Bret Fontecchio
May 15, 2013

Final Project Paper CS12

Goals and Purpose:

I created a website called PlatonicAether which is designed to help people, especially younger students, learn math.

The goal was to create attractive visualizations and and interactive features to help make the concepts clear. It is not intended to replace classroom learning, but it is intended to be a supplement, which perhaps teachers would link to. I indicated in the proposal that, time permitting, I would like to create interactive questions and keep a scoring system in the database. I did not however have time to do this because I ran into a great deal of trouble getting the sql database working properly on my mac, and I believe that my current work has been designed in a scalable, extensible manner which will permit easy migration and additional work on another server.

Implementation:

The site is implemented using a Model-View-Controller (MVC) layout. The index (index.php) can receive queries for content in the q field. e.g. index.php?q=about will lead to the about page. The about page is a default (home) page, and it comes up if a query passed to the index is not found. It contains an accordion box implemented using jQueryUI, which says a bit about the purpose of the site and what inspired it.

The authentication features, including the login, logout, and account creation pages were implemented in php. I used my own code from CS75 as a template. When I was taking CS75 I was able to write php but I unfortunately did not know html or css, so It was good to see these two areas come together. The trouble I experienced with the database, however, was extensive. When I finally revoked all privileges for root by accident while trying to fix these problems, I realized that MAMP (Mac Apache MySQL PHP) is not the easiest to configure server environment. In the future I will use it in a more limited manner, and probably stick to Linux like we did in CS75. I believe the database functionality would work for someone who has the settings appropritate on their system though. Appropriate settings must match the constants declared in models/db.php and controllers/register.php.

The Unit Circle came out very well in my opinion and I believe it is production quality. I will allow students to easily grasp the relationships between angles and trig functions in a colorful and interactive manner. This part of the site was implemented in Processing, which is a dialect of javascript. It can also be compiled to Java, although the functions are slightly different. I wrote the code for this part from scratch. The biggest trouble I had was reconciling the Cartesian coordinate system I'm used to with the pixel based system in Processing. A higher y value goes lower on the screen. It really took a while to get comfortable with. In the future I would perhaps start with code that creates a Cartesian plane and then take it from there.

The trig functions part of the site (index?q=trig_functions) was supposed to be fairly

straight-forward, but ended up having some caveats. The mathML language is not very well supported, so I ended up using mathJAX to work with it. I found mathML somewhat poorly documented, and some of the code that looks simple took a while to put together due to the dearth of tutorials and examples online.

Use of bootstrap throughout the site went flawlessly. It's definitely a library I enjoyed using. I decided to use the flowing container because it places a the nav bar very neatly and I like the smooth appearances.

Future Directions:

I would like to add interactive questions to the site. These questions would be in forms bound by the jQuery Form plugin, and the answers would be sent to the server by AJAX and identified by an id, without leaving the page. The server would send a correct, incorrect, or error response. The server would keep score just in case it's graded by a teacher and someone tries to modify the js. The score would be updated and dynamically loaded to the browser.

The questions would be given random parameters by php each time. The scores would be kept in the database, and displayed by the my_account controller.

I would also like to add tutorials to accompany the content, and try to tie it all together with visuals, interactive features, questions, and text.

I would like to add other areas of math, science, computing, etc. to the scope of the tutorials.

Reflections:

I learned a lot during CS12 and I think my understanding of the web is much higher now. It now makes sense to me how these CSS rules work, why the content is separated from the code, etc. I now know much more about html and I think my ability to code it is now not only more rapid but more deliberate. It is amazing to get away from the trial and error approach which really is just painful. I no longer feel like I'm flying in the dark because I can now look through the manuals and documentation as a native html and css coder, instead of someone trying to conjure it up like black magic.

In creating the project, I learned a fairly hard lesson in scope creep by fiddling with the database before creating the actual content. I definitely should have had a working prototype.

I learned a lot about these different libraries and I learned not to reinvent the wheel for things that are already implemented. I'm glad, for example, that I used mathML instead of trying to create those things myself. Now it is easy to update and extend.