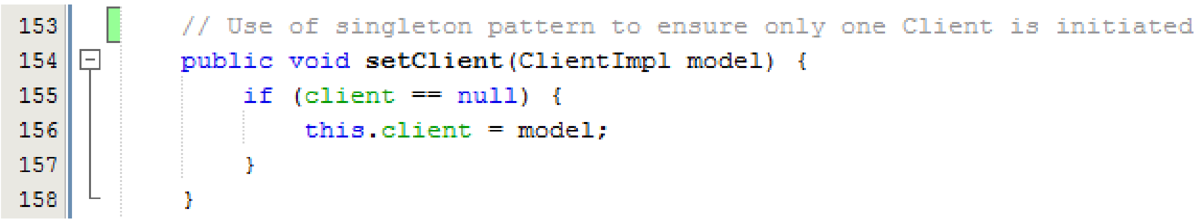
Appendices H

Singleton Pattern



```
// Use of singleton pattern to ensure only one Client is initiated
public void setClient(ClientImpl model) {
    if (client == null) {
        this.client = model;
    }
}
```

Fig x – Extract from HomeForm, setClient()

As you can see from fig x, I have used the singleton pattern within the HomeForm class, which is a class that represents the home screen graphical user interface (GUI). I decided to use the singleton pattern for each GUI class as well as a few other instances, to ensure that only 1 instance of the ClientImpl class can be assigned to a Client variable.

This is because, a GUI class should only be used for 1 client only, and therefore should not be able to be reassigned to a different Client instance.