

09-03-09 - Weg met de Swing-layout!

Iedere beginnende Java-programmeur zal vroeg of laat met de Swing-class te maken krijgen. Mits je natuurlijk van plan bent om Graphical User Interfaces (GUI's) te gaan bouwen. Je zult dan al snel merken dat de alle componenten, zoals JFrame, JMenuBar en JButton, een blauw design krijgen. Hieraan kun je direct een Swing-applicatie herkennen.

Ik hoor regelmatig van studenten dat ze deze blauwe overkill niet mooi vinden en liever echt een applicatie ontwikkelen die past bij het Operating System (OS). Na een tip van een collega kwam ik uiteindelijk op de volgende code uit:

```
<pre lang="PHP" line="1?>
import javax.swing.*;

public class mijnFrame extends JFrame {
    public mijnFrame {
        try {

            UIManager.setLookAndFeel(UIManager.getSystemLookAndFeelClassName());
            SwingUtilities.updateComponentTreeUI(this);
        } catch ( Exception e ) {}
        setDefaultCloseOperation ( EXIT_ON_CLOSE );
        setSize ( 500 , 500 );
        setVisible ( true );
    }
}
</pre>
```

Hiermee zal het thema van het OS van de gebruiker worden toegepast op de GUI. Een JButton onder Windows 2000 zal er dus gewoon grijs uit komen te zien, terwijl onder Mac OS een meer glossy layout toegepast zal worden.

Met deze code kun je dus eenvoudig verbergen dat je met een Swing-applicatie te maken hebt. Het zal altijd een kwestie van smaak blijven, maar ik steun de voorstanders van OS-dependent vensters. Ik wil namelijk een programma dat past binnen het OS dat ik geïnstalleerd heb.