Group 1   
Project 2 State machines  
Cory, Gabe, Lenneth, Jose

Analysis of FSM Refrigerator

The Refrigerator application was programmed in Eclipse using JavaFX and Java. The Main class contains the main method which is used to execute the GUIDisplay, which is the entry point in the program. The GUI gets a reference to the singleton RefrigeratorContext class at the start. The RefrigeratorContext has an attribute called currentState which holds a reference to a state object. RefrigeratorContext is then able to maintain a collection of state transitions. In addition, the RefrigeratorContext instantiates and holds a reference to a thermometer object, a class we decided would hold temperature information.

We have an abstract object GUIButton which extends the JavaFX Button class. All clickable buttons on the GUI implement GUIButton. When a user clicks a GUIbutton, the button calls the handleEvent method in RefrigeratorContext and passes it an event singleton. The RefrigeratorContext then passes the event to the currentState which would have actions based on the current state and the type of event.

The states are also singletons which implement a RefrigeratorState interface. Each state has an enter and leave method. Whenever a state is entered, a timer is started which is tied to a thermometer class. This is so we could process temperature changes as time goes on. When the timer ticks, the current state handles the change in temperature accordingly. The new temp value is checked to see if it is in range or else it will pass an event to the RefrigeratorContext.

We decided on using conditions to check all temperature constraints because it allows us to keep a lower count of States and Events and in turn, keeps the program more organized a readable/writeable.

We have included in the submission a Javadoc so that you can view and read the methods, fields for all classes.