**Application Controller Pattern**

The Application Controller is a pattern of software development where there is a centralized command process for controlling application components. There are different variations or implementations of this development method, however the general idea is that you have the main program with the code, and then you have a separate class as part of the package that is responsible for controlling and switching to different parts of the application. Usually this is done in “views” for the menu or UI of an application. Basically, in the application code, you create different views based on where the user would navigate and what circumstances / conditions apply to take them there and then code the application controller to switch the application to each respective view based on where the user should be at that point.

There are several benefits to coding an application like this:

*Modularity:* By doing this, the application becomes more modular, which is typically an easier way to work with code, because each piece can be designed, tested, and edited in its own sphere and environment without affecting / damaging other parts of the code. It also allows for smaller implementation cycles and more incremental working progress. By adapting the application controller model, you can build your program view-by-view and test out different features as you add more views and expand upon the design from there.

*Security:* By having the application controller fetch and use different classes or views from other parts of the package, you can make your program more secure by limiting the access to the other parts of code. This way, since the application controller will be the one fetching things instead of all the code being up front and accessed directly, you will have a more secure application that is less likely to be exploited using CSS or SQL injection type attacks.

*Maintenance:* Going hand-in-hand with modularity, implementing the application control design pattern for an application will also improve its ability to be maintained. Code for each view can be updated in increments and steadily maintained as the application evolves into different versions.