

## Proyecto Patrones

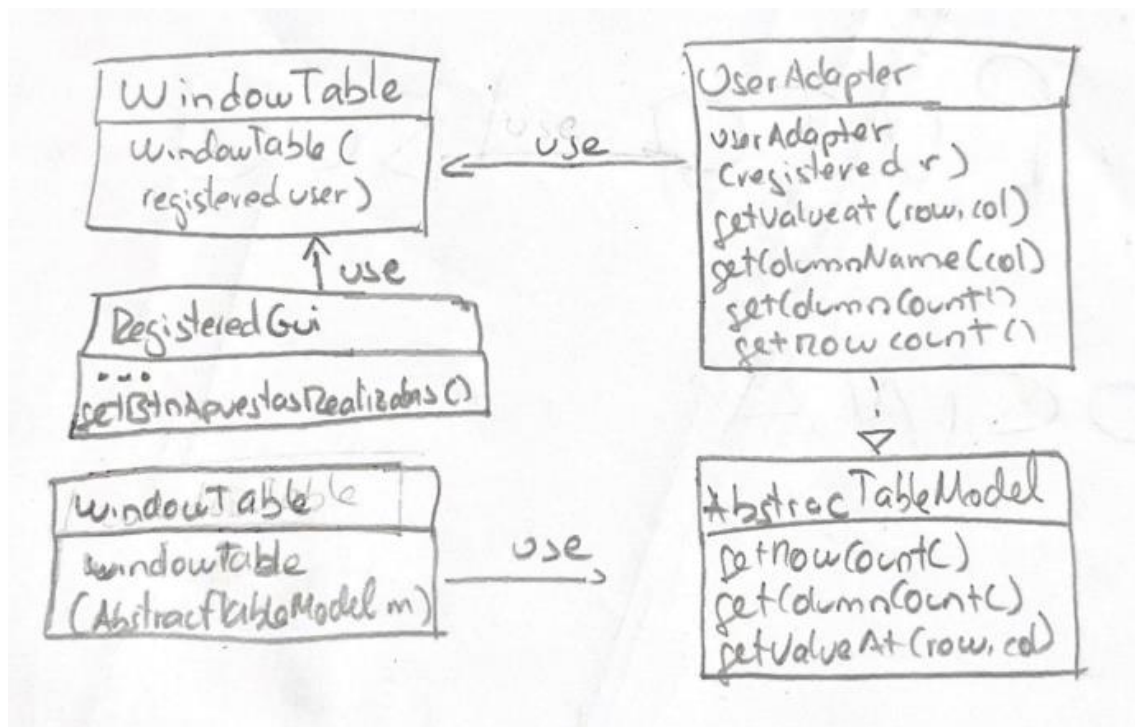
### Contenido

Proyecto Patrones .....	1
EJERCICIO 3 PATRON ADAPTER.....	1
UML con las nuevas clases .....	1
Codigo modificado .....	1
Capturas mostrando la ejecución.....	4

### EJERCICIO 3 PATRON ADAPTER

Realizado por Iñaki Inda Araguás

UML con las nuevas clases



Codigo modificado

En RegisteredGUI.java he creado un botón para poder acceder a la ventana que muestra la información con todas las apuestas realizadas por un usuario

```

private JButton getBtnApuestasRealizadas() {
    if (btnApuestasRealizadas == null) {
        btnApuestasRealizadas = new
JButton(ResourceBundle.getBundle("Etiquetas").getString("APREALIZADAS"));
//$NON-NLS-1$ //$NON-NLS-2$
        btnApuestasRealizadas.setFont(new Font("Tahoma",
Font.PLAIN, 16));
        btnApuestasRealizadas.setForeground(Color.DARK_GRAY);
        btnApuestasRealizadas.setBackground(Color.PINK);
        btnApuestasRealizadas.addActionListener(new
ActionListener() {
            public void actionPerformed(ActionEvent e) {
                WindowTable vt = new WindowTable(user);
                vt.setVisible(true);
            }
        });
        btnApuestasRealizadas.setBounds(10, 391, 282, 68);
    }
    return btnApuestasRealizadas;
}

```

Este botón crea una nueva instancia de la clase WindowTable

```
package gui;
```

```

public class WindowTable extends JFrame {
    private Registered user;
    private JTable tabla;

    public WindowTable(Registered user) {
super("Apuestas realizadas por " + user.getUsername() + ":");
        this.setBounds(100, 100, 700, 200);
        this.user = user;
        UserAdapter adapt = new UserAdapter(user);
        tabla = new JTable(adapt);
        tabla.setPreferredSize(new Dimension(500,
70));

        JScrollPane scrollPane = new JScrollPane(tabla);
        getContentPane().add(scrollPane, BorderLayout.CENTER);
    }
}

```

La cual crea una ventana e instancia la clase UserAdapter.

En esta clase en el método getValueAt accedo a las apuestas de un usuario y para cada apuesta que ha realizado recupero los datos

```
package adapter;
```

```

import java.util.ArrayList;

import javax.swing.table.AbstractTableModel;

import domain.ApustuAnitza;
import domain.Apustua;
import domain.Registered;

public class UserAdapter extends AbstractTableModel{
    private Registered user;
    private String[] headers = new String[]{"Event", "Event Date",
"Question", "Bet(€)"};
    private ArrayList<Apustua> apuestas;
    public UserAdapter(Registered r) {
        this.user=r;
        apuestas = new ArrayList<Apustua>();
        for(ApustuAnitza a : user.getApustuAnitzak()) {
            apuestas.addAll(a.getApustuak());
        }
    }
    @Override
    public Object getValueAt(int row, int col) {
        Object value = "";
        switch (col) {
            case 0:
                value =
apuestas.get(row).getKuota().getQuestion().getEvent().getDescription();
                break;
            case 1:
                value = apuestas.get(row).getApustuAnitza().getData();
                break;
            case 2:
                value =
apuestas.get(row).getKuota().getQuestion().getQuestion();
                break;
            case 3:
                value = apuestas.get(row).getApustuAnitza().getBalioa();
                break;
        }
        return value;
    }
    @Override
    public String getColumnName(int col) {
        return headers[col];
    }
    @Override
    public int getColumnCount() {
        return 4;
    }
    @Override
    public int getRowCount() {
        return apuestas.size();
    }
}

```

Capturas mostrando la ejecución

