Renew Implementation Manual

My original UML had to be changed as I developed the project because of realizations I had come to throughout the process. The new UML is similar to my original one in some ways but different in others.

With my original design, I was going to have a habit list class that put the habits into the home scene of my app. I later realized I wanted to have buttons that the user clicked on to input their habit details. When I figured out the buttons, the habit list was no longer needed. The buttons only allow the user to have 12 habits in total at a time. I did this because when creating a habit and implementing it into your own life, it is important to not have too many at one time. Eventually a habit will become a part of the user’s everyday life and they will not need to track anymore; this makes room for more habits to begin to be created which is when the user can then clear the information in the button and start a new one.

The error class that I added into the new UML was because I realized that there were going to have to be many error statements for users to keep the functionality of my app consistent. The errors I needed were for if a user did not enter the name and description of a habit and if a user did not check any of the checkboxes saying they completed their habit for the day.

Each button has its own habit scene which starts off by a drop-down menu for choices of what category their habit falls under which is a string array. After the user chooses a habit category, another drop down menu for their choice of a sub habit category which are all unique to each class. Exercise, Financial, Health, Wellbeing and AroundTheHouse have their own choices which will be presented based on what the user chose in the previous drop-down menu. In this new UML, I changed the class names because I felt that they were more unique and allowed the user to have a multitude of options.

The new methods in each class are based around each type’s subcategory. I also eliminated the previous way I wanted to calculate the statistics. There are statistics buttons across from each habit button which calculates their average percentage and what days they did it on through using the checkboxes that are all done in the GUI.

The new UML is provided below and shows the implementations I did throughout this process, this includes, the change of class names, each classes sub habit category, and the new error class. These implementations allowed for Renew to be simple and functional all throughout.

