|  |
| --- |
| Lesson Learned |
| MyNewMedia |
| Zachary Merrill, Stephanie Koesters, and John Mark De Los Reyes |

**John**

I’ve learned a lot in the project when it comes to learning how to code in Pyhton, how a database works, the Django framework, and eclipse. Before the project, I had never programmed in Python, used the Django framework, or used eclipse as a text editor. I did learn how to work with a database at my work, but not as much as what I have learned doing the project. So programming in Python was a bit strange for me because of the syntax, but after a while, I started to get used to it. When I started on the project, the only framework that I used was the .NET framework, so learning how Django worked was interesting because it made it easy for people to develop a website fast. Since I never used Eclipse as a text editor before, it was an interesting experience because it would tell me if I had an error in a code and etc., but I didn’t like eclipse’s interface and the lack of background themes.

The classes that helped me while I was working on the project was Introduction to Web Development (CIST 1300) and Programming on the Internet (CSCI 2850) because it helped me out a lot when it came to me understanding more about HTML, CSS, and CGI programming. Even though I never thought I would ever use those classes again, those two classes helped me get a good start on the project and on how to make a decent looking website.

If I could do it all over again, and since we only had three months to finish up the project, I would have learned more about Python because Django is a Python web framework. I would have also looked into other frameworks to use because it took the team time to find a framework that is free and provides fast development. Since looking online didn’t help me much when it came to programming in Python, I would have loved to find a good book to help me program in Python because I find it easier reading code from a book than reading it from a computer. Also, programming tutorial books usually have comments on what is happening in the code, and when I was looking online on how to code specific things, about 99% of the code that I looked at didn’t have any comments in them. I would also have learned more about how to interact with a database because I lacked in knowledge about it.

Since I’m not going to be working on the website after this class, I did enjoy the project because I learned more about web development and a new computer language called Python. If I have the time this summer, I hope I can learn more about Python because programming in Python was a fun experience. Not only was it a fun experience, I learned a new programming language that I can probably use in the future. I’m also hoping that I can learn more about databases after this class because using a database and interacting with it was a bit of a challenge for me while I was programming on the website. With that being said, since I had more experience working with a team, more knowledge about Python, and more knowledge about databases, I hope what I learned can be taken to my next internship this summer and/or this fall.

**Stephanie**

I think the project would've been more feasible with a concrete plan from the beginning and quicker milestones to present to the client. Most of the time vanished when we tried to use things we never previously learned (such as Python, XML parsing, and the Django applications with only fractions of documentation). I've learned through my actual job that Python can be useful, and delving into it slightly with this project was a good experience, although I wish I learned more than the bare minimum required to work with Django objects.

It seems that unless Django is used in other places, I didn't learn much syntax-wise, admittedly because I didn't touch many of the elements of code besides the tags and recommendations. I did, however, like using Eclipse for the first time, and used it for other school projects. If anything, perhaps I learned ways to keep track of monitoring a project's progress, and I see better ways to improve my priorities in future projects.

**Zach**

For me, the biggest lesson learned from this project would be to spend more time considering the scope of the project before the actual coding begins. Our clients expectations required much more time than we had in the semester, and we had to cut back on what was asked, simply because there was no way to fit it into the project. The underlying algorithm was not terribly difficult, but because none of the team had experience with web development, I feel we spent much more time trying to figure out how to implement it than was necessary.

I also discovered that tend to act first and think about the consequences later. This leads to me taking on too much of the code by myself just to figure out if it will work. I need to allow my group members more responsibility when it comes to actually building the project.