Advanced OOP Assessment 1

Daniel Gallagher – G00360986

Design Rationale

For this assessment I decided to use interfaces to help a lot with my program. Interfaces provide great reusability to a project. I started by removing as many dependencies from Alpha that I could. Implementing the two new interfaces Eta and Theta allowed me to do this. Rather than alpha having knowledge of everything being declared inside it I used the interfaces which took this away.

The first interface I implemented was Eta. Eta was used for items of slight low or medium status. Inside alpha we created a new instance of Eta which then was used inside the simple If / else statement provided in the stubs. If the status was Slight or Low, we used Beta which overrode the method inside of the Eta interface. If it was medium status, we did the same, but we used the method inside of gamma to override Eta. This allowed us to take out the declarations of Beta and Gamma from this class and change two dependencies to one. Also, in the process we are taking knowledge away from Alpha, so It is not close to a God class anymore.

The second interface was Theta. If the status was no equal to slight, low or medium we simple entered the switch statement. The statement simply used the Date function and depending on the day returned the method contained in Theta. Depending on the day this function contained inside of Theta would differ and be overridden by the class set to handle it. (. i.e. Monday and Tuesday would return the function from Theta which has been overridden by Delta.) This means that alpha has no knowledge of what goes on in Delta, Zeta, Epsilon, Kappa or Sigma it is essentially just taking the method provided by Theta which has been overridden by our classes depending on the day. This way of coding allows for great readability as it's clear and concise and great reusability as the interfaces can be used very easily by any other classes that may need it.