Design Paradigm: Object-oriented design/Event-driven design

Software Architecture: We used a three tier design architecture. The three levels are presentation (the Interface and various displays), logic (spread throughout the files), and data (action catalog, card Catalog, unit Catalog). The reason we have used this architecture is so that we may edit the various parts (presentation, logic, and data) easier, without changes to one directly interfering with another. In addition to that we have also chosen this form of software architecture due to the inability of the other types to create the type of program we wished. The Client Server Architecture is focused on communication between a server and a separate client machine, thus making it a poor choice for our single machine program. Peer to peer architecture is much the same, except that instead of the client machine communicating with a central server, it instead communicates directly to another client on the network. Whilst this can allow for computational resources to be shared among separate machines, it is not necessary for our program, and would add unneeded complexity without addressing any of our actual needs. The Pipes and Filters Architecture whilst more applicable to our program than the previous two was discarded due to how it would bottle neck parts of the code without really making the overall structure simpler. Thus it was decided that the best architecture to use was the three tier architecture. It has served us well in this assignment and we are satisfied by it’s use. We will likely use it in the future if applicable.

Design Pattern: Builder, Composite, Decorator, Iterator, Mediator, State,