

¿Qué ofrece Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.
 Ese apoyo que siempre quiso tener...

1. Desarrollo de componentes y proyectos a medida



2. Auditoría de código y recomendaciones de mejora

3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces,
 HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)
 Gestor de contenidos (Alfresco)
 Aplicaciones híbridas

Tareas programadas (Quartz)
 Gestor documental (Alfresco)
 Inversión de control (Spring)

Control de autenticación y
 acceso (Spring Security)
 UDDI
 Web Services
 Rest Services
 Social SSO
 SSO (Cas)

JPA-Hibernate, MyBatis
 Motor de búsqueda empresarial (Solr)
 ETL (Talend)

Dirección de Proyectos Informáticos.
 Metodologías ágiles
 Patrones de diseño
 TDD

BPM (jBPM o Bonita)
 Generación de informes (JasperReport)
 ESB (Open ESB)



[Home](#) | [Quienes Somos](#) | [Empleo](#) | [Foros](#) | [Tutoriales](#) | [Servicios Gratuitos](#) | [Contacte](#)

	<p>Tutorial desarrollado por: Roberto Canales Mora 2003-2005 Creador de AdictosAlTrabajo.com y</p> <p>Director General de Autentia S.L.</p> <p>Recuerda que me puedes contratar para echarte una mano:</p> <p>Desarrollo y arquitectura Java/J2EE Asesoramiento tecnológico Web Formación / consultoría integrados en tu proyecto</p> <p>No te cortes y contacta: 655 99 11 72 rcanales@autentia.com.</p>	
---	--	---

Descargar este documento en formato PDF [temporizador.pdf](#)

[Contabilidad y Laboral](#)

Software ultima generacion
Empresas y Profesionales

[Análisis Financiero.](#)

Software de análisis financiero.
Auto-analice su empresa fácilmente.

[Master Java J2ee Oracle](#)

Prácticas laborales 100%
aseguradas Nuevo temario de Struts. Trabaja ya

[Curso Web J2EE](#)

Curso Avanzado en Desarrollo
Web con J2EE

Anuncios Goooooogle

Anunciarse en este sitio

Análisis de disponibilidad de un Web

Cuando somos responsables de un Web empresarial, debemos cuidar el nivel de servicio que proporcionamos a nuestros clientes. Normalmente se da la circunstancia de que cuando se cae nuestro Web nos enteramos por una protesta de un cliente o un usuario de negocio.

Hay veces que contratamos un Hosting y llegamos a un acuerdo de nivel de servicio (SLA). Más habitualmente de lo que me gustaría poder contar, estos niveles de servicio no se respetan.

Debemos tener la capacidad de medir estos servicios por nuestros propios medios. Hace no mucho, os mostrábamos un tutorial sobre una herramienta profesional que cubre de sobra esta funcionalidad, llamada e-test ([Rendimiento de aplicaciones Web](#)).

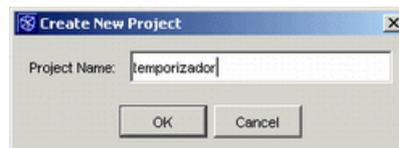
A mi personalmente me gusta tener la capacidad de verificar por mis propios medios (y desde mi casa con ADSL) si el sistema funciona bien o mal. Os recomiendo una herramienta llamada [WebSupervisor](#), que puede cubrir esta funcionalidad.

Si sabéis un poco de Java, es bastante sencillo construir un pequeño programa para este propósito... y con algún valor añadido.

Vamos a empezar a hacerlo paso a paso. El objetivo es construir el esqueleto de lo que podrá ser una herramienta de monitorización y mostrar algunas notas para poder darle continuidad.

Creación de la funcionalidad básica

Creamos un nuevo proyecto en NetBeans 3.6



Debemos ser consciente que el interfaz de usuario es lo menos importante a la hora de diseñar un programa. Debemos intentar acotar la funcionalidad y hacerla independiente de como sea invocada.

```
public String EjecutarTarea() {
    try
    {
        // inicializamos el destino si es necesario
        if(destino == null)
        {
            destino = new URL("http://www.adictosaltrabajo.com/");
        }
    }
}
```

```

// abrimos la conexión al servidor
URLConnection enlace = (URLConnection)destino.openConnection();
String respuesta = " - Código: " + enlace.getResponseCode();

//respuesta += recuperaContenido(enlace);

return respuesta;
}
catch(Exception e)
{
    return "Error " + e.getMessage() + "\n";
}
}

// esta función recupera el texto de la página
String recuperaContenido(URLConnection enlace)
{
    try
    {
        String respuesta = "";
        BufferedReader in = new BufferedReader (new InputStreamReader(enlace.getInputStream()));

        String cadena = "";

        while (cadena != null)
        {
            cadena = in.readLine();
            if (cadena != null)
            {
                respuesta+="(\n" + cadena);
            }
        }

        in.close();
        return respuesta;
    }
    catch(Exception e)
    {
        return "Error " + e.getMessage() + "\n";
    }
}
}

```

Con la función anterior, tenemos la capacidad de conectarnos a una URL y recuperar el código de retorno, e incluso recuperar el contenido de la página (`//respuesta += recuperaContenido(enlace);`)

El diseño inicial no es muy bueno así que tenedlo en cuenta en vuestros desarrollos (esto solo es un tutorial parcial) :

- La URL esta fija y dentro del código
- No se gestionan ni propagan excepciones

Temporizar una función

Java dispone de una clase para planificar la ejecución de una función de un modo periódico, sin bajar al detalle de los hilos (Threads)
Timer

```

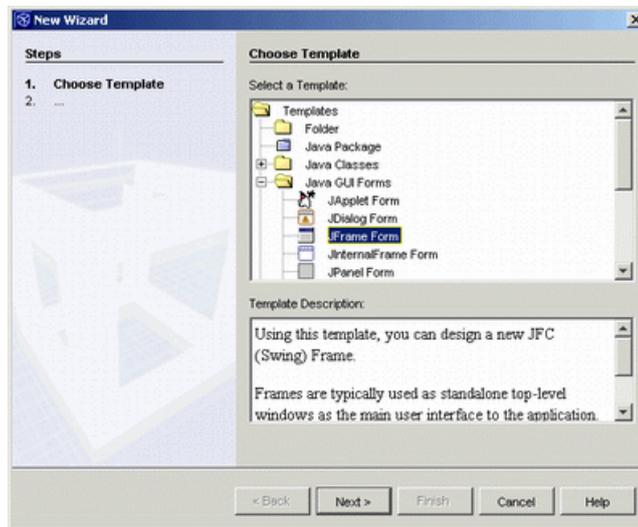
public void Arrancar()
{
    java.util.Timer timer = new java.util.Timer();

    // creamos una clase inline
    timer.scheduleAtFixedRate(new TimerTask() {
        public void run() {
            String resultado = EjecutarTarea();
            AddTexto("" + new Date().toString() + resultado );
        }
    }, retardo, periodo);
}

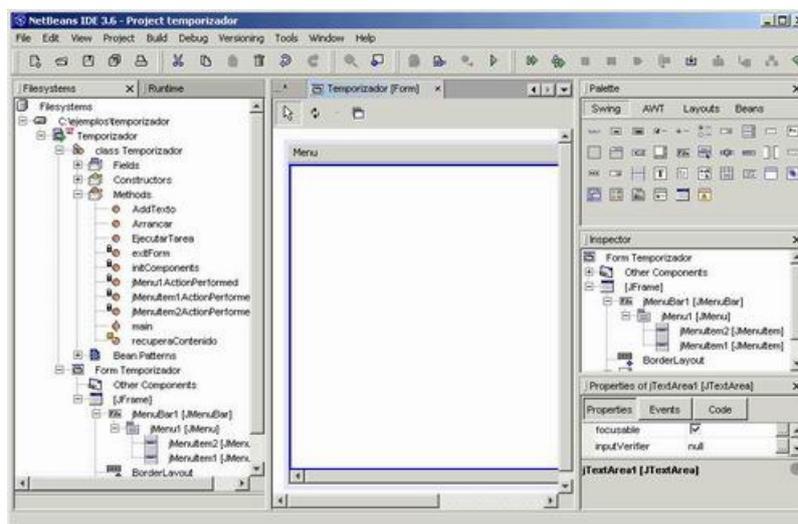
```

Creación del Interfaz de Usuario

Usando las capacidades de NetBeans, vamos a crear un interfaz sencillo



NetBeans es un entorno bastante intuitivo, aunque puede costar un poco cogerle el tacto...



Como primera aproximación podemos obtener el siguiente código

```

/*
 * Temporizador.java
 *
 * 1 Julio, 2004
 * El objetivo de este programa es comprobar el estado de una página Web
 *
 * Es un ejemplo parcial ...
 */
import java.util.*;
import javax.swing.*;
import java.net.*;
import java.io.*;

/**
 *
 * @author Roberto Canales
 */
public class Temporizador extends javax.swing.JFrame {

    /** Creates new form Temporizador */
    public Temporizador() {
        initComponents();
    }

    /** This method is called from within the constructor to
     * initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is
     * always regenerated by the Form Editor.
     */
    private void initComponents() {
        jScrollPane1 = new javax.swing.JScrollPane();
        jTextArea1 = new javax.swing.JTextArea();
        jMenuItem1 = new javax.swing.JMenuItem();
        jMenuItem2 = new javax.swing.JMenuItem();
        jMenuItem3 = new javax.swing.JMenuItem();
    }
}

```

```

setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);
setTitle("Ejemplo de Temporizador");
addWindowListener(new java.awt.event.WindowAdapter() {
    public void windowClosing(java.awt.event.WindowEvent evt) {
        exitForm(evt);
    }
});

jTextArea1.setPreferredSize(new java.awt.Dimension(640, 480));
jScrollPane1.setViewportView(jTextArea1);

getContentPane().add(jScrollPane1, java.awt.BorderLayout.CENTER);

jMenu1.setText("Menu");
jMenu1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jMenuItem2ActionPerformed(evt);
    }
});

jMenuItem2.setText("Arrancar");
jMenuItem2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jMenuItem2ActionPerformed(evt);
    }
});

jMenu1.add(jMenuItem2);

jMenuItem1.setText("Salir");
jMenuItem1.setToolTipText("Menu principal");
jMenuItem1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jMenuItem1ActionPerformed(evt);
    }
});

jMenu1.add(jMenuItem1);

jMenuBar1.add(jMenu1);

setJMenuBar(jMenuBar1);

pack();
}

private void jMenuItem2ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    Arrancar();
}

private void jMenuItem1ActionPerformed(java.awt.event.ActionEvent evt) {
    dispose();
}

private void jMenuItemActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
}

/** Exit the Application */
private void exitForm(java.awt.event.WindowEvent evt) {
    System.exit(0);
}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    new Temporizador().show();
}

// la función Arrancar simplemente crea un temporizador
// y ejecuta una función periódicamente
public void Arrancar()
{
    jMenuItem2.setEnabled(false);

    java.util.Timer timer = new java.util.Timer();

    // creamos una clase inline
    timer.scheduleAtFixedRate(new TimerTask() {
        public void run() {
            String resultado = EjecutarTarea();
            AddTexto("" + new Date().toString() + resultado );
        }
    }, retardo, periodo);
}

// la tarea a ejecutar es independiente del interfaz de usuario
public String EjecutarTarea() {
    try
    {
        // inicializamos el destino si es necesario
        if(destino == null)
        {

```

```

        destino = new URL("http://www.adictosaltrabajo.com/avisos");
    }

    // abrimos la conexión al servidor
    HttpURLConnection enlace = (HttpURLConnection)destino.openConnection();
    String respuesta = " - Código: " + enlace.getResponseCode();

    //respuesta += recuperaContenido(enlace);

    return respuesta;
}
catch(Exception e)
{
    return "Error " + e.getMessage() + "\n";
}
}

// esta función recupera el texto de la página
String recuperaContenido(HttpURLConnection enlace)
{
    try
    {
        String respuesta = "";
        BufferedReader in = new BufferedReader (new InputStreamReader(enlace.getInputStream()));

        String cadena = "";

        while (cadena != null)
        {
            cadena = in.readLine();
            if (cadena != null)
            {
                respuesta+="("\n" + cadena);
            }
        }

        in.close();
        return respuesta;
    }
    catch(Exception e)
    {
        return "Error " + e.getMessage() + "\n";
    }
}

// añadimos texto al componente en pantalla
public void AddTexto(java.lang.String nodo) {
    jTextArea1.append(nodo + "\n");
}

// Variables declaration - do not modify
private javax.swing.JMenu jMenuItem1;
private javax.swing.JMenuBar jMenuItemBar1;
private javax.swing.JMenuItem jMenuItem2;
private javax.swing.JMenuItem jMenuItem1;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTextArea jTextArea1;
// End of variables declaration

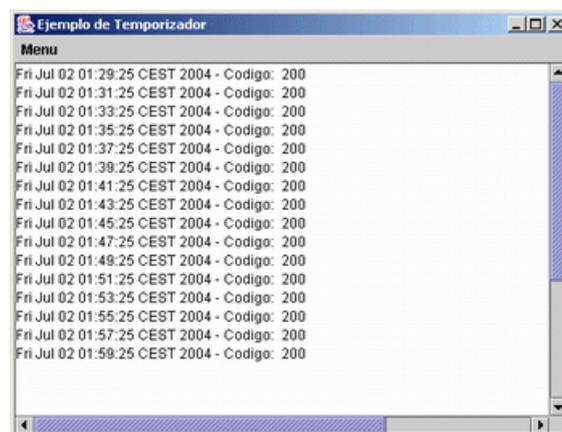
// inicializamos las variables no gestionadas por el compilador NetBeans 3.6
private URL destino = null;

private int retardo = 0; // no esperar
private int periodo = 2 * 60 * 1000; // ejecutar cada 5 minutos
}

```

Ejecución

Si ejecutamos el programa, obtenemos la siguiente respuesta... Nuestro Hosting de www.adictosaltrabajo.com es bastante estable.

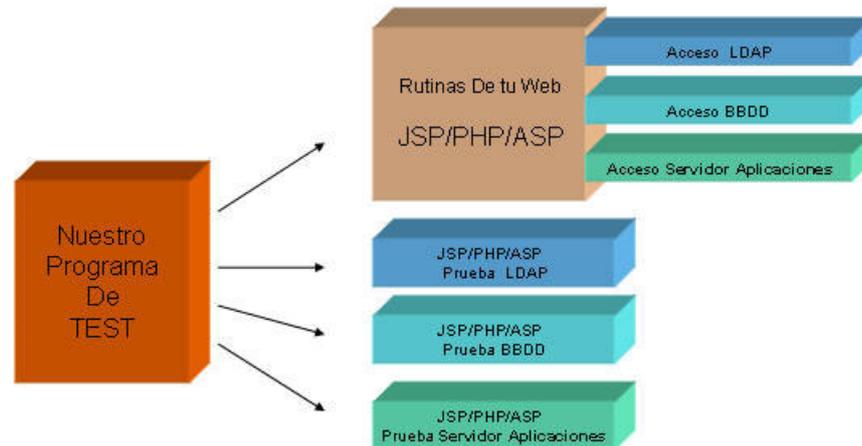


Otros usos adicionales

Si somos un poco hábiles, podemos determinar las dependencias de nuestro sistema Web a otros subsistemas y crear un pequeño JSP/PHP/ASP que sea capaz de probar si tenemos acceso a ellos:

- LDAP
- Base de Datos
- Host transaccional
- Servidor de Aplicaciones
- Sistema remoto de ficheros
- Middleware
- etc...

Podríamos también modificar nuestro programa para que, al encontrar un error en la página probada, recorriendo cada una de las rutinas que prueban cada subsistema individualmente, analizase quién lo está produciendo.



De este modo, cuando se produzca un problema en producción, desde vuestra casa podréis prever errores (siempre y cuando lo que falle no sea el propio servidor Web).

Con un sistema de estas características (que se puede desarrollar en día) podéis tanto medir la estabilidad del sistema como adelantarnos a la llamada del usuario de negocio cabreado porque la aplicación, en medio de una demo, lleva media hora sin funcionar. Además, si cuando se produce un error, mandáis un email a una pasarela SMS, te enterarás sin tener que estar atento al correo.

Otra mejora puede consistir en monitorizar al mismo tiempo un Web muy estable (como google) y solo considerar los datos como válidos cuando esté último de una respuesta correcta ya que sería una pena que lo que fallase fuera nuestra configuración local ;-))

- Enlace ADSL
- Red Inalámbrica
- Configuración TCP/IP
- Desconexión de algún cable (que pasa muy habitualmente)

Conclusiones

Muchos sabréis que hay un montón de causas por las que fallan las aplicaciones y normalmente es por falta de procedimientos organizativos a la hora de operar sistemas:

- Lagunas de memoria en programas (normalmente contruidos por nosotros).
- Cambios de contraseñas en un turno que se aplican al rea-rrancar en otro.
- Caducidad de certificados digitales.
- Modificaciones aparentemente inocuas del LDAP.
- Llenado de los discos o espacio de tablas de base de datos.
- Borrado accidental de ficheros/campos.
- Falta de pruebas automáticas de regresión funcional
- Y un largo etc

Ampliando este ejemplo un poquito, las posibilidades son interesantes:

- Obtener una gráfica de disponibilidad
- Enviar un correo electrónico cuando el código de retorno la petición cambie
- Analizar el contenido HTML para verificar que no cambia... o es sustituido (nos han jaqueado)
- Asegurarnos que nuestras fuentes de contenido (sindicado o alimentado por autores) funcionan correctamente

Determinar la disponibilidad, rendimiento y capacidad de los sistemas, requiere de práctica y técnica.

Adquirir una herramienta puede ser muchas veces más interesante que construirla aunque puede ser un negocio rentable (o un buen proyecto de fin de carrera con salida comercial) para la demanda creciente de estos servicios

[Sobre el Autor...](#)

Si desea contratar formación, consultoría o desarrollo de piezas a medida puede contactar con

Creatividad Internet

[Autentia S.L.](#) Somos expertos en:
J2EE, C++, OOP, UML, Vignette, Creatividad ..
 y muchas otras cosas

Nuevo servicio de notificaciones

Si deseas que te enviemos un correo electrónico cuando introduzcamos nuevos tutoriales, inserta tu dirección de correo en el siguiente formulario.

Subscribirse a Novedades	
e-mail	<input type="text"/>
	<input type="button" value="Enviar"/>

Otros Tutoriales Recomendados ([También ver todos](#))

Nombre Corto	Descripción
WebDav con IIS	En este tutorial podemos aprender como gestionar los contenidos de nuestro servidor Web sin necesidad de utilizar FTP.
Introducción al Java Web Start	Os mostramos como podéis distribuir aplicaciones de consola de un modo sencillo a través de Java Web Start. Es el mismo principio de los applets aplicado a aplicaciones.
XML y XSL en Cliente	En este tutorial os enseñamos como formatear documentos XML directamente en vuestro navegador a través de Plantillas XSL. En cursos sucesivos veremos como hacerlo en el servidor, para no crear dependencias con el navegador del cliente.
Generación .exe con ficheros Java	Os mostramos como, utilizando la herramienta gratuita JSmooth, podemos generar un fichero .exe a partir de nuestros programas Java
Motor de Búsqueda en Tu Web	En este tutorial te enseñamos como configurar un buscador externo para proporcionar capacidades de búsqueda a tu Web. El buscador en este ejemplo es Atomz.
Apache, MySQL y PHP	Os mostramos como configurar Apache, MySQL y PHP en vuestra máquina
JSP 2.0, JSTL y Lenguaje de expresiones	Os mostramos las novedades de JSP 2.0: Nuevas librerías estandar de etiquetas y el lenguaje de expresiones con ejemplos de acceso a base de datos, XML y XSL en JSP
Como crear un Cron en Unix	Cuando desarrollamos aplicaciones, numerosas veces defemos programar tareas que se ejecuten periodicamente. Os ensañamos a hacerlo en Unix
Generación de Informes pdf con DataVision	Os mostramos como simplificar la creación de informes un múltiples formatos, incluyendo pdf, con una fantástica herramienta visual (DataVision). Os mostramos como integrarla con Servlets
Rendimiento de aplicaciones Web	En este tutorial veremos una introducción al funcionamiento de la Suite e-Test de Empirix.

Nota: Los tutoriales mostrados en este Web tienen como objetivo la difusión del conocimiento.

Los contenidos y comentarios de los tutoriales son responsabilidad de sus respectivos autores.

En algún caso se puede hacer referencia a marcas o nombres cuya propiedad y derechos es de sus respectivos dueños. Si algún afectado desea que incorporemos alguna reseña específica, no tiene más que solicitarlo.

Si alguien encuentra algún problema con la información publicada en este Web, rogamos que informe al administrador rcanales@adictosaltrabajo.com para su resolución.

[Patrocinados por enredados.com Hosting en Castellano con soporte Java/J2EE](#)

