

MAPS 3 month summary (January-March)

As I continued to work with the initialization file stuff, I really got to like it. It makes the application feel more like a real application. It will also allow more flexibility for the user in the future to customize the application as they please, perhaps hiding features they don't use (although I don't think there are really any unnecessary parts). Having the initialization file contain path information solved some of our problems of opening the script in the right directory. It does, however, cause some problems when the program is transferred to a different machine along with the INI file. This will have to be dealt with in the future.

The redesign of the layout of the form turned out to be a big task, but well worth it as it looks and feels much better. I am still really pleased with the new button control that we bought as it has helped enormously in contributing to a smooth look for MAPS.

January was spent continuing to add and fix things that we discovered in our meeting with Anja. She had many good suggestions and it will be some time until they are all implemented. One thing in particular that she suggested was a great fix for the last thing that I felt was still 'clunky' on our design. The Insert checkbox really didn't fit in with the intuitive design/use of our program. She suggested having a line appear between the boxes in the row of five nodes on the bottom where you could then insert a blank node to fill. Although I was unable to create it so that the line would only appear when the user moused over, (which I think turned out to be better after all), there are lines in between the boxes that when clicked will insert a blank node. I think this is much more intuitive for the design, and finally I could be rid of the checkbox. (The checkbox also stunk because the user had to *uncheck* the box in order to not insert another node there. Also, the user would have to check the box, and then select a picture/sound before the node was inserted, and this was very confusing. The new design is much better!).

Anja also made some good suggestions for improving the wording on lots of our buttons and menu items. For example, we changed our reference to viewing the power-point view to 'Whole Script View' which describes what it does much better. Anja also managed to crash our program a few times which helped weed out some database and other bugs.

Some of those other bugs included things where we needed an error message to pop up. This prompted me to create our own error form, which initially I made specifically for one type of error and then generalized into a function so that I could use it for other error messages. Eventually, however, we got rid of our own error box function in favor for Visual Basic's message box function because we wanted to record the response people made (Yes/No) in the calling function rather than just having a Yes/No function. Originally when I tried to have my own Yes/No function, they were not general enough for the general error box function, and it would have been too long a list of parameters. Plus, there was no easy way for the calling function to find out what had happened (what button the user had chosen). I started going the route of having another global variable, but since my error box was just its own form, it wasn't modal so the calling function would have been allowed to continue without the user having made a choice in the error box pop-up. So the lesson learned here it to not reinvent the wheel. Search for a currently existing tool before trying to create your own, even if you think it will be better. The only real drawback to using the Visual Basic MsgBox is that we can't

have it match the main form in color, look and feel. I realized that this wasn't a big deal, however, because most users of windows are used to the message boxes look, and perhaps our own form would have been less familiar.

Other suggestions that Anja made were to make the form change in noticeable ways according to mode to make it clearer to the user (when it is important for the user to know, that is) when the mode has changed. For example, when in edit mode, the words on the buttons change, but this is not enough of a signal for the regular user to really notice, and if they still think they are in append mode, then they could waste time doing the wrong thing. As of now I still haven't altered it to change color when it changes mode (partly because there was more important functionality updating that needed to be done, and partly because I hadn't found a complimenting color scheme that I wanted the form to change to).

February was spent continuing to fix bugs and also to implement new functionality in the program. I had a friend come in and thrash at the program to see what bugs he could find. He found several. He also made some good suggestions of other functionality or improvements to make. I have pages of lists of things, many of them implemented, many of them left to be added. It's amazing how an application just grows and grows. We're always adding new things. But I think we are doing it right because adding things is not a difficult task, even though our code has become quite extensive (not sure how many lines, but a couple thousand is my guess).

Now to cover some of the new functionality and other cool stuff that I added. The menu items have now greatly increased. The properties/preferences box now contains not only boxes to set up the Picture and Prompt paths that the form will open up in, but also an option on how fast the script will play when the run script button is pressed (for demoing it in the application only). Right now it is just set to three speeds, but eventually we will have a slider bar so there are more choices of speed. I also plan to add something that shows how fast a particular speed it, because right now it is just kind of a guess (for the user). Another option the user has in the preferences is to choose whether double clicking on a sound adds it to the script or plays the sound. The default is to play the sound.

Taking note of some of Anja's earlier suggestions, I've made full use of the message boxes as feedback for performing certain tasks, such as loading the script to the device, saving the script, etc. In fact, there were some database problems that came to light after adding these notification boxes. Apparently we were not properly saving scripts to the database, and when a script was being loaded sometimes the nodes would all be out of order. This was easily fixed by adding another SQL statement to the insert call.

A very exciting addition was the Picture Select Display. Anja had suggested that the user be able to preview thumbnails of the pictures in the list they were choosing to add to the script. So I created a display of the pictures that are available to add to the script and put this display over the area where the user previews their script (the Run Script area). This was a nice way to make the best use of space that during the majority of script creation was going unused. There is a toggle button so the user can toggle back to the Run Script view and run their script. Since these events are mutually exclusive, the user will never need to see both areas at the same time. User (friend) testing that followed the addition of this got great success, they really liked it. In fact, the list of

picture names is almost unnecessary now. Perhaps in the future we will integrate the picture names near the boxes.

Another thing that was improved (and still needs to be made better for some buttons perhaps) was to change the icons on some of the buttons. In the case of running the script, I changed it to words, which made more sense than just a triangle resembling 'play' on a VCR.

Stefan had me add a comment block to each function that included the date the function was last edited, where that function was called from, what the function does, and any other info/tricky parts in the function. This is very useful in making our code robust so we know when we changed things. It is also a good habit to get into for writing code especially when other people have to look at the code. I also made sure that all of the variable names in functions had useful names rather than ones that didn't explain anything. Despite some variable names being rather long, I still think that is better than having a bunch of hard to understand variable names. It is amazing what a difference clear variable names can make in improving code.

A lot of time was spent thrashing at the application to find bugs and fix it up so that when we demoed the program, it wouldn't embarrassly crash. By the time I presented our project to a group of high school girls on an engineering preview day, the program was crash-proof.

In the debugging, crash-proofing process we came across a need to add some stuff to the database. The database now includes another table called CurrentScript that holds the script currently to be loaded to the device. This enabled us to upload less overall to the device (rather than loading all of the unused scripts also). This turned out to be a fairly simple and straightforward process, demonstrating the power of Sybase.

Coming up in the next couple of months (maybe?) we're going to add some more cool functions such as a pause button for the run script action. Another thing that will end up being pretty easy to implement will be a delete script function. Very soon I will add a box to the script so the caretaker can input the amount of time each step in the script should be allotted to take before it causes an error to occur.

Most excitingly, we will begin user testing very soon. Thanks to Anja's connections, we will be able to test our GUI out on real caretakers and also test out MAPS with real users.

In order to accomplish consistency in how we test subjects, I am going to be creating instructional videos on how to use each of the systems. For the general application (the GUI) I will also be making help videos that show how to do everything on the form. These are all thanks to camtasia, a really groovy program that records screen action and allows you to play it back. Then I just record sounds, add them to it, and viola!

Another exciting thing was for MAPS to be accepted to attend the RESNA conference in June. I will be attending and presenting along with Stefan and Jim Sullivan.