Criterion B: Design

The program was design via the front and back end. The back-end consisted of logic of the application, and the front-end which consisted of the development of the GUI.

Logic:

The logic of the program consists of various methods that will be used to convert specific units that are called within the program. These methods are called by the GUI then return the desired value to complete the calculation for the user.

User Input:

The user must select options and input values to complete the desired calculations. The user must specify the unit group which contains the same types of units (temperature, distance, time, etc.) this will prevent the user from attempting to convert between illogical groups, such as a time to a temperature. The user must then specify the amount imputed to be calculated.

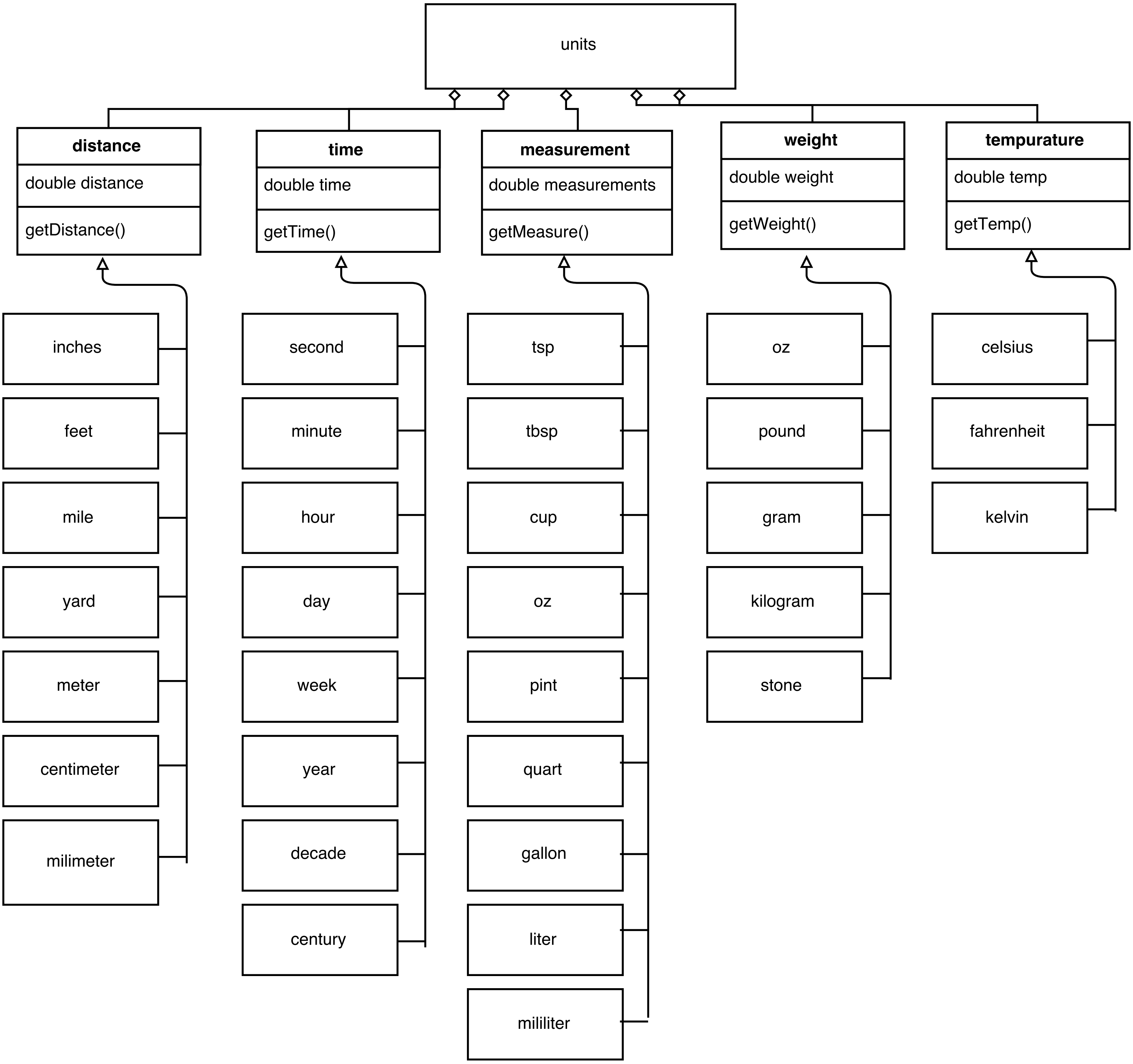
Generate solution:
The algorithm behind this program is very simple. The unit group selected will indicated what units, and what the base unit will be. This means that when the user enters their input value and unit, the program will run the method that converts that unit into the base unit for that unit group. The program then reads the output unit specified by the user and specifies the base units into the desired unit, returning the completed calculation. When the user clicks the 'Calculate' button, the GUI will return the stored value as text within the GUI. The stored value is then reset for the next calculation.

Action tested	Expected outcome
Conversion methods work successfully	Test the methods through a tester program that will input the calculations through the console. And make sure the calculations are correct
Scanner/input work correctly	The program will read and store the input within the console
GUI actions and listeners work successfully	Action listeners are read through the GUI and output something within the console to check the functionality
Inputs are read successfully	Program stores input through the GUI
Calculations are output correctly	Outputs match correct calculations through methods and the input read from the GUI
Stored values erased for next use	Values are erased, and reset for another calculation

User Interface: ConverterFrame is the JFrame. It displays a row of radio buttons on the SOUTH panel to allow the user to select their desired unit group. The NORTH frames contain the input and output combo boxes which change depending on the radio button selected. The user must choose a unit within both the input and output combo boxes. Also within this panel is a textField in which the user types their desired initial amount to be converted. The ‘calculate’ button on the EAST panel uses a button listener to send the calculation to the program and return the output. The result appears within the SOUTH panel as text.

Application is a unit converter program in which the user selects a unit (in a category or family) and another in the same category and types a value to be calculated.

I am not sure if I am doing this right (the UML flowchart). Or what things need to go under each box/level. So don't judge too harshly :'(


units
distance
double distance
getDistance()
time
double time
getTime()
measurement
double measurements
getMeasure()
weight
double weight
getWeight()
tempurature
double temp
getTemp()
inches
feet
mile
yard
meter
centimeter
milimeter
second
minute
hour
day
week
year
decade
century
tsp
tbsp
cup
oz
pint
quart
gallon
liter
mililiter
oz
pound
gram
kilogram
stone
celsius
fahrenheit
kelvin