Lessons Learned and Future Work

To be honest, going into this project, I wasn’t expecting to come away with it with any real learned lessons. We expected to get from it essentially what we got from any other programming projects or assignments. Sure, we would learn some new coding tricks, pick up a working knowledge of a couple new library features, and maybe come across something that really helps us as a programmer. However, a project of this size and scope made us all progress much more than we expected and sharpened our hindsight, more than a little bit. When making a game there a list of key features the game designers and creators must keep in mind:

1. Don’t underestimate the project, or your final product won’t be as good as it could be. This is not to imply that we aren’t proud of our final product. We created a card game that is both entertaining to play and without experience hindering bugs. But, had we really taken the planning phase to seriously consider all of the possible future complications and perfected our design, our final product could have been better. But, then again, a final product can always be better.
2. Optimal division of labor should be prioritized. This is something I think can only be learned through experience. In other subjects a group project might be a list of 20 questions to be solved by a due date. In this case, the division of labor is simple: simply assign each member to solve a certain number of the problems. In programming, it is far from this simple. It can be difficult to divide the effort because a large portion of the code to be completed will use code that must already be written and because coding styles differ. We, as a group, had trouble with this. Our differences in coding logic and strategy made it difficult at times to make use of code other group members had written. We think a good solution to this would have been to assign different objectives to each member, something we wish we would have realized sooner.
3. Something we learned from this project that was definitely new and refreshing was a greater appreciation of programming and design teams on large gaming projects. Several times throughout the project we caught ourselves imagining the complexities and challenges in coding something like a “AAA” gaming title. Imagine the coding challenges that designing something like a “hit-box” in shooter games entails. The object interaction and constraints you imagine when thinking about it really make you appreciate something like gaming so much more, and in a different way, too.
4. Don’t be afraid to get creative and have fun. At times it’s easy to think so much about finishing the project that you leave creativity at the door. We realized that if you don’t think creatively about the game and don’t actively try to enjoy creating it, the project will become dry and the end product will suffer.

We mentioned earlier that a final product can always be improved upon, and that applies to our final product as well. Our game works as intended, and we had a fun time testing it out after that last line of code was finished. But, there are certainly ways we could continue to improve on our terminal based card game.

1. Gameplay balance. Card games are strategic, and an unbalanced game, even if slightly unbalanced, can greatly hinder the overall experience. Something that could be done in the future is the nerfing and buffing of certain cards to allow the winner to be more reflective of greater strategy and not just luck of the draw.
2. Create more decks to allow for more strategic diversity. As of now we only have two decks, from which 10 cards are drawn. If you look at other highly praised card games such as Hearthstone you will notice that there is a plethora of strategic options to explore. In the future we would create different decks that synergize well and allow for the player to play a different strategy with each different deck.
3. Implementation of a fluid and attractive graphical user interface. Our group did an excellent job creating a user interface that is pleasant to look at and one that does its job well. But a way to make the game better would undeniably be to implement a GUI to make the game take the player to a fictional realm that reflects the theme of the game; and this is just not possible with a game that is played in the console.