

# Tutorial 4

## Aplicaciones Web con NetBeans 5.0

### Creación de un Proyecto

Para crear un programa de consola en Java utilizando **NetBeans 5.0** lo primero que hay que hacer es crear un proyecto. Un proyecto nos permite administrar los archivos con el código fuente y compilado de una aplicación. Para crear un proyecto se sigue el siguiente procedimiento:

1. Ejecute el programa **NetBeans 5.0**. Al hacerlo aparecerá la ventana principal del programa como se ilustra en la figura 4.1.

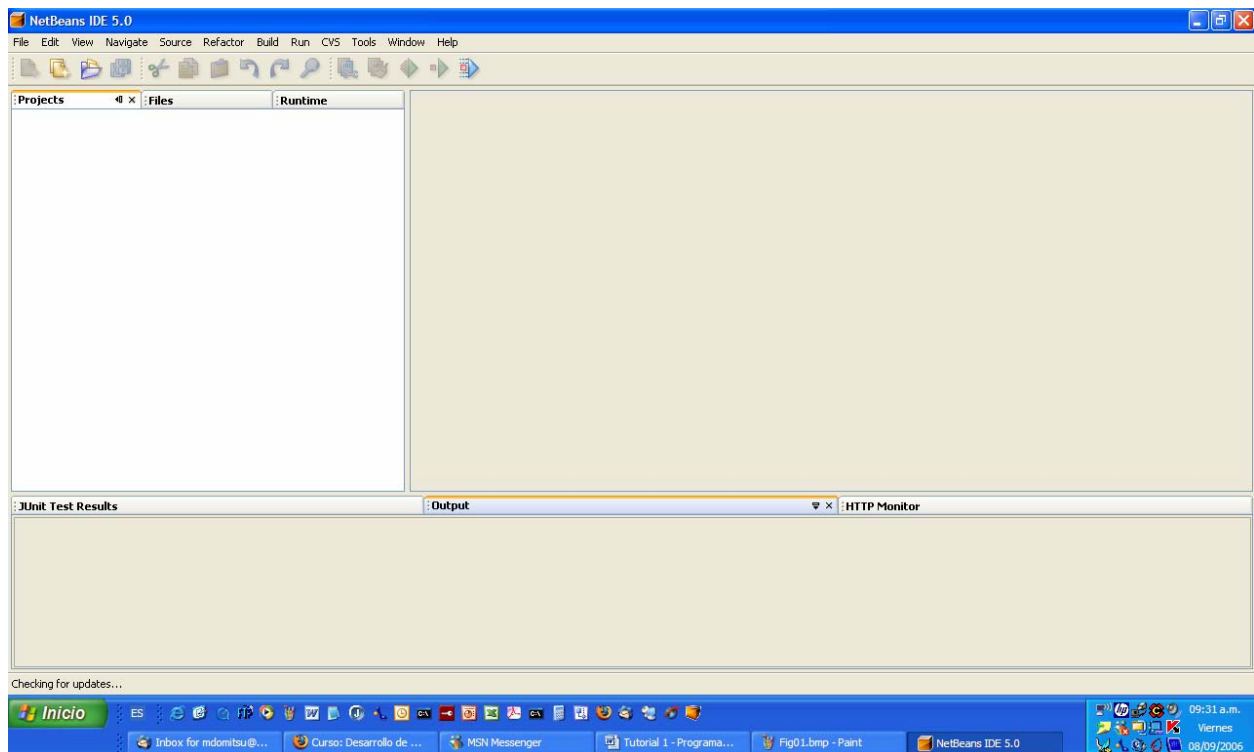


Figura 4.1

2. Del menú principal de NetBeans 5.0, figura 1.1, seleccione la opción **File/New Project ...**, presione las teclas **Ctrl+Mayúsculas+N** o haga clic en el icono **New Project** mostrado en la figura 4.2.

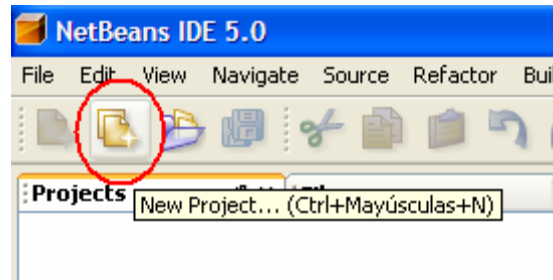


Figura 4.2

3. Aparecerá el primer cuadro del asistente para crear un nuevo proyecto, figura 4.3.

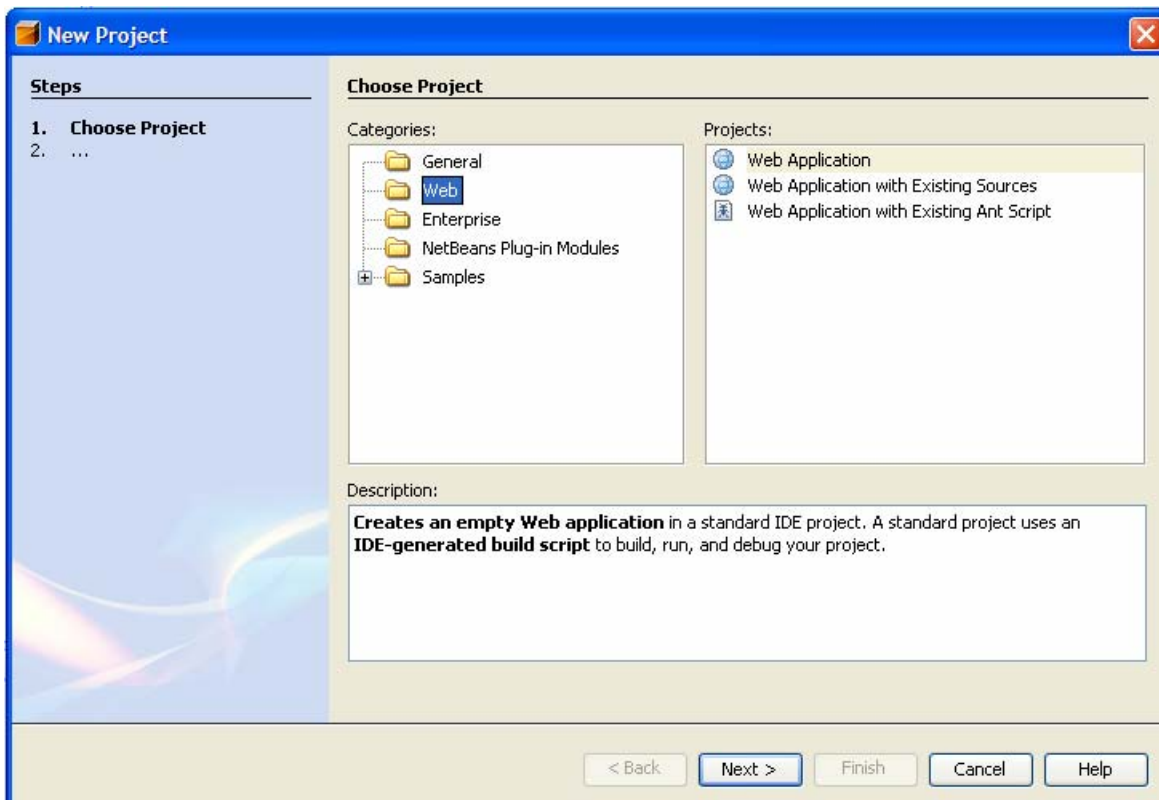


Figura 4.3

4. En este cuadro del asistente seleccionaremos el tipo de proyecto que deseamos crear. Como vamos a crear una aplicación Web, seleccionaremos la opción **Web** en el recuadro **Categories:** y la opción **Web Application** en el recuadro **Projects:**, y luego presionaremos el botón **Next>**.
5. Aparecerá el segundo del asistente para crear proyectos, mostrada en la figura 4.4. En este cuadro seleccionaremos el nombre y la ubicación del proyecto.

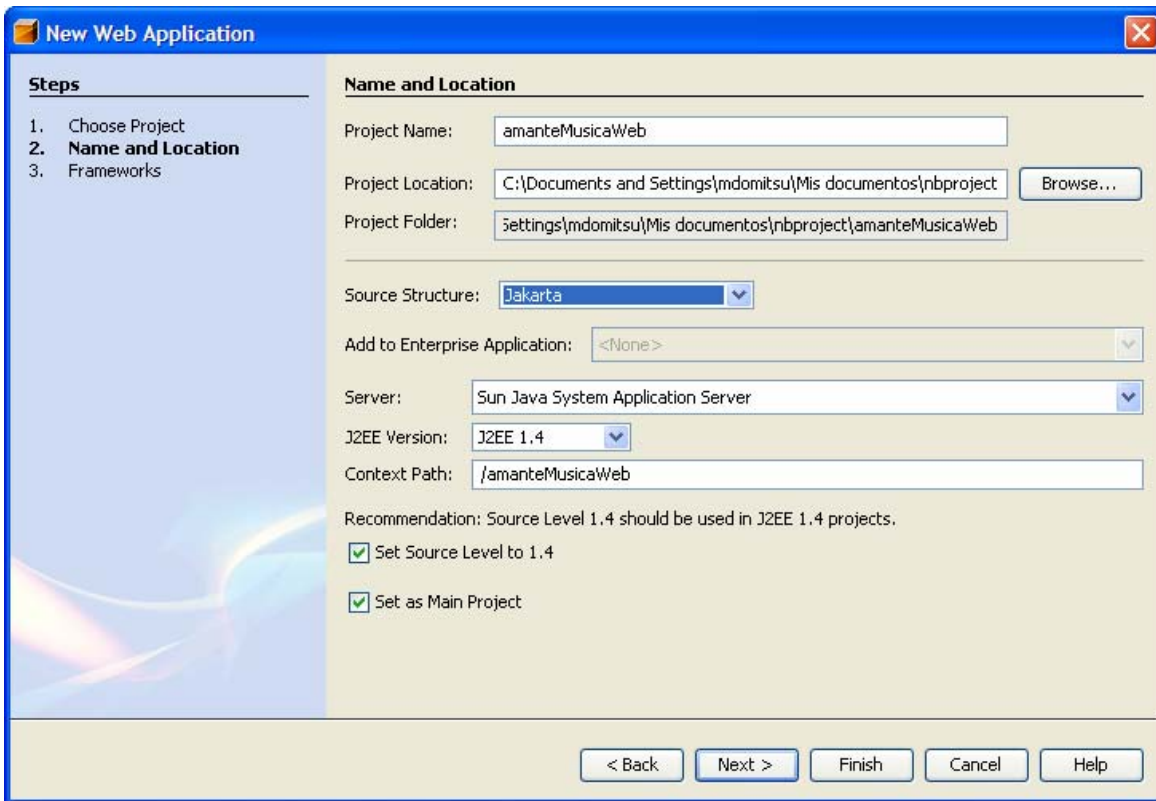
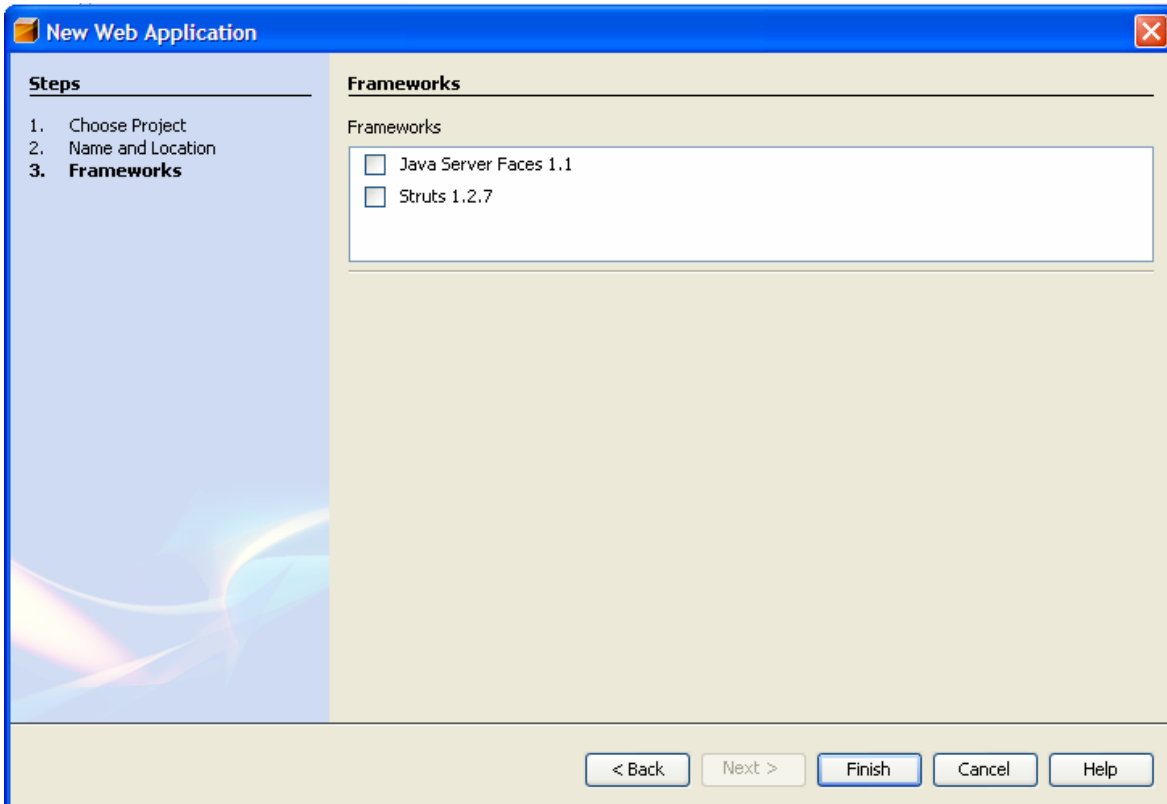


Figura 4.4

- a) Establezca el nombre del proyecto (**Project Name**): Por ejemplo, **“amanteMusicaWeb”**.
  - b) Establezca el directorio donde se almacenará el proyecto (**Project Location**). Por ausencia en Windows 2000 y XP, el directorio es el directorio inicial del usuario: **“C:\Documents and Settings\usuario”**. En este ejemplo, el proyecto se ubicó en: **“C:\Documents and Settings\mdomitsu\Mis documentos\nbproject”**. En la línea siguiente puede verse la ubicación del directorio en el que se almacenarán los archivos del proyecto: **Project Location**, que es el directorio con el nombre del proyecto dentro del directorio donde se ubica el proyecto. En este ejemplo es: **“C:\Documents and Settings\mdomitsu\Mis documentos\nbproject\amanteMusicaWeb”**.
  - c) Seleccione en la caja combinada: **Source Structure**: (estructura del código fuente:) la opción **Jakarta**.
  - d) Seleccione en la caja combinada: **Server**: (Servidor:) la opción **Sun Java System Application Server**.
  - e) Seleccione en la caja combinada: **J2EE Version**: la opción **J2EE 1.4**.
  - f) Asegúrese que las casillas de selección: **Set Source Level to 1.4** (Establezca el nivel del código fuente a la versión 1.4) y **Set as Main Project** (Haga que este proyecto sea el proyecto principal) estén seleccionadas.
  - g) Presione el botón **Next>**.
6. Aparecerá el tercer cuadro del asistente para crear proyectos, figura 4.5. Este cuadro nos permite establecer si nuestra aplicación Web usará el marco de Java

Server Faces 1.1 o Struts 1.2.7. Como no se utilizará ninguno, presione el botón **Finish**.



**Figura 4.5**

7. Desaparecerá el asistente para crear un nuevo proyecto y aparecerá lo mostrado en la figura 4.6. Del lado derecho aparece el editor de NetBeans con el esqueleto de la página JSP principal: `index.jsp`, mientras que del lado izquierdo aparece el árbol de los proyectos, en el que aparece el proyecto **amanteMusicaWeb**.
8. Si en el recuadro del árbol de los proyectos hacemos clic en la pestaña Files, aparecerá un árbol con todos los archivos de los proyectos, figura 4.7. Otra vez, en este momento sólo aparecen los archivos del proyecto **amanteMusicaWeb**.

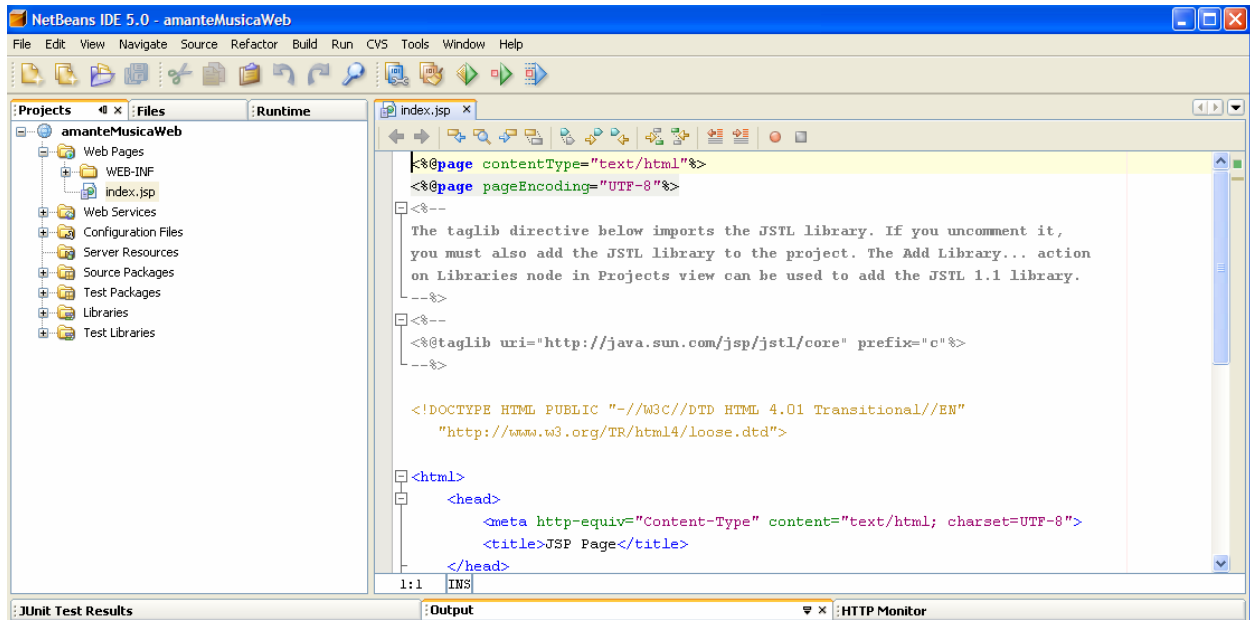


Figura 4.6

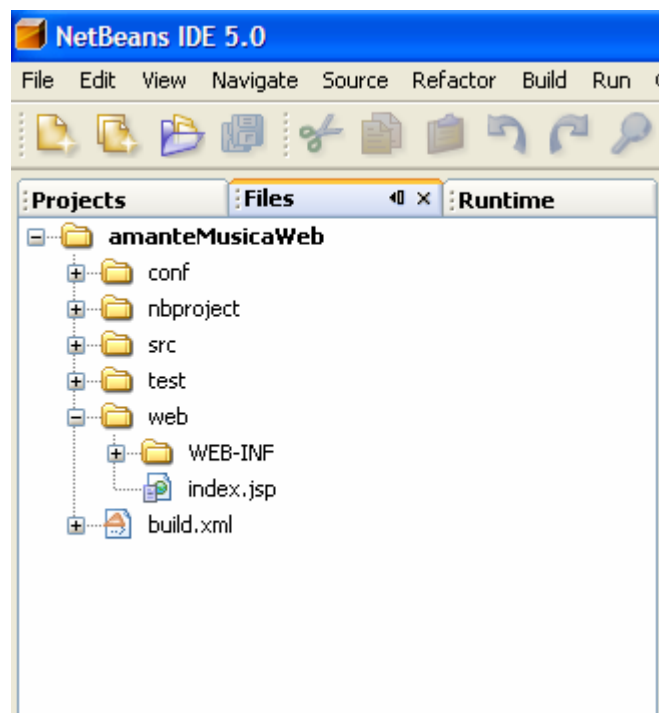


Figura 4.7

9. Edite el archivo `index.jsp` para que su código sea el siguiente:

### `index.jsp`

```

<%@page contentType="text/html"%>
<%@page pageEncoding="UTF-8"%>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"

```

```
"http://www.w3.org/TR/html4/loose.dtd">

<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Amante Música</title>
  </head>

  <body>
    <h1 align="center">Amante Música</h1>

    <div align="center">
      Selecciona la tarea deseada

      <!-- Formulario para seleccionar y enviar la tarea deseada.
           La tarea seleccionada se envía a la página control.jsp -->
      <br>
      <br>

      <form action="control.jsp" method="post">

        <!-- Menú desplegable -->
        <select name="tarea">
          <option value="tarea1" selected>Agregar una canción</option>
          <option value="tarea2">Eliminar una canción</option>
          <option value="tarea3">Listar todas las canciones</option>
        </select>

        <!-- Botón enviar -->
        <input type="submit" name="enviar" value="Continuar">
      </form>
    </div>
  </body>
</html>
```

10. Guarde la página JSP seleccionando del menú principal la opción **File/Save**, presione las teclas **Ctrl+S** o haga clic en el icono **Save All**, mostrado en la figura 4.8.

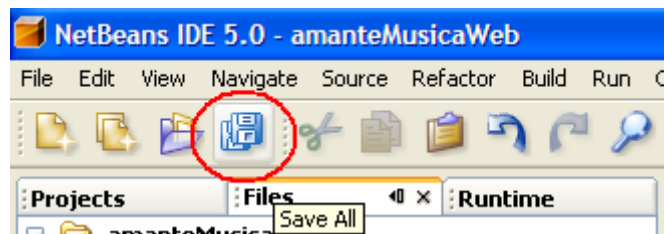


Figura 4.8

# Ubicación de los Archivos de una Aplicación Web

La figura 4.9, muestra los directorios y archivos generados al crear el proyecto. El proyecto se guarda en una carpeta con el nombre del proyecto, **amanteMusicaWeb** en este caso. Las páginas JSP se almacenan en un directorio llamado **web**, dentro del directorio del proyecto.

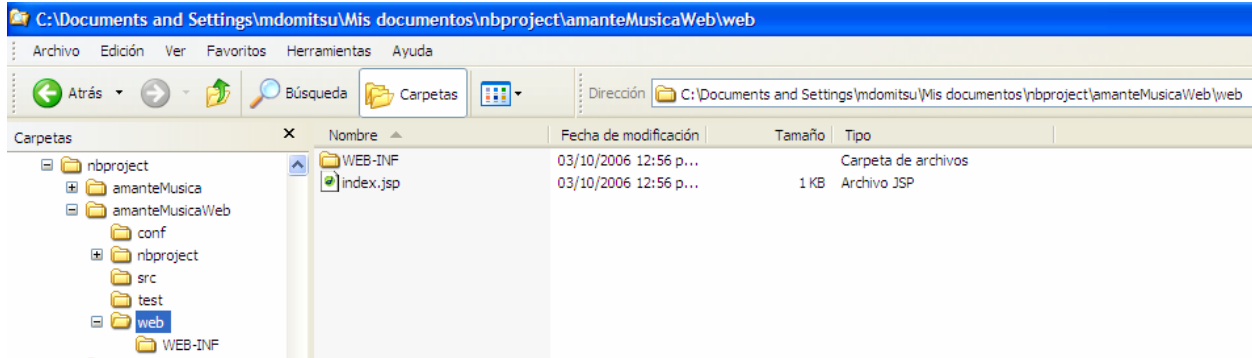


Figura 4.9

Por otro lado si se crean clases, el directorio raíz de su estructura de paquetes será directorio **scr**.

## Creación de una Página JSP

Para crear una página JSP se sigue el siguiente procedimiento:

1. Del menú principal de NetBeans 5.0, figura 4.1, seleccione la opción **Files/New File**, presione las teclas **Ctrl+ N** o haga clic en el icono **New File**, como se muestra en la figura 4.10:

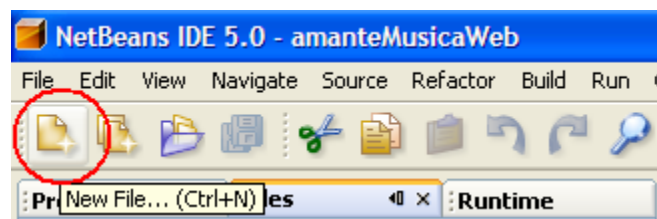
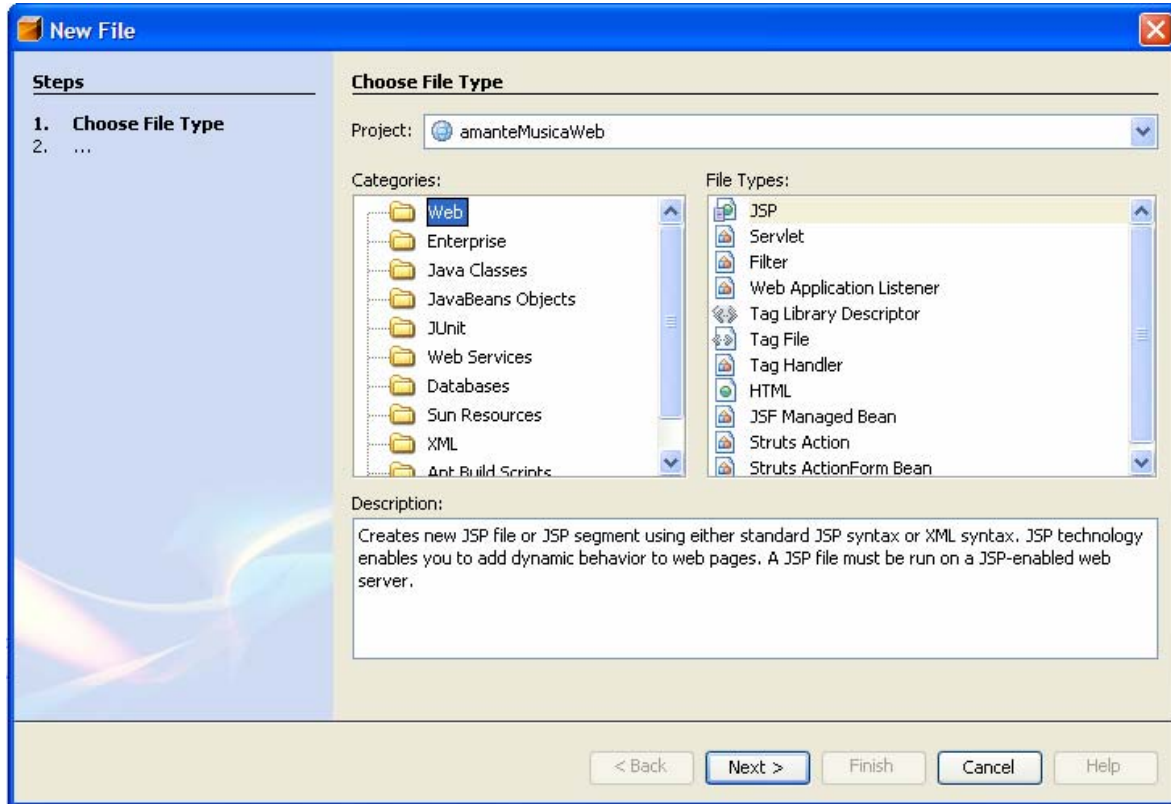


Figura 4.10

2. Aparecerá la primera ventana del asistente para crear un nuevo archivo, figura 4.11. En esta ventana del asistente seleccionaremos el tipo de archivo que deseamos crear. Seleccionaremos la opción **Web** en el recuadro **Categories:** y

la opción **JSP** en el recuadro **File Types:**, y luego presionaremos el botón **Next>**.



**Figura 4.11**

3. Aparecerá la segunda ventana del asistente para crear un archivo, mostrada en la figura 4.12. En esta ventana seleccionaremos el nombre y la ubicación de la página JSP.
  - a) Establezca el nombre de la clase (**JSP File Name**): Por ejemplo, **“control”**.
  - b) Deje el resto de los campos a sus valores preestablecidos.
  - c) Presione el botón **Finish**.
4. Desaparecerá el asistente para crear un nuevo archivo y aparecerá el esqueleto de la página JSP creada.
5. Edite el archivo control.jsp para que su código sea el mostrado en la página siguiente:
6. Guarde la página JSP.



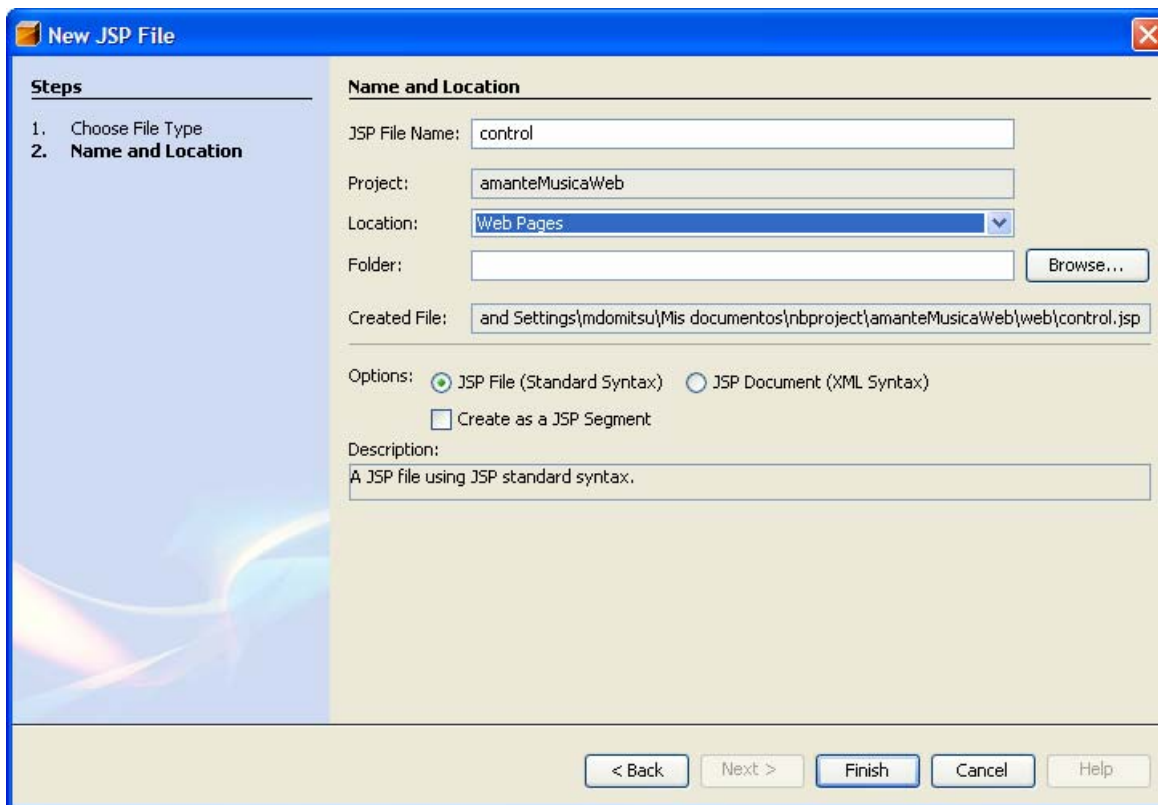


Figura 4.12

## control.jsp

```

<%@page contentType="text/html"%>
<%@page pageEncoding="UTF-8"%>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">

<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  </head>

  <body>
<%
  // Obten el número de la tarea seleccionada en index.jsp
  int opcion =
    Integer.parseInt(request.getParameter("tarea").substring(5));

  switch(opcion) {
    // Si la tarea seleccionada es la 1 o 2
    case 1:
    case 2:
%>
    <%-- Carga la página capturaClave.jsp para capturar la
         clave de la canción a agregar o borrar --%>
    <jsp:forward page="capturaClave.jsp"/>
<%

```

```

// Si la tarea seleccionada es la 3
case 3:
%>
    <%-- Carga la página consultaCancionesTodas.jsp para
        desplegar todas las canciones --%>
    <jsp:forward page="consultaCancionesTodas.jsp" />
<%
}
%>
</body>
</html>

```

## Edición de Múltiples Páginas JSP

Podemos editar más de una clase o página JSP en la ventana de edición, cada una en su propio panel, como se muestra en la figura 4.13 en la que tenemos en la ventana de edición de NetBeans dos páginas JSP: `index.jsp` y `control.jsp`. Las pestañas en la parte superior nos permiten seleccionar la clase o página JSP que se desea editar.

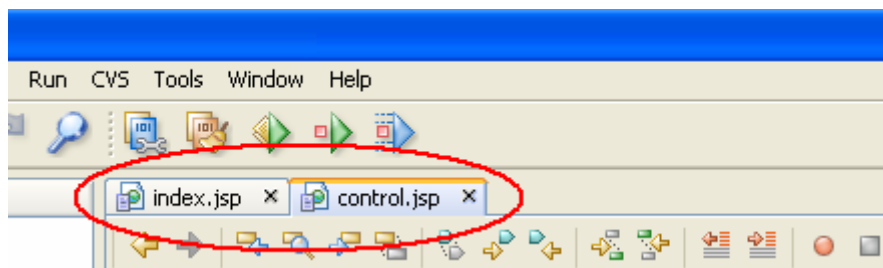


Figura 4.13

Podemos eliminar una clase o página JSP de la ventana de edición, haciendo clic en el icono con la X que se encuentra en la pestaña de cada clase, figura 4.14.

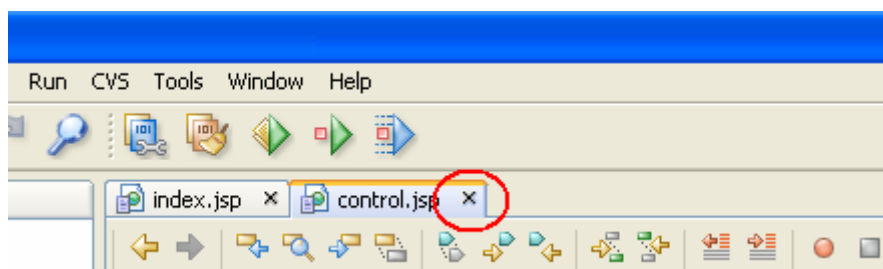


Figura 4.14

Si deseamos agregar una clase o página JSP ya creada a la ventana de edición podemos hacer clic en su nombre en el árbol de archivos o en el árbol de proyectos que se encuentran a la izquierda, como se muestra en la figura 4.15.

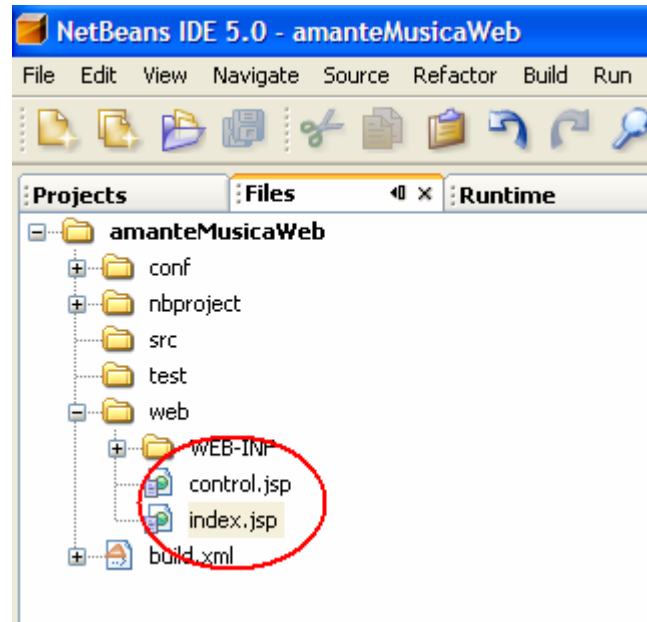


Figura 4.15

## Compilación de una Página JSP

Para compilar la clase que se encuentra en la ventana de edición seleccione del menú principal la opción **Build/Compile "NombreJSP.jsp"**. "**NombreJSP**" es el nombre de la página JSP a compilar, por ejemplo "**index**", figura 4.16. También puede presionar la tecla **F9**.

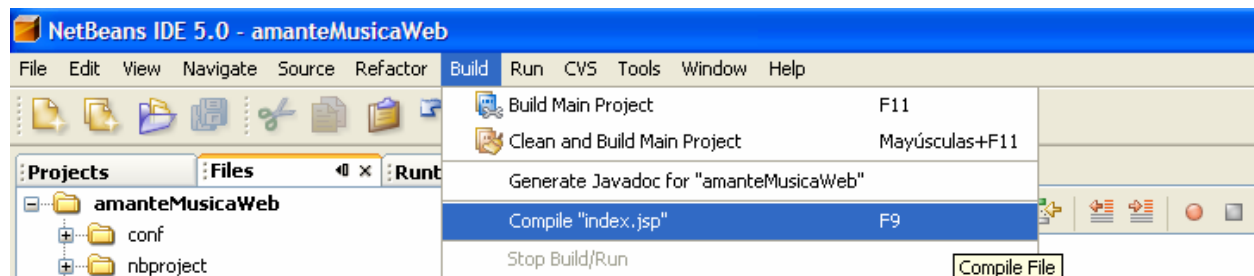


Figura 4.16

## Compilación del Proyecto

Para compilar todas las clases y páginas JSP de un proyecto, seleccione del menú principal la opción **Build/Build Main Project**, figura 4.17, presione la tecla **F11** o presione el icono **Build Main Project**, mostrado en la figura 4.18.

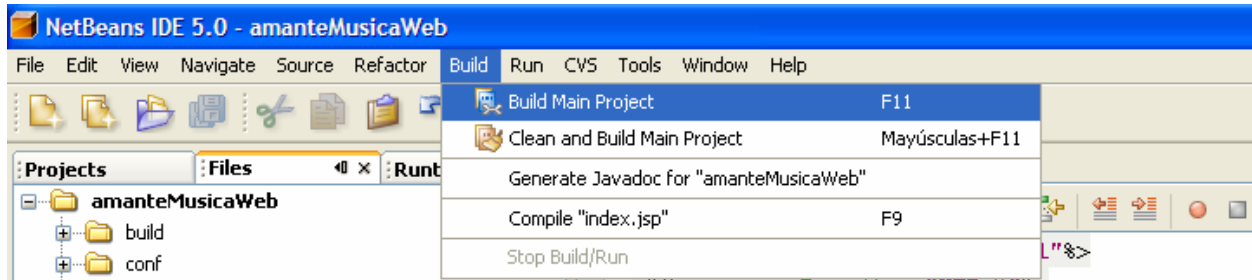


Figura 4.17

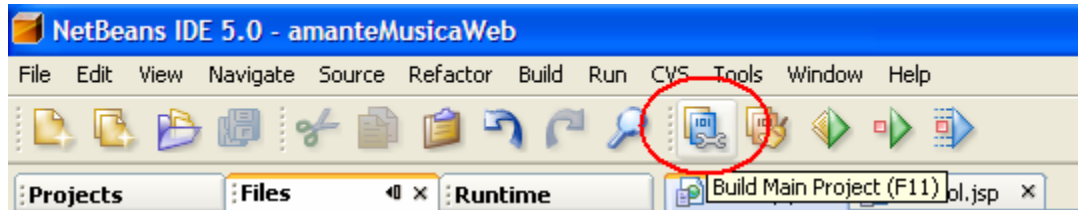


Figura 4.18

Durante la compilación, NetBeans muestra los mensajes resultantes del proceso, como se muestra en la figura 4.19.

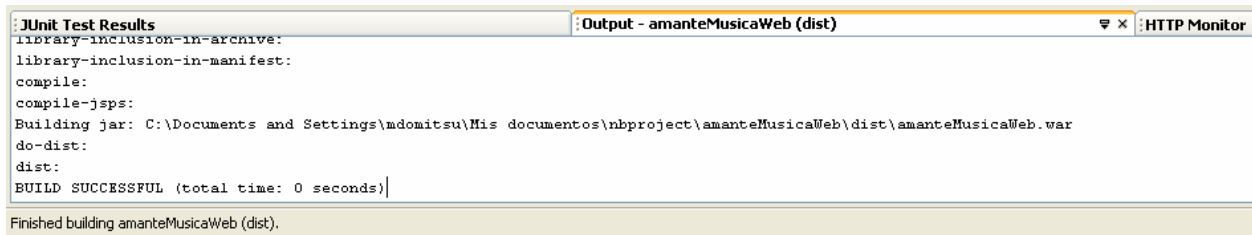


Figura 4.19

## Ubicación del Archivo de despliegue de la Aplicación Web

Al compilar todo el proyecto de una aplicación Web, NetBeans empaqueta los archivos con el código “byteCode” de las clases, las páginas JSP, los archivos con los recursos y los archivos de configuración en un archivo WAR, con el nombre del proyecto y la extensión “.war” y lo almacena en el directorio “dist” dentro del directorio del proyecto, figura 4.20.

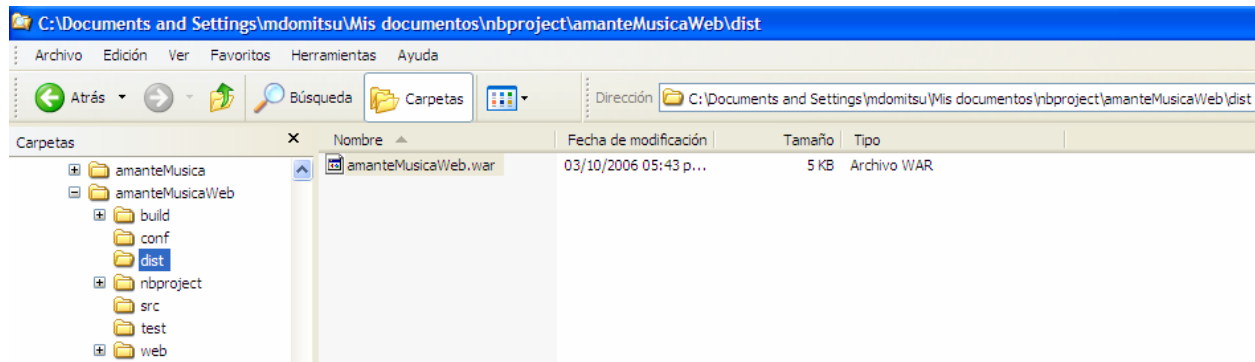


Figura 4.20

## Ejecución de una Aplicación Web

1. Para ejecutar la aplicación dentro de NetBeans, seleccione del menú principal la opción **Run/Run Main Project**, figura 4.21, presione la tecla **F6** o haga clic en el icono **Run Main Project**, mostrado en la figura 4.22.

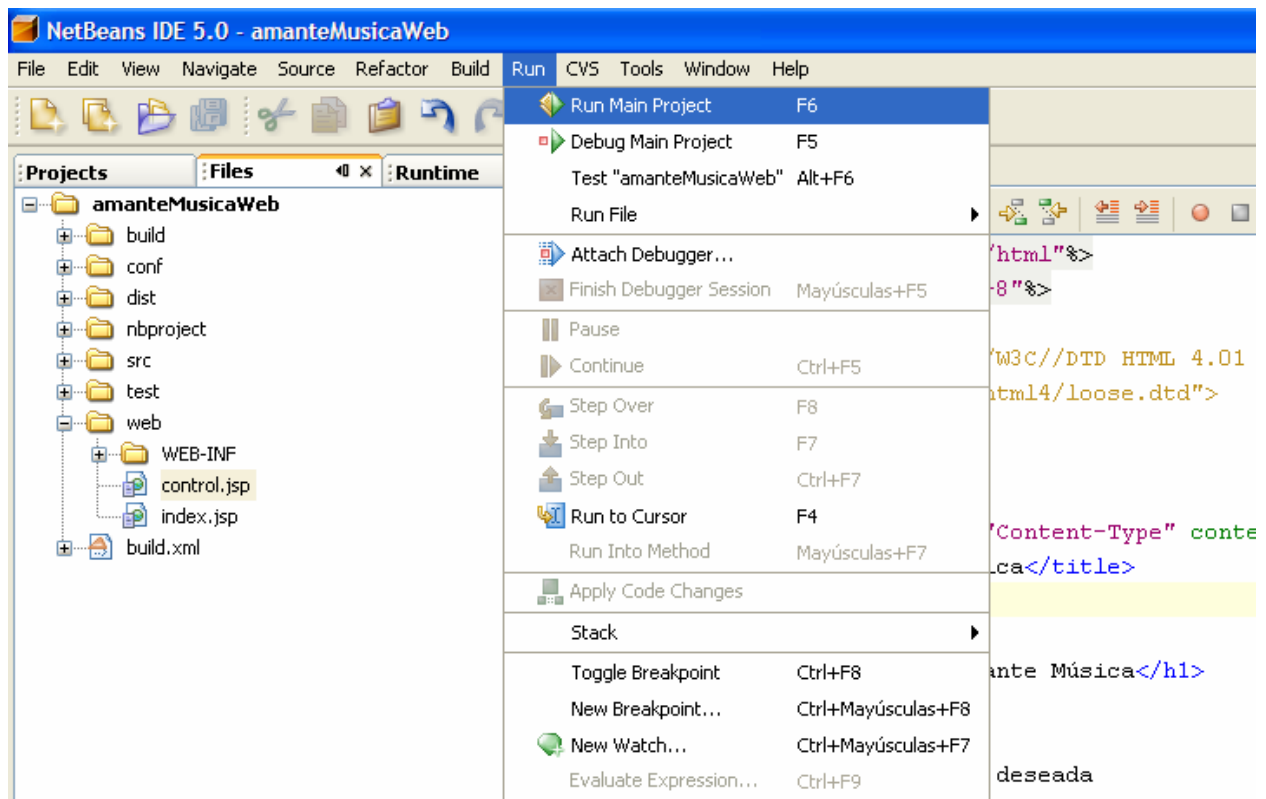


Figura 4.21



Figura 4.22

- Al hacerlo NetBeans, compilará las páginas JSP y las clases del proyecto. Enseguida iniciará la ejecución del servidor de aplicaciones que tiene empotrado, mostrando el proceso en la ventana de salida: **Output:**, figura 4.23. Al terminar le enviará la página JSP principal (index.jsp) al navegador Web, figura 4.24.

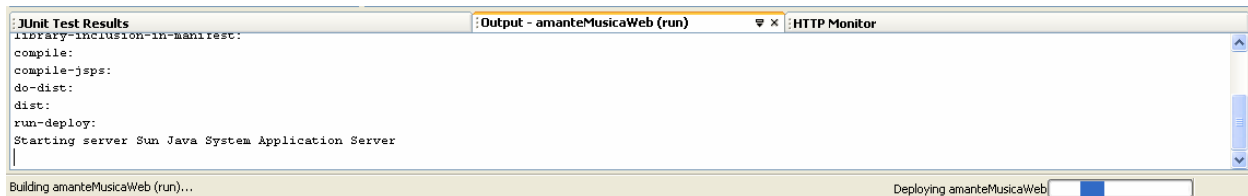


Figura 4.23

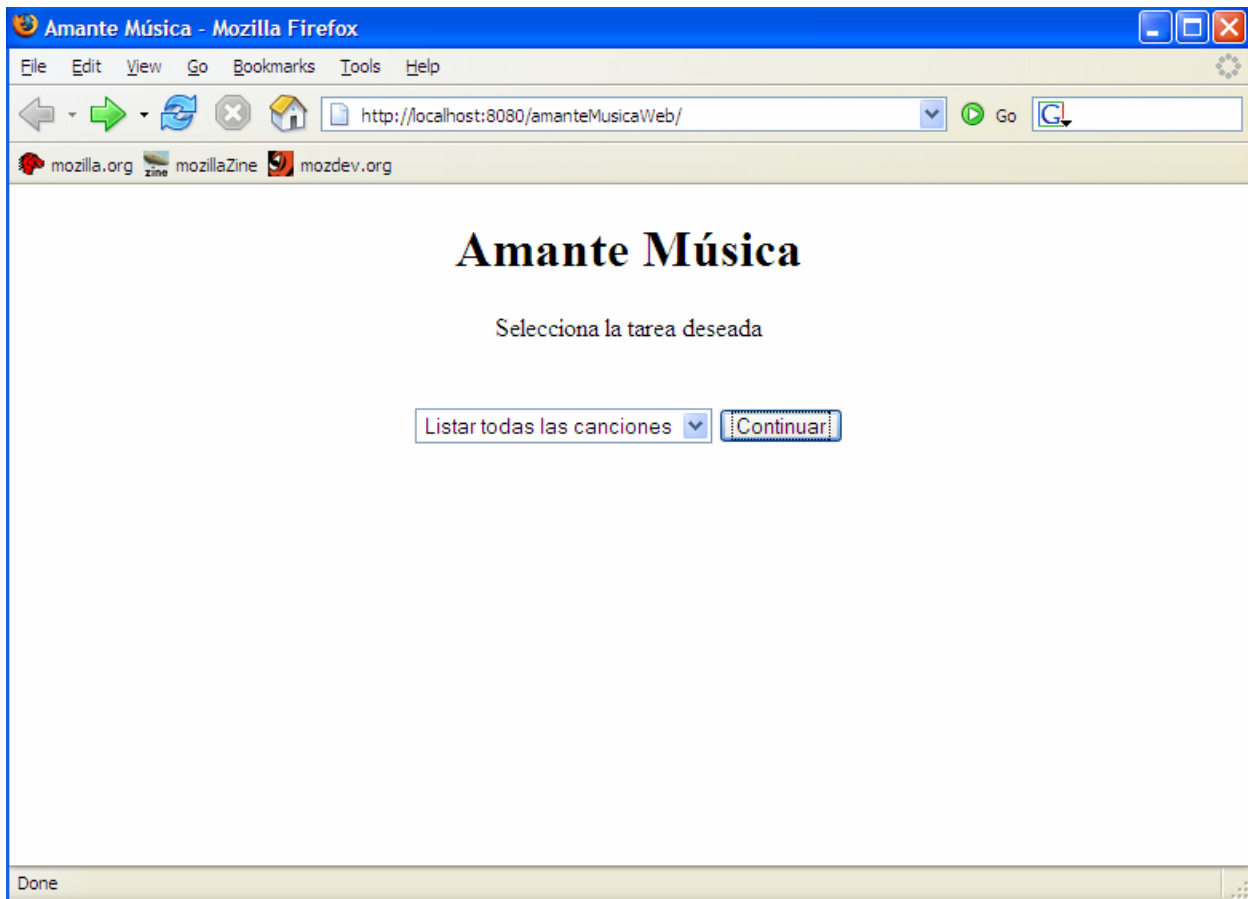


Figura 4.24

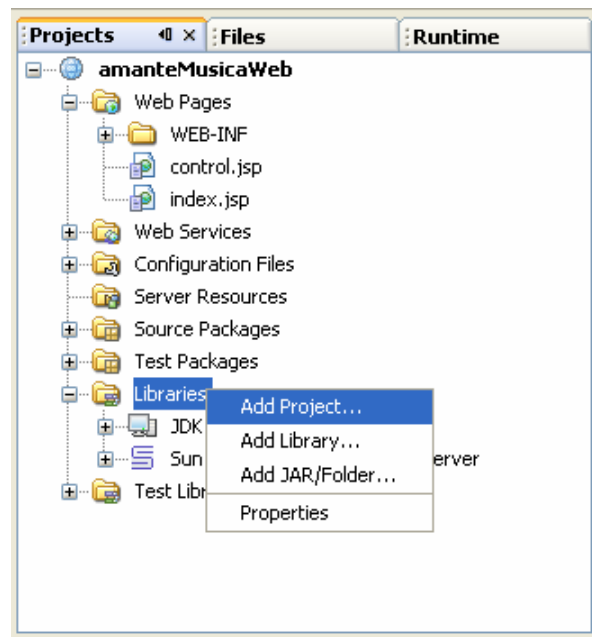
# Dependencias entre Proyectos

Al desarrollar una aplicación grande con un gran número de paquetes y clases, es común separar el código en varios proyectos. Cada proyecto podría tener su propia clase principal, posiblemente para probar las clases de ese proyecto. Un proyecto puede depender de uno o más proyectos, esto es, utiliza las clases de esos proyectos.

NetBeans nos permite establecer que un proyecto depende de otro, dándonos acceso a las clases de éste último. Para ello NetBeans le agrega a nuestro proyecto el archivo de distribución o JAR (el archivo con extensión “.jar”) del otro proyecto. Recuerde que ese archivo JAR se genera al compilar el proyecto.

Para hacer que un proyecto dependa de otro se sigue el siguiente procedimiento:

1. En el árbol de proyectos hacer clic con el botón derecho en el nodo **Libraries**, figura 4.25.



**Figura 4.25**

2. Aparecerá el cuadro de diálogo para seleccionar el proyecto del que dependerá nuestro proyecto, figura 4.26
3. Este cuadro de diálogo tiene, en la parte superior, una caja combinada para seleccionar la carpeta con los proyectos existentes, mismos que se muestran en el recuadro central (cada uno en su propia carpeta). Al seleccionar un proyecto, su nombre aparece del lado derecho y debajo de éste la lista de archivos JAR que hay en este proyecto. Se selecciona el archivo JAR deseado y se presiona el botón **Add Project JAR Files**.

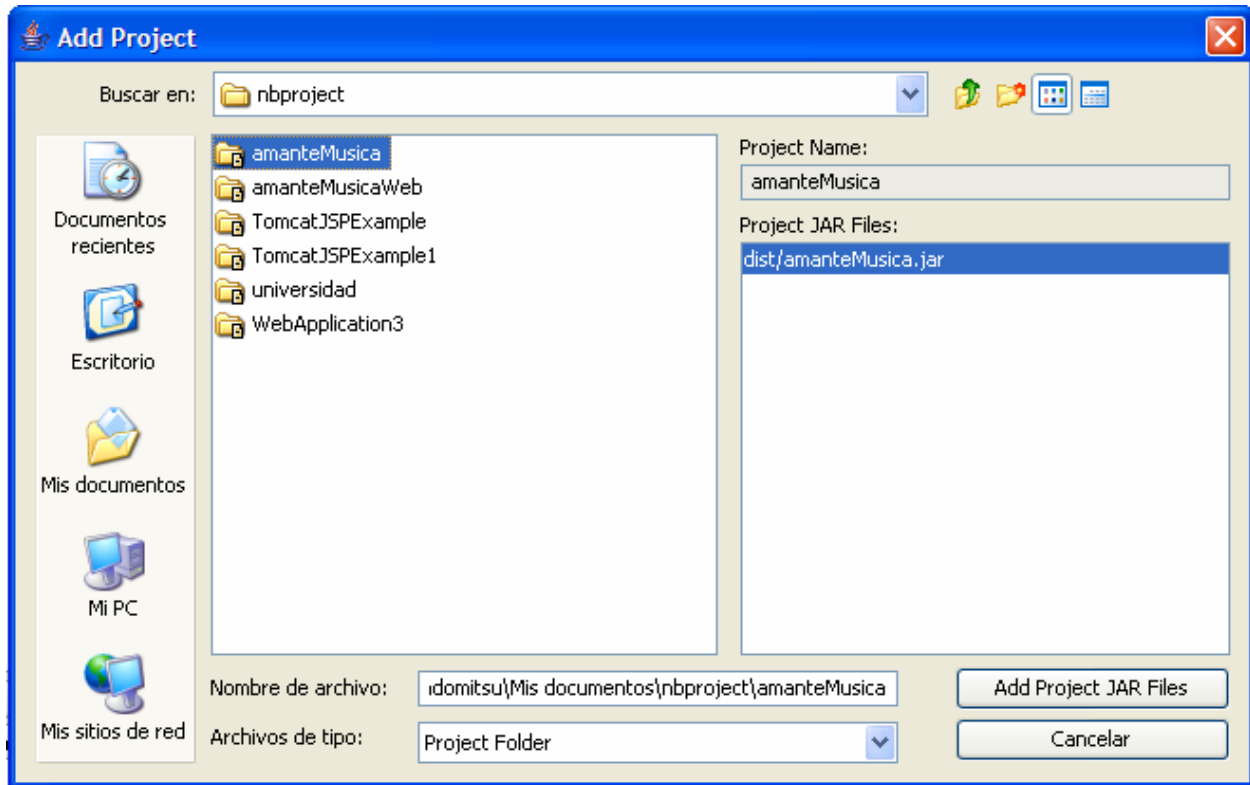


Figura 4.26

4. Al hacerlo, NetBeans crea la dependencia de nuestro proyecto con el proyecto seleccionado y crea un nodo dentro del nodo Libraries, con el nombre del proyecto del que depende nuestro proyecto, figura 4.27.

## NetBeans y Múltiples Proyectos

NetBeans nos permite tener abiertos varios proyectos y crear, editar, compilar clases en ellos. Sólo uno de ellos será nominado el proyecto principal. Aquel que será compilado y ejecutado al seleccionar las opciones: **Build Main Project** (Compilar Proyecto Principal) y **Run Main Project** (Ejecutar Proyecto Principal). Sin embargo, podemos cambiar la designación del proyecto principal entre los diferentes proyectos. Para abrir un proyecto en NetBeans se sigue el siguiente procedimiento:

1. Seleccionar la opción File/Open Project ... de la barra de menús de NetBeans, figura 4.28.
2. Al hacerlo, NetBeans desplegará un cuadro de diálogo para seleccionar el proyecto a abrir, figura 4.29.



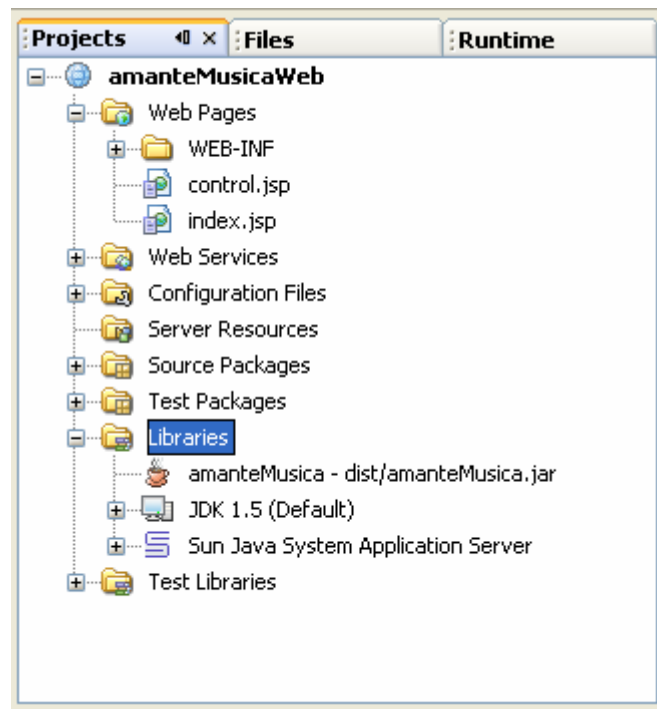


Figura 4.27

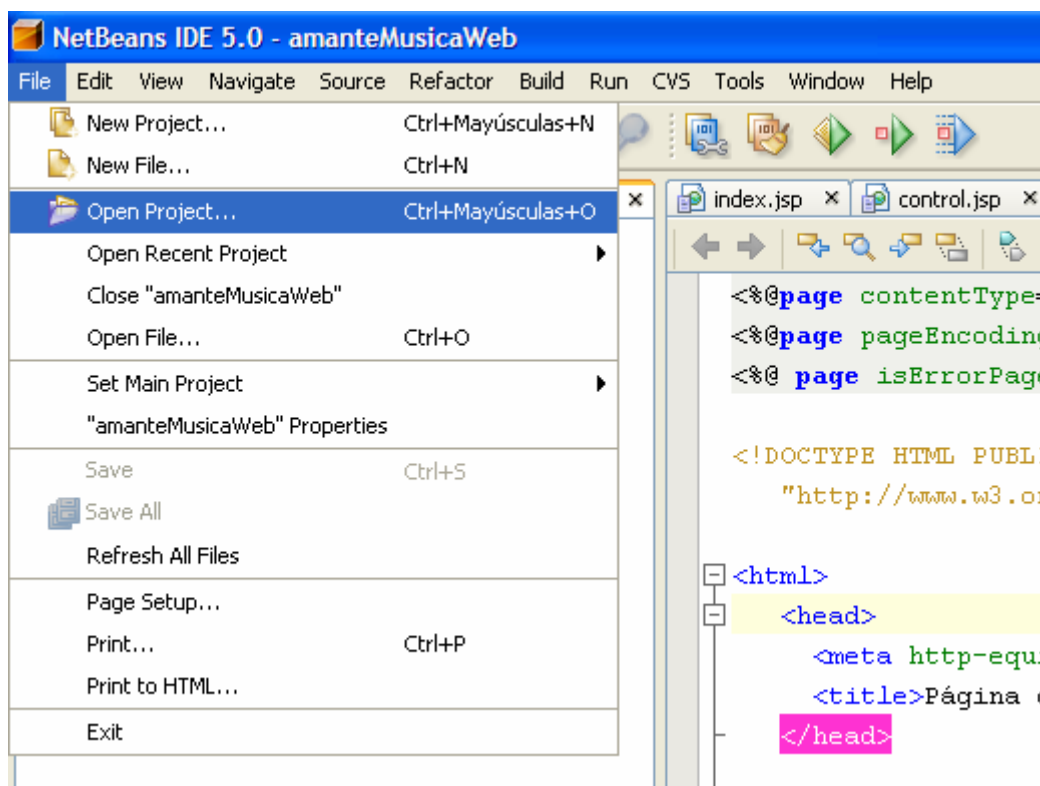


Figura 4.28

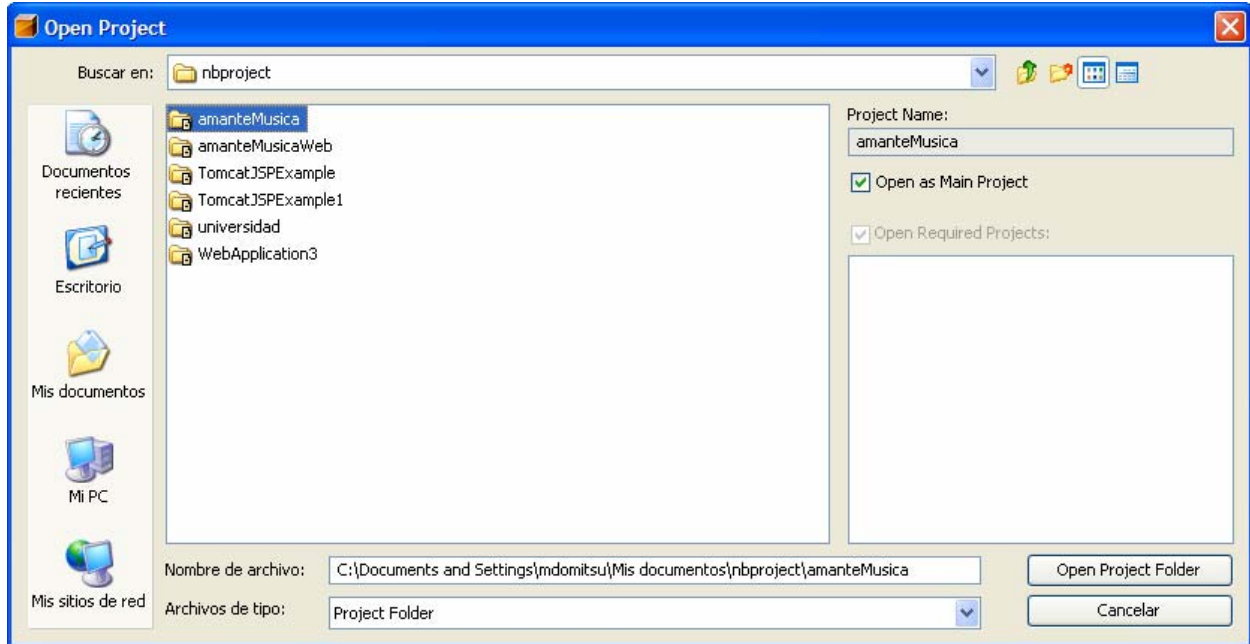


Figura 4.29

3. Este cuadro de diálogo tiene, en la parte superior, una caja combinada para seleccionar la carpeta con los proyectos existentes, mismos que se muestran en el recuadro central (cada uno en su propia carpeta). Al seleccionar un proyecto, su nombre aparece del lado derecho y debajo de éste hay dos casillas de selección: **Open as Main Project** (Abrir como proyecto principal) y **Open Required Projects** (Abrir Proyectos Requeridos). La primera nos permite establecer que el proyecto a abrir se convierta en el proyecto principal y la segunda, que sólo estará activa si el proyecto a abrir depende de otros proyectos, hace que NetBeans abra el proyecto deseado además de abrir los proyectos de los que depende. Una vez seleccionadas las opciones deseadas se presiona el botón **Open Project Folder**.
4. Al hacerlo, NetBeans abrirá el proyecto o proyectos seleccionados, mostrando los proyectos en el árbol de proyectos, figura 4.30.
5. Para cambiar la designación de proyecto principal, hacemos clic con el botón derecho en el nodo del proyecto que deseamos que sea el proyecto principal y seleccionamos la opción **Set Main Project**, del menú emergente, figura 4.31.

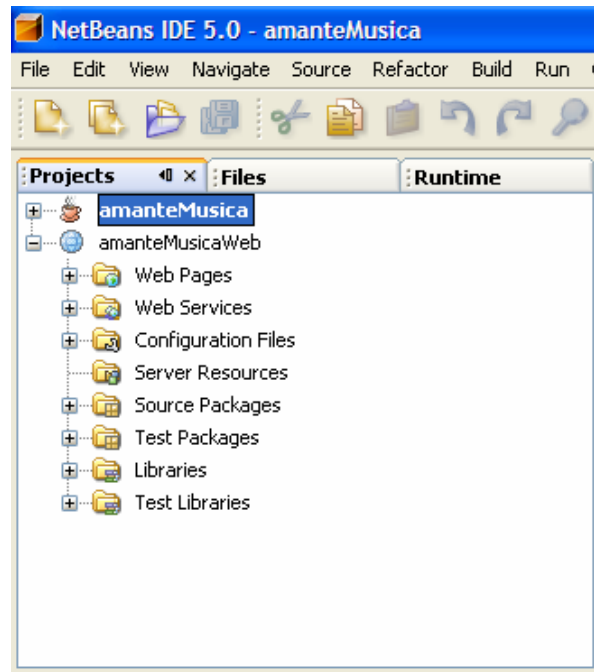


Figura 4.30

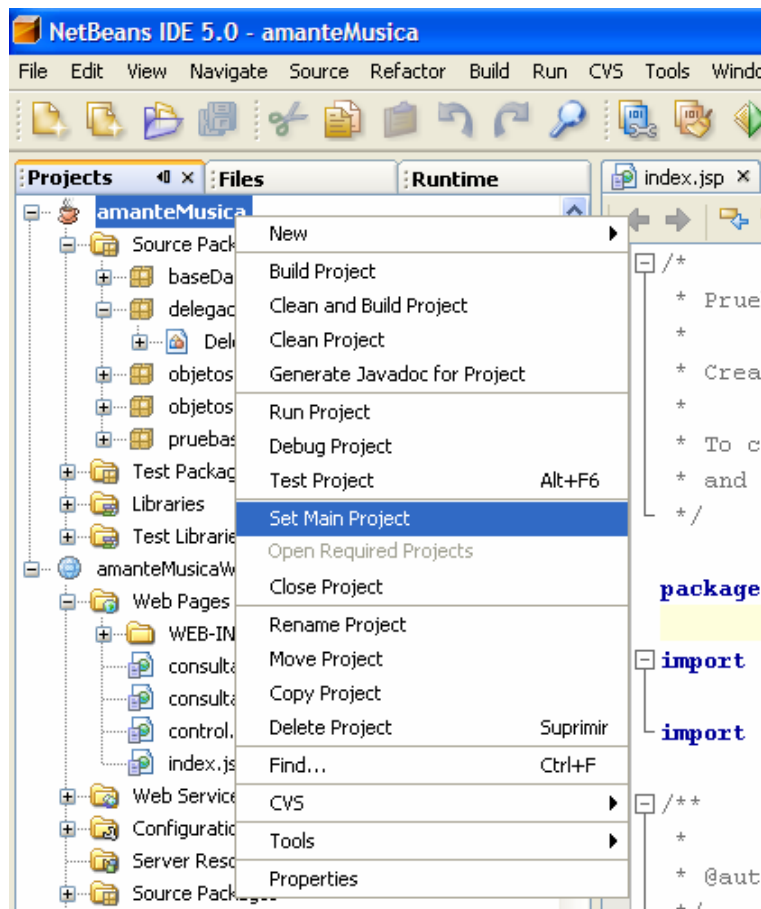


Figura 4.31.

# Páginas JSP con JSTL

Si en una página JSP queremos usar elementos de acción de la Biblioteca de Etiquetas Estándar de JSP, JSTL, debemos agregarle al proyecto de la aplicación la biblioteca de etiquetas. Para ello se sigue el siguiente procedimiento:

1. Expanda el nodo **Libraries** (bibliotecas) del proyecto al que se le va a agregar el conector para ver las bibliotecas que tiene agregadas, figura 4.32.

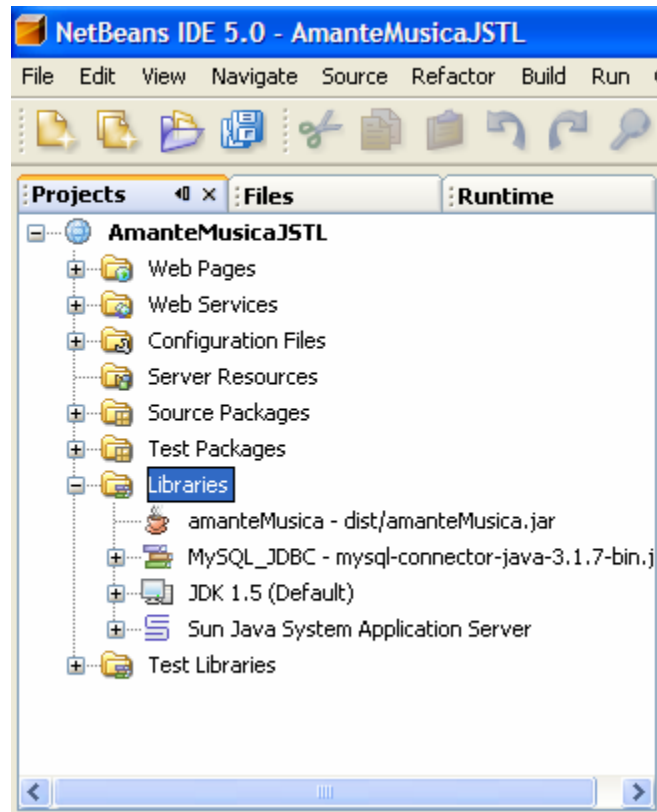


Figura 4.32

2. Haga clic con el botón derecho sobre el nodo **Libraries** del proyecto y seleccione la opción **Add Library ...** del menú emergente, figura 4.33.
3. Aparece un cuadro de diálogo para seleccionar la biblioteca a agregar, en este caso la biblioteca de Etiquetas estándar de JSP: **JSTL 1.1**, Figura 4.34. Lo seleccionamos y presionamos el botón **Add Library**.
4. El cuadro de diálogo desaparece y veremos que el conector se ha agregado al nodo **Libraries** en el árbol del proyecto, figura 4.35.

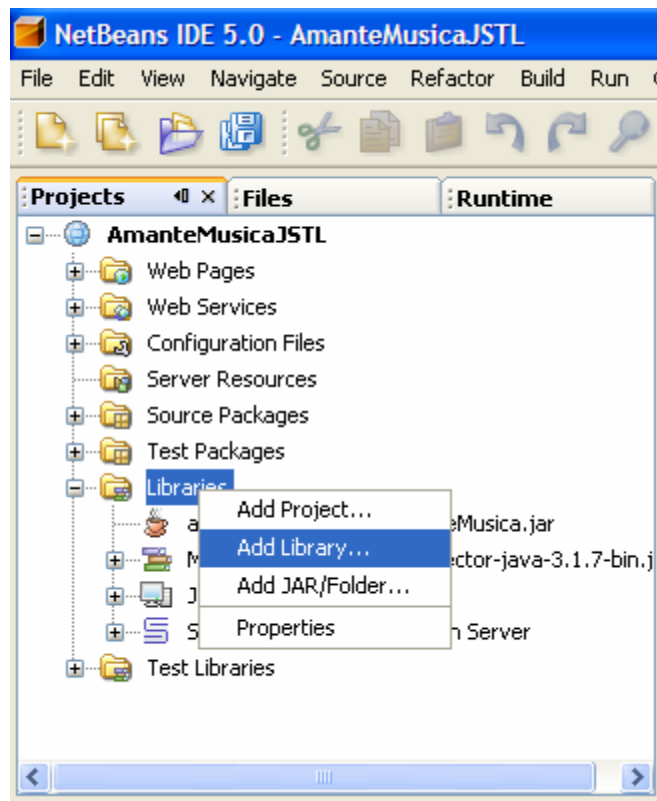


Figura 4.33

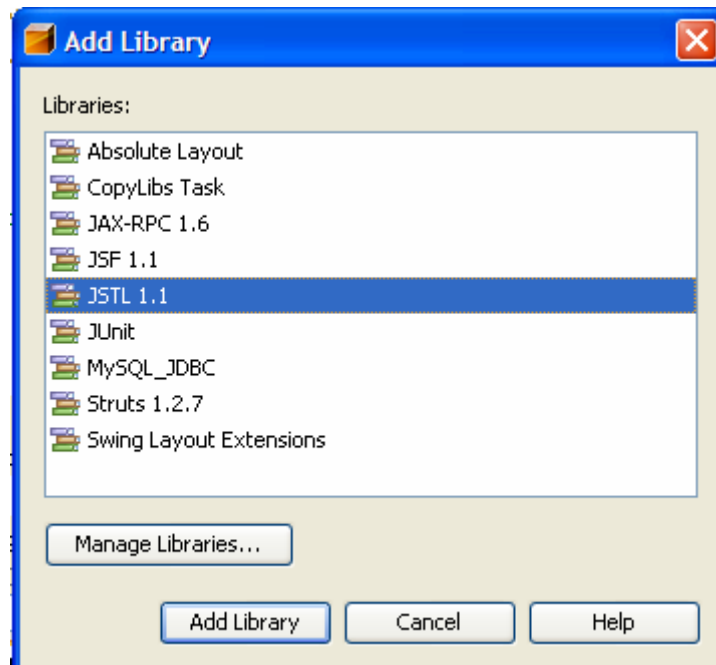


Figura 4.34

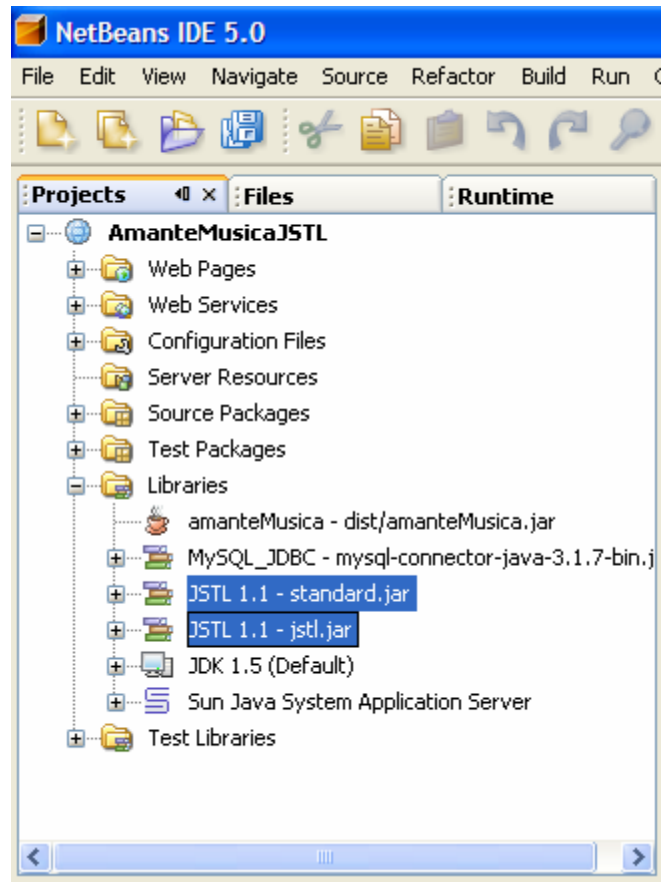


Figura 4.35