

# Javadoc con NetBeans

## Javadoc y NetBeans 5.0

Javadoc es una herramienta creada por Sun Microsystems para generar documentación en formato HTML a partir de los comentarios de documentación incluidos en el código fuente. Los comentarios de documentación deben seguir el estándar Javadoc.

NetBeans incluye también esa herramienta y un número de características para facilitar la generación de la documentación.

## Comentarios de Documentación en los Archivos Fuente

Podemos agregar comentarios de documentación para las clases, interfaces, constructores, métodos y atributos. La herramienta Javadoc extrae esos comentarios de los archivos fuente y los pone en formato HTML en archivos de documentación.

Un comentario de documentación empieza con los caracteres `/**` y termina con los caracteres `*/`. Cada comentario consiste de una descripción seguida de una o más etiquetas. Se pueden usar etiquetas de formateo HTML en los comentarios de documentación.

Sugerencias:

- Los caracteres del inicio de comentario de documentación `/**` deben tener la misma sangría que el código documentado.
- Empiece las siguientes líneas de comentario con un `*` con la misma sangría.
- Empiece la descripción en la siguiente línea a la que tiene los caracteres `/**`.
- Inserte un espacio en blanco entre el `*` y el texto descriptivo o la etiqueta.

Por ejemplo, el siguiente es un comentario de documentación de un método:

```
/**
 * Este metodo regresa la clave del genero
 * @return La clave del genero
 */
```

```
public String getCveGenero() {
    return cveGenero;
}
```

En el archivo HTML generado el comentario se muestra en la figura 2.1:

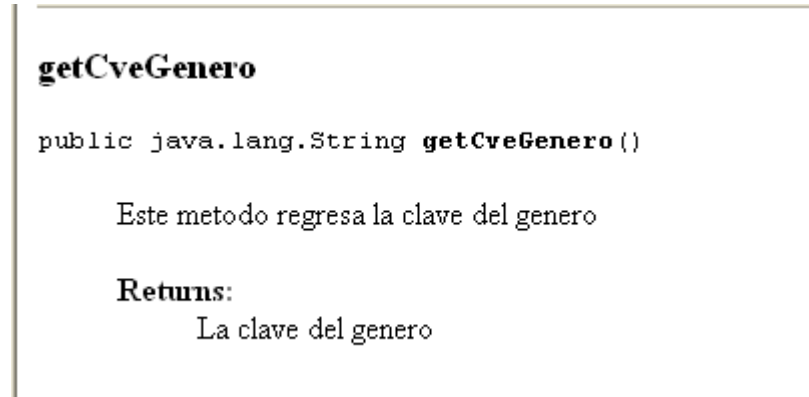


Figura 1

Note como Javadoc convirtió la etiqueta `@param` en un encabezado. También le agregó un guión que separa el nombre del parámetro de su descripción. Cuando se escriba la parte descriptiva de un comentario, haga la primera línea un resumen. Javadoc copia la primera oración del comentario a la tabla de resumen de la clase, interfaz o miembro.

## Ubicación de los Comentarios de Documentación

Se pueden insertar comentarios de documentación para clases, interfaces, métodos, atributos y constructores.

- Los comentarios de la clase o interfaz se colocan en la parte superior del archivo, después de las sentencias **import** e inmediatamente antes de la declaración de la clase o interfaz.
- Los comentarios de los atributos se colocan inmediatamente antes de su declaración.
- Los comentarios de los métodos y constructores se colocan inmediatamente antes de la declaración de su firma.

Por ejemplo, el siguiente listado muestra una clase con comentarios de documentación:

### Medio.java

```
/*
 * Medio.java
 *
 * Creada el 8 de septiembre de 2006, 01:14 PM
 */
package objetosNegocio;
```

```
import objetosServicio.Fecha;

/**
 * Esta clase contiene los atributos y métodos comunes a las clases
 * Cancion y Pelicula del programa AmanteMusica
 *
 * @author mdomitsu
 */
public class Medio {
    protected String clave;
    protected String titulo;
    protected Genero genero;
    protected int duracion;
    protected Fecha fecha;

    /**
     * Constructor predeterminado
     */
    public Medio() {
    }

    /**
     * Constructor que inicializa los atributos de la clase
     * @param clave Clave de la canción o película
     * @param titulo Título de la canción o película
     * @param genero Género de la canción o película
     * @param duracion Duración de la canción o película
     * @param fecha Fecha de liberación de la canción o película
     */
    public Medio(String clave, String titulo, Genero genero, int duracion,
                 Fecha fecha) {
        this.clave = clave;
        this.titulo = titulo;
        this.genero = genero;
        this.duracion = duracion;
        this.fecha = fecha;
    }

    /**
     * Regresa la clave de la canción o película
     * @return Clave de la canción o película
     */
    public String getClave() {
        return clave;
    }

    /**
     * Establece la clave de la canción o película
     * @param clave Clave de la canción o película
     */
    public void setClave(String clave) {
        this.clave = clave;
    }

    /**
     * Regresa el título de la canción o película

```

```
* @return Titulo de la canción o película
*/
public String getTitulo() {
    return titulo;
}

/**
 * Establece el título de la canción o película
 * @param titulo Titulo de la canción o película
 */
public void setTitulo(String titulo) {
    this.titulo = titulo;
}

/**
 * Regresa el género de la canción o película
 * @return Género de la canción o película
 */
public Genero getGenero() {
    return genero;
}

/**
 * Establece el género de la canción o película
 * @param genero Género de la canción o película
 */
public void setGenero(Genero genero) {
    this.genero = genero;
}

/**
 * Regresa la duración de la canción o película
 * @return Duración de la canción o película
 */
public int getDuracion() {
    return duracion;
}

/**
 * Establece la duración de la canción o película
 * @param duracion Duración de la canción o película
 */
public void setDuracion(int duracion) {
    this.duracion = duracion;
}

/**
 * Regresa la fecha de liberación de la canción o película
 * @return Fecha de liberación de la canción o película
 */
public Fecha getFecha() {
    return fecha;
}

/**
 * Establece la fecha de liberación de la canción o película
 * @param fecha Fecha de liberación de la canción o película
 */
```

```

*/
public void setFecha(Fecha fecha) {
    this.fecha = fecha;
}

/**
 * Regresa una cadena con los valores de los atributos de una instancia de
 * la clase
 * @return Cadena con los valores de los atributos
 */
public String toString() {
    return clave + ", " + titulo + ", " + genero.getNombre() + ", " +
        duracion + ", " + fecha;
}
}
    
```

## Etiquetas Javadoc

En la tabla 2.1 se muestran las etiquetas de documentación.

**Tabla 2.1**

Etiqueta	Descripción	Tipo de Etiqueta
<b>@author</b> <i>nombre</i>	Agrega una entrada de autor con el nombre especificado en el documento. Normalmente no se incluye en la documentación.	Resumen, paquete, clase, interfaz
<b>@versión</b> <i>versión_no</i>	Agrega un subencabezado con el número de versión especificado. Normalmente no se incluye en la documentación.	Resumen, paquete, clase, interfaz
<b>@param</b> <i>nomPar descripción</i>	Agrega un parámetro al subencabezado Parámetros.	Constructor, método
<b>@return</b> <i>descripción</i>	Agrega un subencabezado Return con el texto de la descripción.	Constructor, método
<b>@deprecated</b> <i>texto</i>	Agrega un comentario indicando que la API ha sido descontinuada. Normalmente no se incluye en la documentación.	Paquete, clase, interfaz, atributo, constructor, método
<b>@exception</b> <i>nomClase descripción</i>	Agrega un subencabezado Throws al documento con el nombre de la clase de la excepción que puede lanzar.	Constructor, método
<b>@throws</b> <i>nomClase descripción</i>	Sinónimo de <b>@exception</b>	Constructor, método
<b>{@value}</b>	Se utiliza con atributos estáticos. Despliega el valor de la constante en la página de atributos constantes.	Atributos

## Edición de los Comentarios de Documentación

NetBeans nos ayuda a editar los comentarios de documentación de los diferentes elementos de una clase: clase, atributos, métodos. Para agregarle los comentarios de documentación a una clase se sigue el siguiente procedimiento:

1. Coloque el cursor en la línea anterior a la línea con el nombre del atributo o método al que desea agregarle un comentario de documentación y teclee los caracteres `/**`, figura 2.

```

public class Genero {

    private String cveGenero;
    private String nombre;
    private char tipoMedio;

    /**
     * Constructor vacio
     */
    public Genero() {
    }

    /**
    public Genero(String cveGenero, String nombre, char tipoMedio) {
        this.cveGenero = cveGenero;
        this.nombre = nombre;
        this.tipoMedio = tipoMedio;
    }

    public Genero(String cveGenero) {
        this(cveGenero, null, ' ');
    }

    public String getCveGenero() {
        return cveGenero;
    }
}

```

**Figura 2**

2. Al presionar la tecla **Intro**, veremos que NetBeans ha agregado un esqueleto para la documentación del elemento, figura 3. Note que en este caso le ha agregado una etiqueta Javadoc por cada parámetro del constructor. NetBeans hará algo similar si el método regresa un valor o si lanza una excepción.
3. En la segunda línea del comentario (la resaltada en azul) agregue la descripción de lo que hace el método.
4. Al lado derecho del nombre del parámetro de cada etiqueta `@param` describa el propósito del parámetro.
5. Al lado derecho de cada etiqueta `@return` describa qué regresa el método.
6. Al lado derecho del nombre de la excepción de cada etiqueta `@throws` describa por qué razón se lanza la excepción.

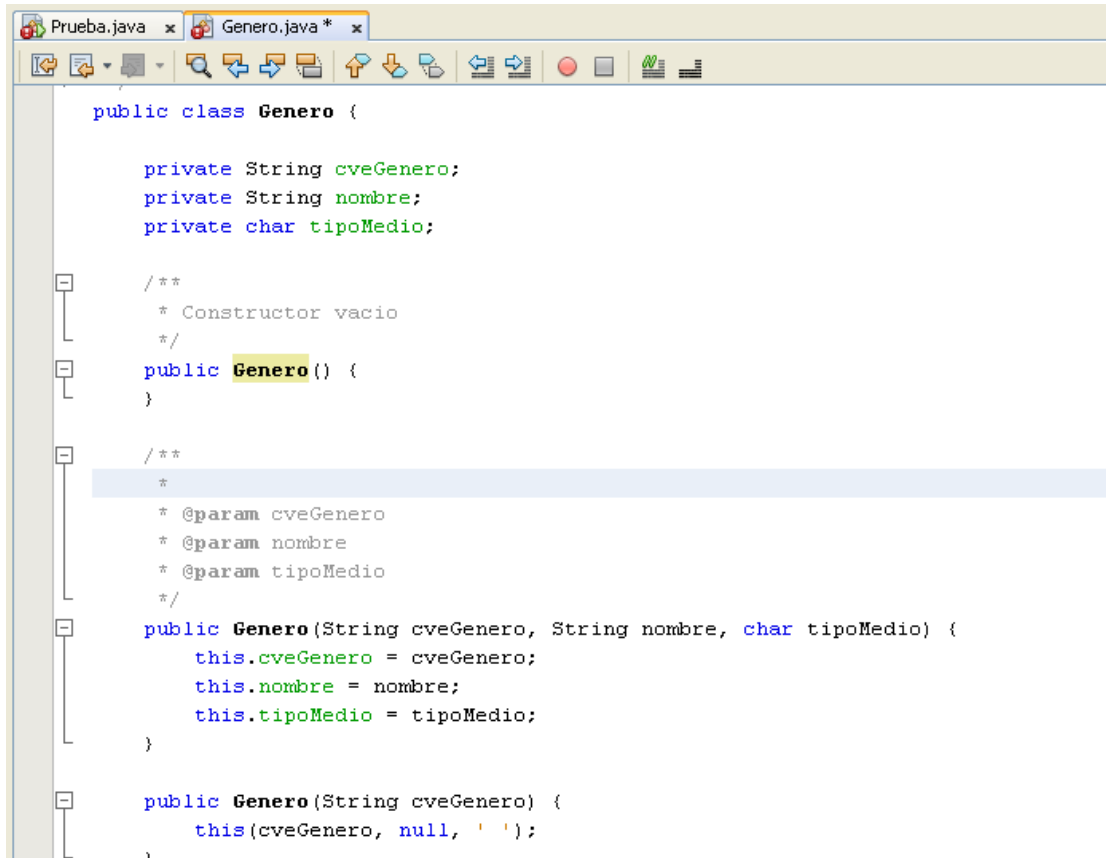


Figura 3

7. Si hay atributos no privados agregue un comentario de documentación, describiendo su uso.

## Generación de la Documentación Javadoc

NetBeans puede producir un conjunto de páginas HTML Javadoc que describe las clases, constructores, métodos y atributos, a partir de la estructura del código y los comentarios de documentación incluido en el código.

Para generar la documentación Javadoc seleccione de la barra de menú de NetBeans la opción **Run/ Generate Javadoc (nomProyecto)** donde nomProyecto es el nombre de proyecto, por ejemplo amanteMusicaObjNeg, figura 4

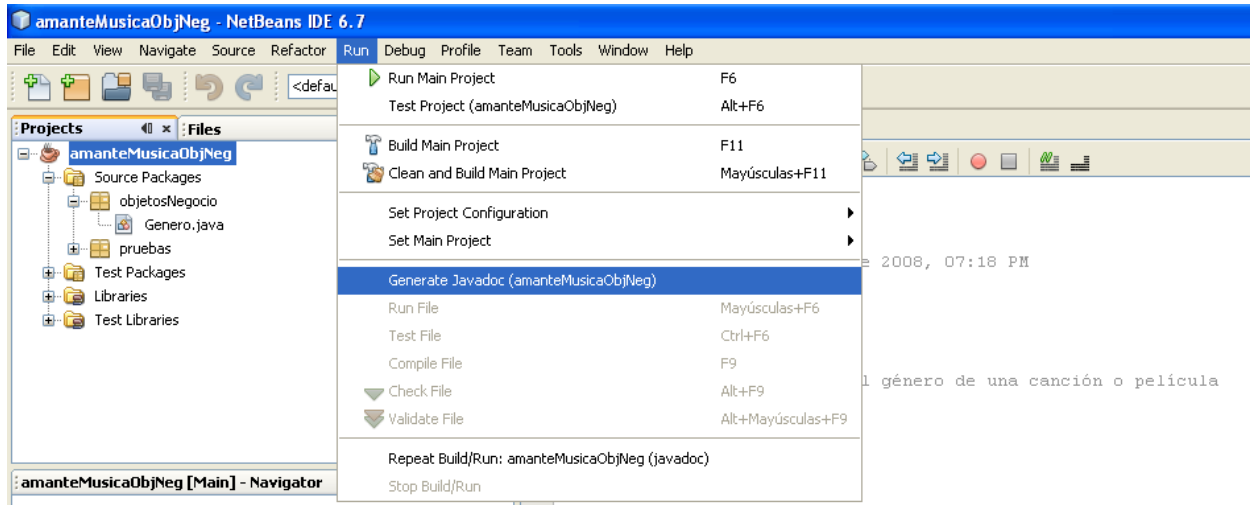


Figura 4

Otra forma de generar la documentación Javadoc, es hacer clic con el botón derecho en el nombre del proyecto en el árbol de proyectos y seleccione la opción **Generate** del menú emergente, figura 5.

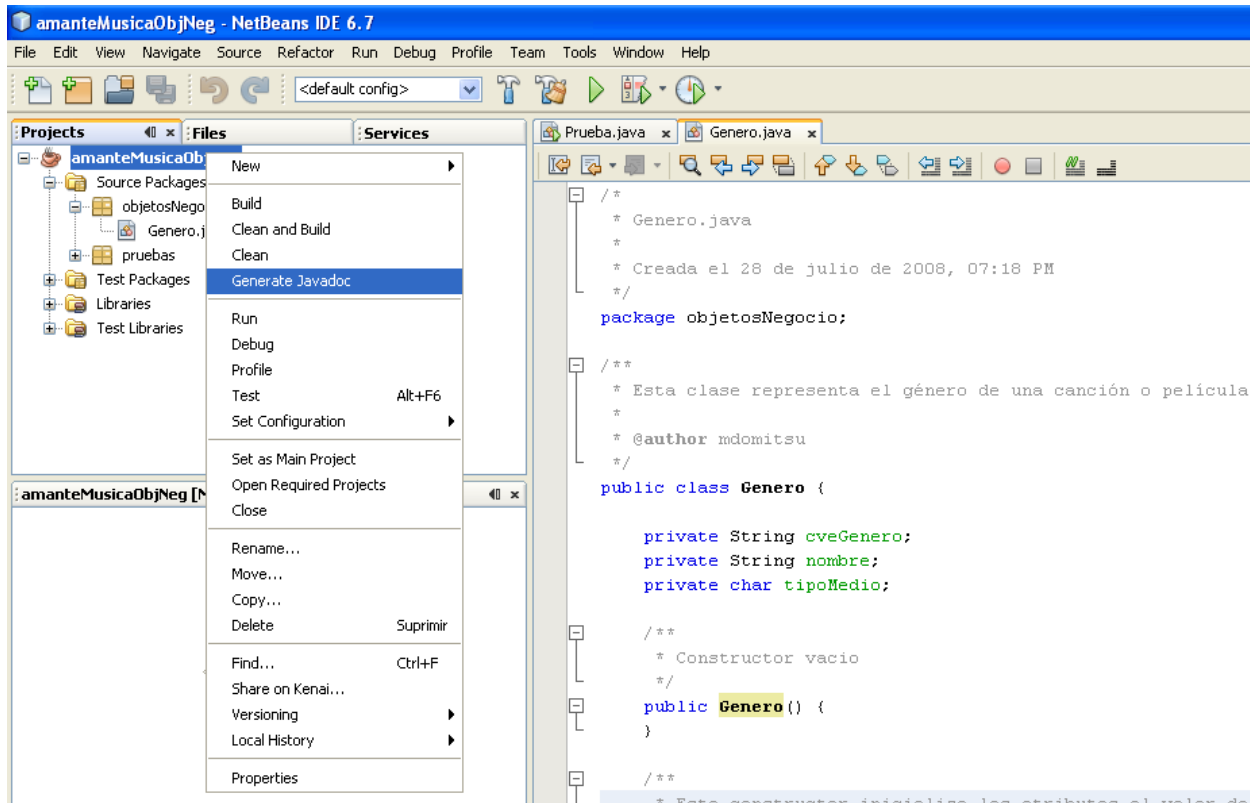


Figura 5

En ambos casos, NetBeans genera las páginas HTM Javadoc, mostrando el avance en la ventana de salida, figura 6.



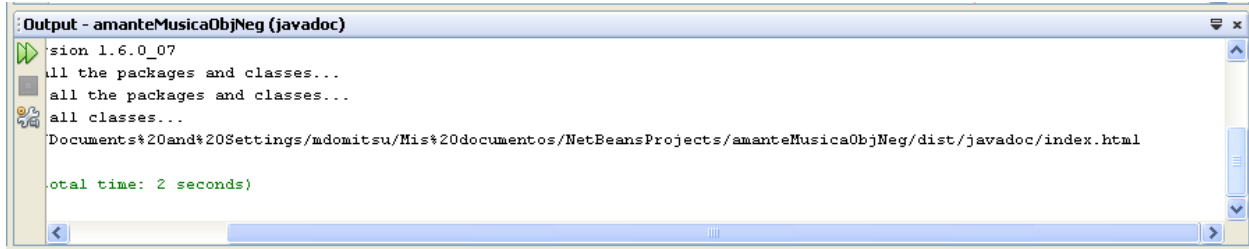


Figura 6

Al terminar, Netbeans abre el navegador Web predeterminado y despliega en él la página inicial de la documentación del proyecto como se muestra en la figura 7.

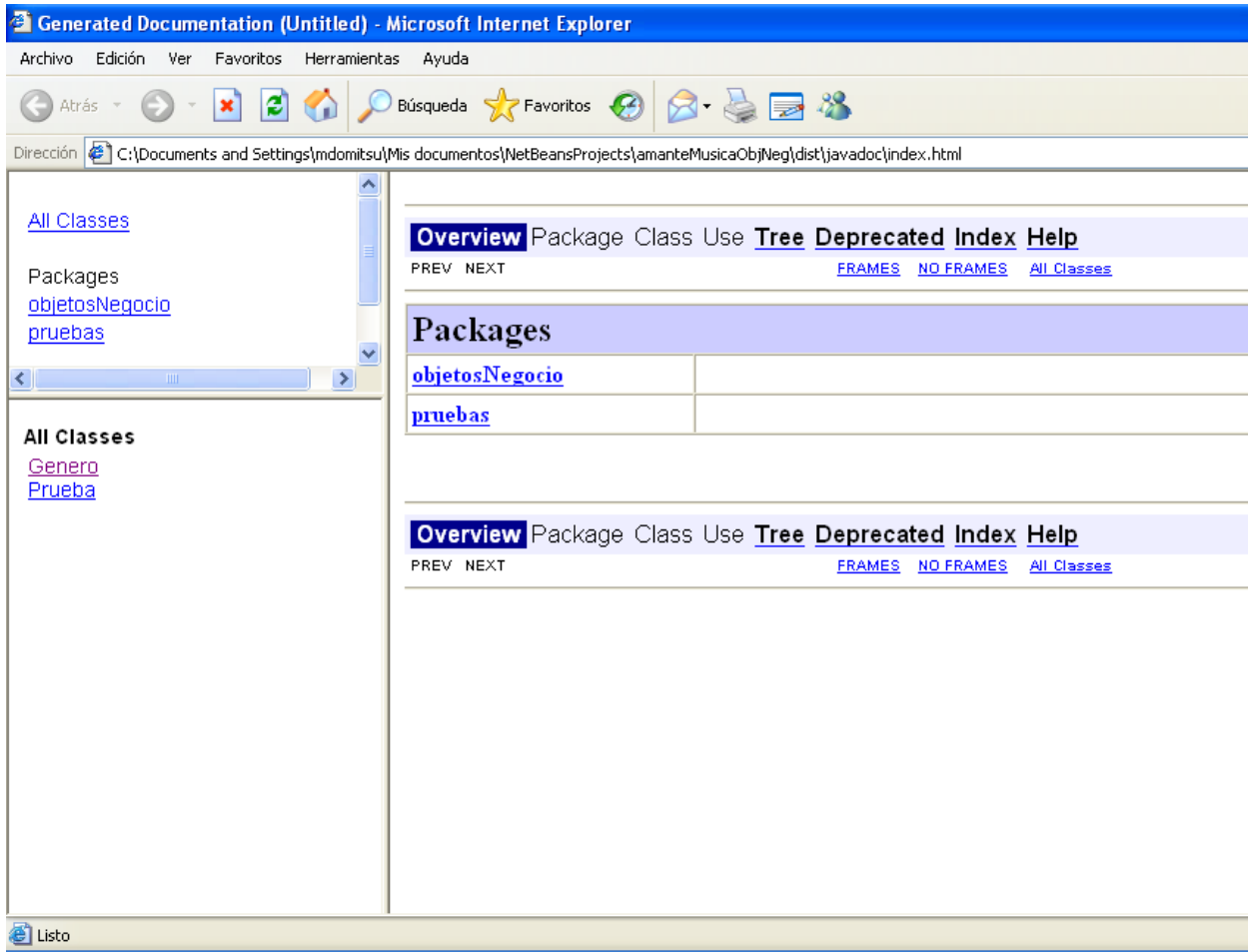


Figura 7

Haciendo clic en las ligas podemos ver la documentación de las diferentes clases del proyecto. Por ejemplo, en la figura 8 se muestra la documentación de la clase Genero.

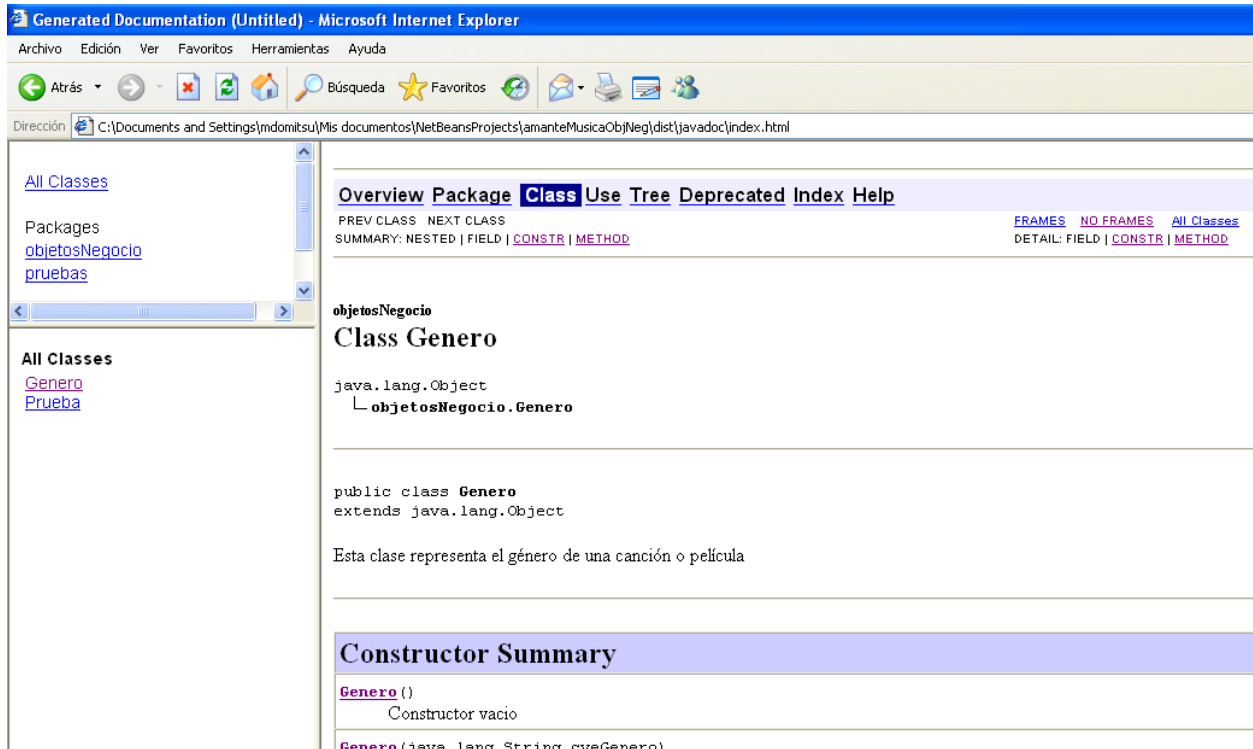


Figura 8

## Configuración de Javadoc

Podemos configurar como NetBeans generará la documentación Javadoc para el proyecto

1. Haga clic con el botón derecho en el nombre del proyecto, en el árbol de proyectos y seleccione la opción **Properties** en el menú emergente, figura 9.
2. Aparecerá un cuadro de diálogo para establecer las propiedades del proyecto. Seleccione el nodo **Documenting** en el árbol **Categories**, figura 10, y seleccione las opciones deseadas. y haga clic en el botón **OK**.

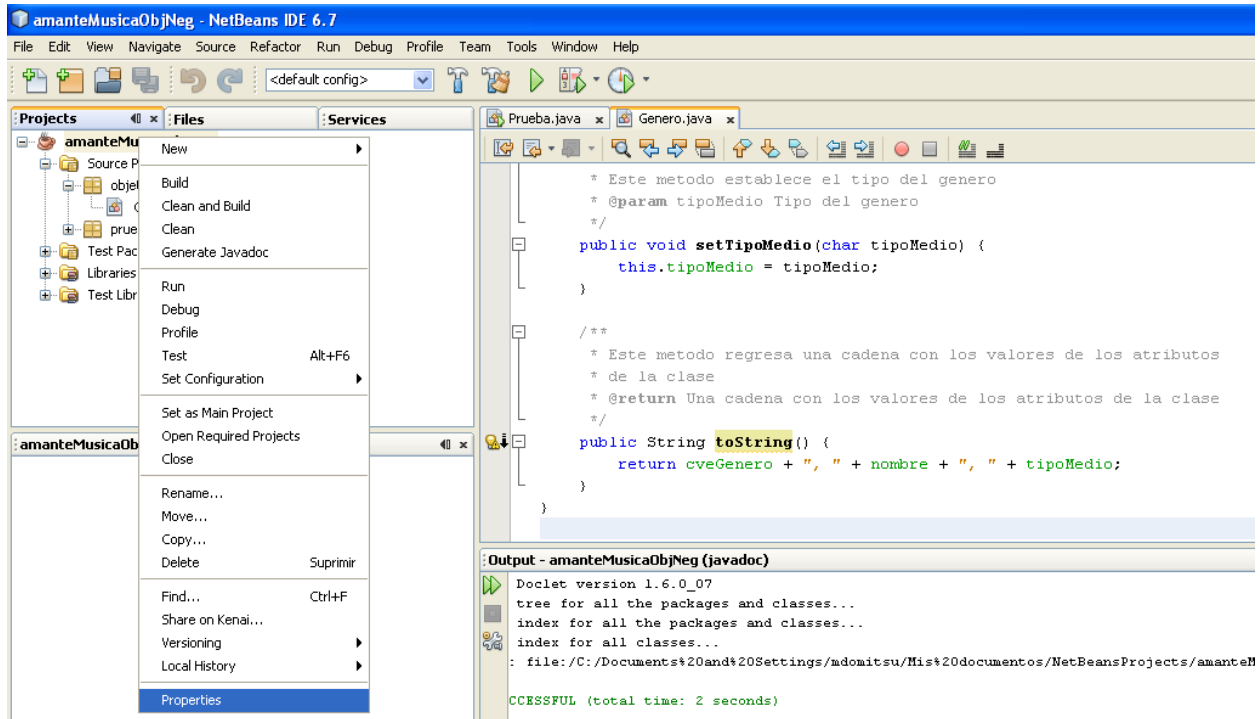


Figura 9

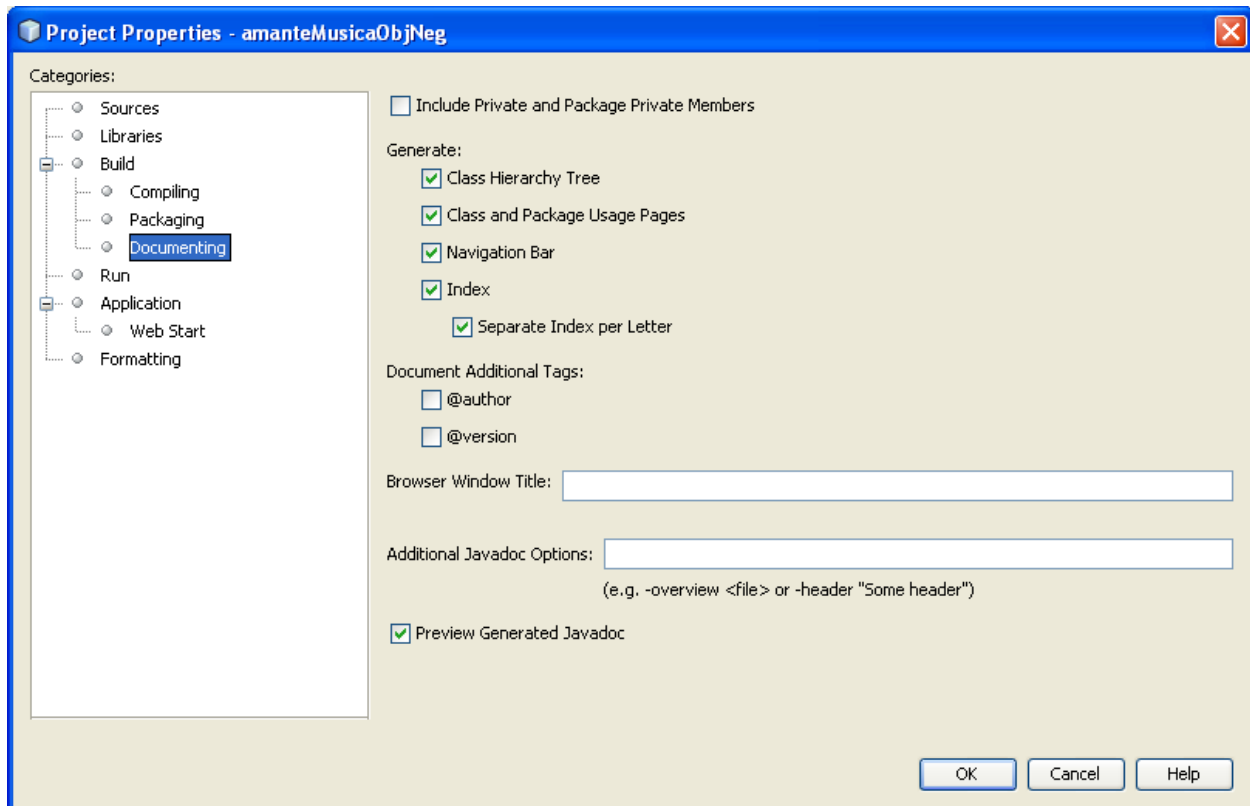


Figura 10

## Ubicación de los Archivos con la Documentación

La figura 11, muestra los archivos HTML Javadoc generados. Se encuentran dentro de la carpeta “**dist/javadoc**” que a su vez está dentro de la carpeta del proyecto.

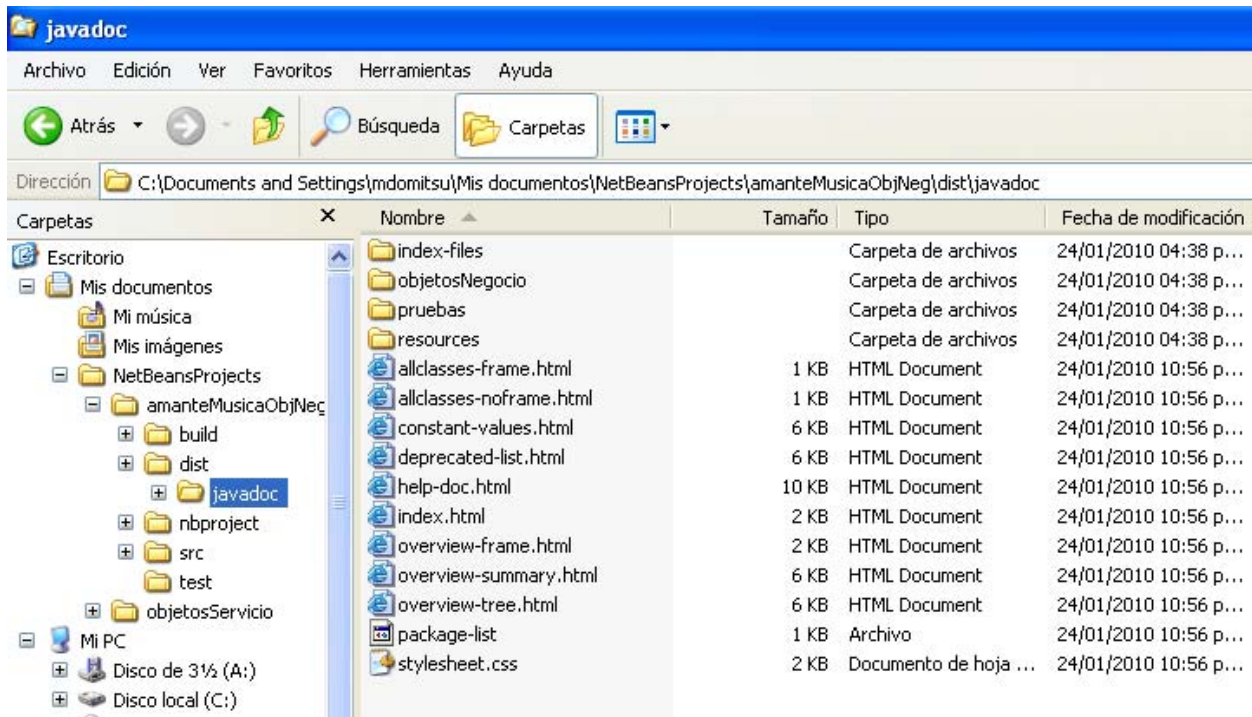


Figura 11