Narrative Ryan Cook

This was definitely not what I expected from a programming class. My high school CS class involved a strict set of guidelines that we would follow for each project, and allowed for practically no deviation from the one way to do something. Naturally this led me to believe that programming was way lame; this idea soon left my head after a few weeks in your class.

As generic as it is to say, I learned a lot in this class. Despite my daily usage of a computer, I never understood the basic driving forces behind them. After learning about logic gates and the theory behind binary I began to understand how little green pieces of silicone and wiring could add up and provide such a multitude of functions. The visual basic programming language took a little while to pickup on, but eventually things clicked and our group was making small programs left and right. One of my proudest moments in this class was the creation of our "Moody Messenger" for Lab 6. It was a culmination of everything I had learned in the class up to that point and the end product was a working and, although fairly simple, useful program. It showed me that I had the potential to actually create something concrete through programming and was the first time that I felt that I "got it." Knowledge based systems was an interesting deviation from the regular programming we were doing previously and I understood the importance of such a method. Our "Awesome Guitar" KBS illustrated elements of planning using branching decision trees and demonstrated how people with specific knowledge can create a program that helps those that seek expert opinions.

Our semester long project seemed that it would be something that everyone could use while still being within the scope of our class, workload wise. We were right about only one of these things. After the initial design and conceptualizing face of the project we ran into roadblock after roadblock concerning development. At first we didn't know how to create a database, and then link a database to the program, and then fill the database with user-submitted information, etc. The internet proved to be a double edged sword, with people offering snippets of code but not providing any sort of explanation as to why it worked, so it was practically impossible to incorporate into our own program. The greatest leaps in progress were made during hacking session visits, which helped immensely in getting us as far as we did. It was a rollercoaster ride of uphill grinds and thrilling advancement. Looking back, I still wouldn't change the aim of our project. To this day I still feel it will be useful, though it may not be complete now, and its development will continue after this class ends.

Overall I feel that I deserve a(n) _____ in this class. I attended every class (except a Wednesday once, I missed the bus I swear) and worked with my group on each lab to the best of our ability. I was able to grasp the majority of the concepts that we went over, earning grades of 92 and approximately 77 on the first and second tests, respectively. I gained a newfound respect for the amount of sweat and tears that go into creating what can appear to be the simplest of programs. I also gained a new sense of pride for my older brother, a JMU CS Major Alum., who makes his living programming. Before I'd (lightheartedly) call him a nerd but now I'm in awe of the amount of information he has crammed away in his brain to be able to do what he does. This was an awesome class; I really enjoyed my time that I devoted towards it, and I value the things that I have taken from it. This would most likely not be true if anyone but you was the professor. Your unorthodox methods spawned a classroom environment that facilitated learning instead of getting good grades. I know you don't believe in grades, but you definitely deserve an A! (I love ending papers on cheesy notes, my bad.)