

Sample report template for design doc

1. Class/ objects and organization

(insert diagram – boxes and arrows are fine) of the objects you used, their classes, and how they are related. If you have defined a new class (using the **extends** keyword) show this. Note: this is a class diagram, not a visual hierarchy. It should communicate what class has created what. So, for example. This code snippet:

```
public class energyJournal implements ActionListener,
WindowListener
{
    public JFrame MyWindow = new JFrame("Energy Journal");

    public JButton = MenuBtn1,MenuBtn2,MenuBtn3;

    MenuBtn1 = new JButton ("Command 1");
    MenuBtn2 = new JButton ("Command 2");
    MenuBtn3 = new JButton ("Command 3");
    JLabel textLbl = new JLabel("Label");

    public JPanel WinPanel = new JPanel();
    WinPanel.setLayout(new BorderLayout());
    WinPanel.add(MenuBtn1,...);
    WinPanel.add(textLbl,...);
    (add additional buttons here)

    MyWindow.setContentPane(WinPanel);

    MenuBtn1.addActionListener(this);
    .....

    public void actionPerformed (ActionEvent e)
    {
        String btnCmd = e.getActionCommand();
        textLbl.setText(btnCmd);

        //textLbl.setText(e.getActionCommand());
    }
}
```

should have a diagram that looks something like this (Figure 1):

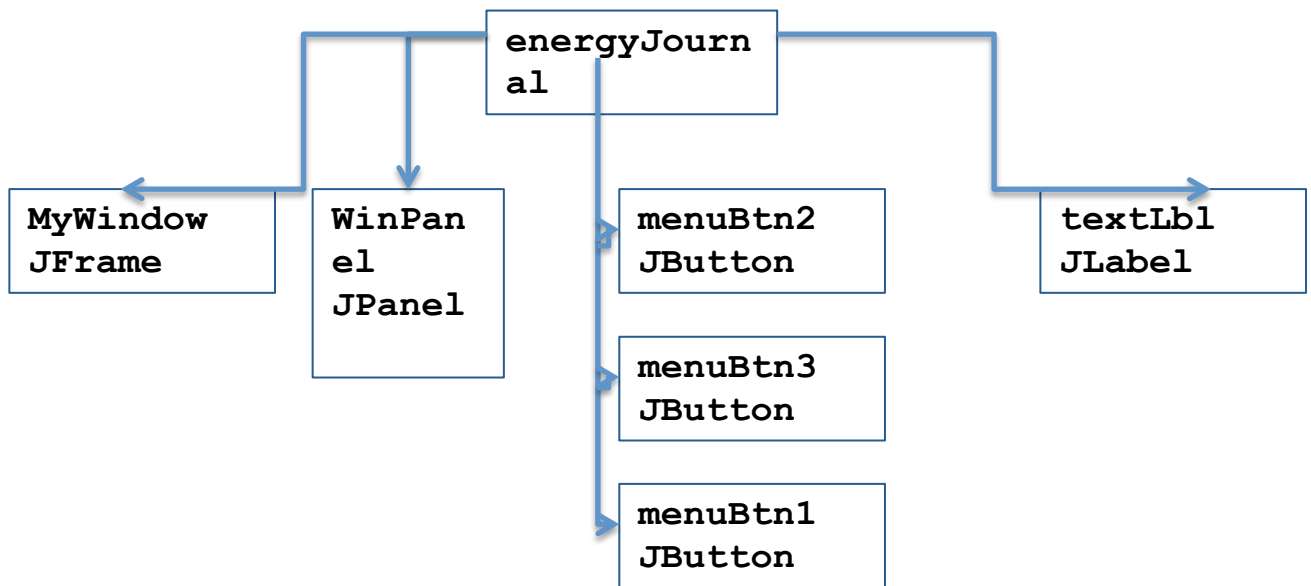


Figure 1. class organisation

Describe the interactions you are building with a diagram something like this:

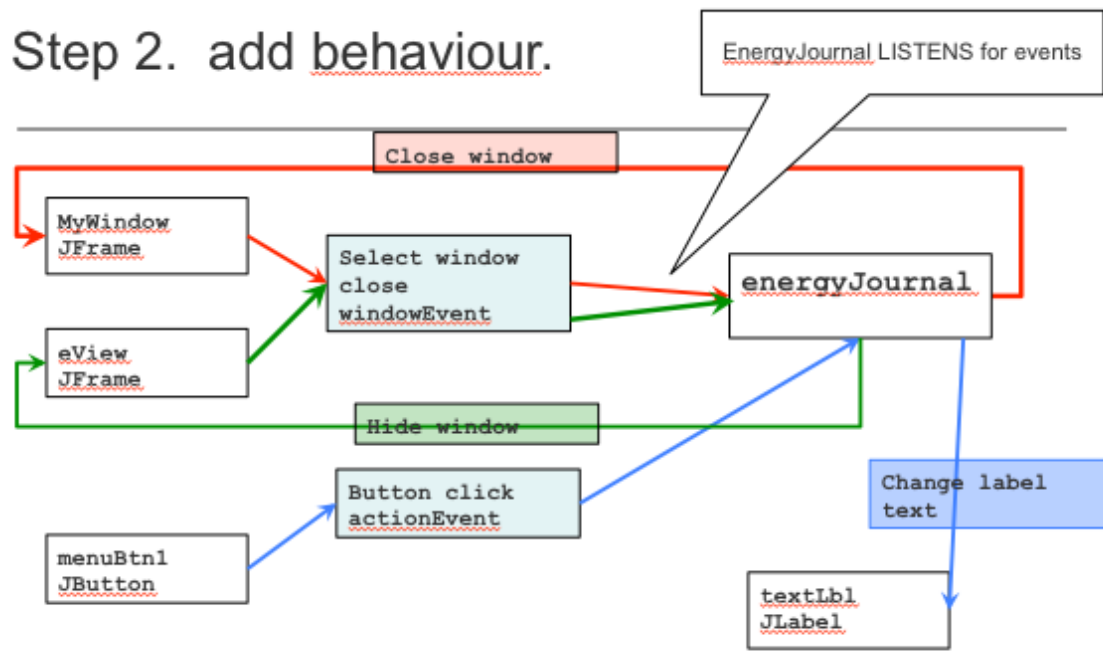


Figure 2. Interactions

If your interaction diagram gets too complicated do it in pieces.

Provide a brief description of what each object does. This should include:

What kind of object it is (subclassed from? Instance of?)

What parameters it takes;

What objects it contains, if any;

What input it expects ;

What behaviour (functions) it has;

What output (if any) it generates.

A bulleted list is fine.

NOTE: IF you have two objects that are the same class and do exactly the same thing, you only need to explain **one**. If they are the same thing but generate different behaviour, (as in two menus), describe each.

Simple example:

WinPanel

- JPanel
 - Takes in background colour, name
 - Contains 2 buttons : Menu1, Menu2(Jmenu)
 - Contains a text display TextView (JLabel)
 - Gets input from each button (a string)
 - Changes the content of TextView to the string received
- Make sure you are clear about where the input is processed!

2. Visual organisation

You can provide a rough visual wireframe of what it is supposed to look like.

3. Issues

If you did not get it working properly, describe issues here.

4. Pair Programming log

Put your PP report of time spent here.