization Digital Ensemble System**

Ideally, eModes would use much more advanced hardware than we were able to attain. We started with an initial hardware concept of three machines. One of these machines would be labeled the "Server". One would be labeled "Director" and the last would be

labeled “Student”. Each of these machines would hold the information pertaining to who would be using the respective machine.

Initially, we viewed these machines as all running Linux. By modifying the operating systems on the machines, we could tailor them to our personal use. The server machine would have Apache Web Server installed on it as well as MySQL for the database and PHP for web-connectivity to the database.

### ***Hardware Implementation and Evolution***

We used three computers for the first phase of our hardware evolution. The Server computer was a Sun SPARC. The Director and Student computers were two IBM PCs. Both had Pentium II 233Mhz processors in them. Each had 64MB of RAM and a Cirrus Logic 6D-546x VGA 2MB video card. Each computer was also equipped with a 4GB hard drive.

The second phase of hardware evolution was very similar to the first phase. We decided we no longer needed the Sun SPARC machine and phased it out of production. MySQL and PHP were not compiling properly, but we could successfully install them on a different computer. The IBM named Director became the new server for the system and was also used as a workstation. The IBM named Student was still used as a workstation which then connected to the Director server.

For the third and final phase of hardware evolution, the Student computer was also phased out. By using software on machines at other parts of the campus, we could use FTP for file transfer and design the site from more up to date computers. We left the server because of its permanent IP address. The Director was used as a server still and personal computers were used for workstations.

### ***Software Implementation and Evolution***

During the design of eModes, the server and workstation computers went through several software changes. The protocols used to transfer information between computers were also changed.

During the first hardware phase of three machines, the server and workstation computers were joined together on a closed network in a small workroom. Network File System, or NFS, was used for file sharing between the server and workstation computers during this phase. The Sun SPARC, running as the server at this point in time, was running Mandrake Linux 7.0 for SPARC as its operating system. Also installed on the Server was Apache Web Server x.x for internet connectivity and MySQL (My Structured Query Language) for a database system. PHP (PHP: Hypertext Preprocessor) was not used during this phase of software.

The Director and Student IBM computers were also using Mandrake Linux for an operating system. The version used on the IBM computers was Mandrake Linux 9.0 for x86 architecture. The software which was used for text editing was Bluefish x.x and VI.

For the second phase of software, the IBM computers were still using NFS for file transfer and were still on a closed network. The Director computer was updated with Apache Web Server 1.23 and MySQL 3.23 for x86. PHP 4.1 was installed on the Director as well. The software used to establish communication between Apache, MySQL, and PHP was EZPublish. The Student still had the same credentials as the previous software phase.

With the removal of the Student computer, the structure of the system was changed for the third phase. The Director computer was connected to the Doane College network and a permanent IP address was received. The name was no longer Director and was changed to Lintest.doane.edu. An FTP Server was installed on the computer for intranet and internet access for editing.

### **eModes Usage Scenarios**

The following are scenarios which describe the functionality we would like to see in the eModes prototype.

#### ***Server***

If a higher level institution were to implement eModes, the first place the director would start would be the server. The server has standard administration functions which would allow the director to enter information about their ensemble. The director logs onto the server with a set password and username. Immediately, they create an account for themselves with a username and password, which will allow them access to the server. They are now known by eModes as a "Director," and their title, first and last names, username, and password are now stored in the *director* table in the database.

After a director has successfully created an account, they can begin adding ensembles to the system. The ensemble name and whether it is an instrumental or vocal ensemble are added to the database in the *ensemble* table. Also, during this process, the director can assign specific directors with specific ensembles. This information is stored in the *directorProfile* table. Next, the director would add student records to eModes. Student information, such as first and last names, high school, high school director, home and college addresses, are then stored in the *user* table.

The prototype of eModes comes with a standard set of instrumental and vocal parts entered into the *parts* table. An example of a part would be Alto, Soprano, Tenor, or Bass for vocal and Trumpet, Clarinet, Flute, etc. for instrumental. However, the director has the opportunity to add parts to the database.

Now that the director has entered ensembles which they are the director of, along with students and the necessary parts in the database, they can start assigning students to ensembles. The student, ensemble, part, and chair are all then placed in the *profile* table. Now, the director can look at the student's profile and see which ensembles they are entered in, as well as all information in the system about the student.

At any point in time, the director can also edit any information about the student or themselves. They can add or remove students from ensembles and add or remove parts from the system. Directors are given the option of editing the name of ensembles, if it were to change.

The director now has students, ensembles, and a full list of parts in the database. The director is now able to upload music to eModes. They do this by adding a piece to the database, which is stored in the *library* table. Now that they have a piece in the system, they can add their music for that piece, which is stored in the *sheet* table. Because of the fact that institutions normally have large amounts of music in their libraries, eModes comes equipped with an Active Repertoire for each ensemble. The active repertoire is very similar to the music which a student would keep in a music folder. The director can add and delete music to the ensemble's active repertoire at any time from the director stand. This information is stored in the *activerep* table.

### ***Student***

When a student is at an institution using eModes, getting into the system is very easy. The student simply goes to the student page. They pick which ensemble they are in, and then they are given a list of people in that ensemble. All they do is pick their name from the menu, and log into the system.

Now that the student is in the system, they can see the music in their active repertoire. Because their director has already entered which part they play for each ensemble, they simply click on a song and their part is automatically displayed.

### ***Director***

When it is time for practice, the director goes to the director page. They choose which director they are, and which ensemble they want to log into. Once they are in the main page, they can click on "Active Repertoire" to see that ensemble's active repertoire or click on a song name to have the score displayed.

### **Conclusion**

Overall, the project was a success. The prototype of the system works, but has a few bugs in it. The current method of displaying the music is simply displaying a JPEG image in a web browser. The system is only effective with music which is only on one sheet. A better system for turning pages would need to be implemented. As the system

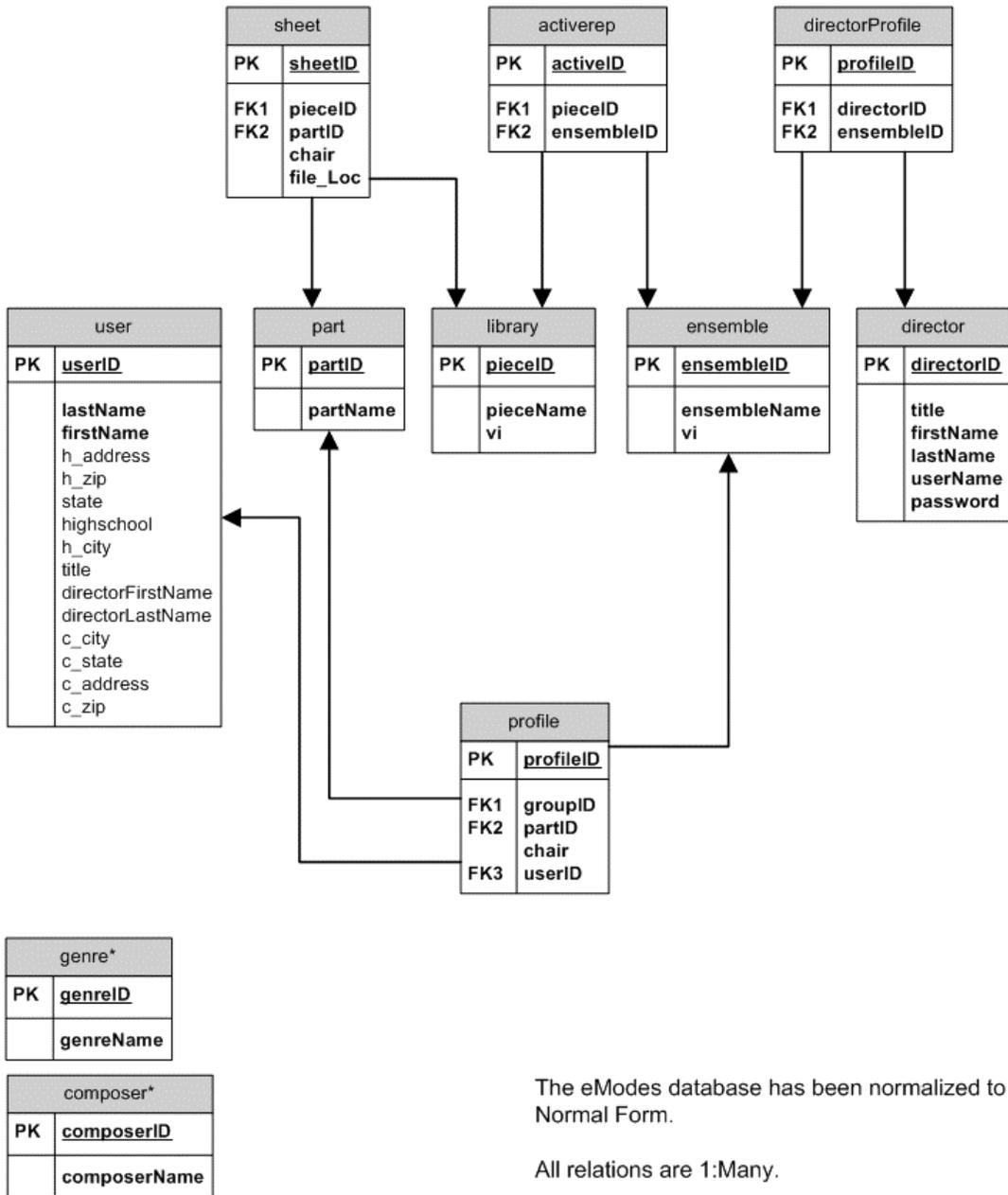
stands right now, after adding a piece of music, each sheet must be uploaded on the server manually. A system with a set naming convention implemented would allow all sheets to be uploaded at the same time, with all parts assigned automatically.

Before beginning this project, neither person had ever programmed in PHP or used MySQL before. We learned about the Linux operating system and how to set up Apache Web Servers, MySQL, PHP, along with compiling software from source.

After getting to this point on the prototype, we have learned many key things about this project. First off, we learned the importance of software engineering concepts and could have reused code more efficiently. We could have more extensively researched other development environments like VB.net or other high level programming languages, such as Java with its built-in applets.

## Appendix A – Database Design

### eModes Database Design



The eModes database has been normalized to Third Normal Form.

All relations are 1:Many.

\*These tables are part of the database. However, they are not currently being utilized.

Figure 1: eModes database design.

## Appendix B – Site Map

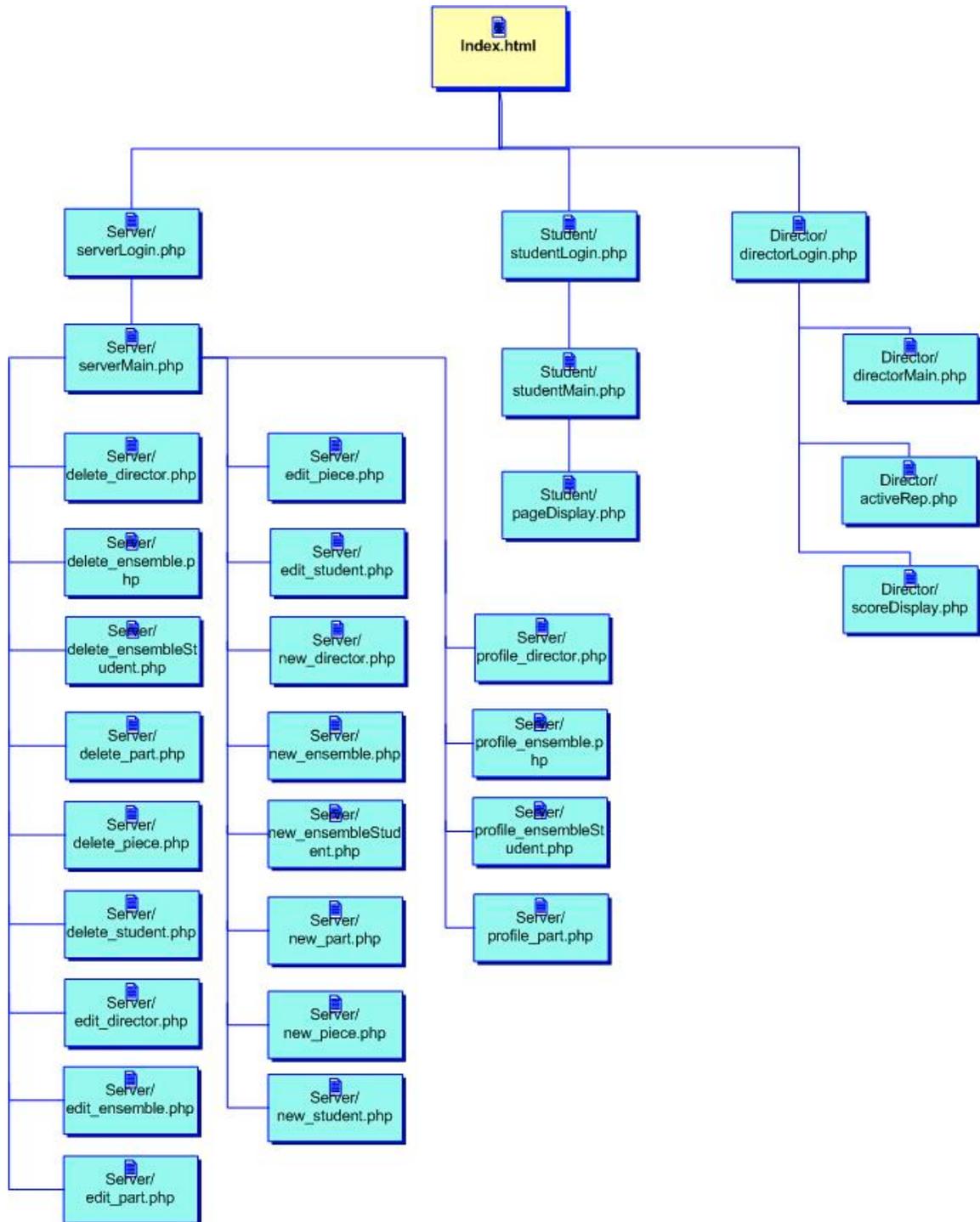


Figure 2: eModes Site Map.