# Manual de instalación

**JWebSocket**

**Arduino Remote Control Demo**

**1.0**

# Control de versiones

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Versión** | **Descripción** | **Autor** |
| 13/02/2012 | 1.0 | Creación del documento | Dariel Noa Graverán |

1. **Proceso de descarga de la aplicación**

La aplicación está dividida en dos partes, la aplicación cliente, y el servidor, además es necesario descargar las librerías nativas que permiten que el servidor pueda escribir datos en el puerto USB, seguidamente se podrán descargar los elementos antes mencionados:

|  |  |  |
| --- | --- | --- |
| **Paquete** | **Tamaño** | **Descargar** |
| Arduino Remote Control Server | **83.9 Mb** | [Descargar](http://repo.hab.uci.cu/svn/tesis/Segundo_Corte_de_Tesis/JWS/Dariel_Noa/2do_Corte/Implementacion/server-side) |
| Arduino Remote Control Client | **6.84 Mb** | [Descargar](http://repo.hab.uci.cu/svn/tesis/Segundo_Corte_de_Tesis/JWS/Dariel_Noa/2do_Corte/Implementacion/client-side/demos) |
| Librerias nativas rxtx | **2.04 Mb** | [Descargar](http://repo.hab.uci.cu/svn/tesis/Segundo_Corte_de_Tesis/JWS/Dariel_Noa/2do_Corte/Implementacion/native-library) |
| Drivers Arduino Mega ADK | **4 Kb** | [Descargar](http://repo.hab.uci.cu/svn/tesis/Segundo_Corte_de_Tesis/JWS/Dariel_Noa/2do_Corte/Implementacion/driver-ArduinoMegaADK/Arduino_ADK.zip) |

**2. Características del entorno de instalación**

La aplicación debe ser ejecutada en un ambiente que cumpla con los siguientes requisitos:

* Sistema Operativo Windows XP o superior, GNU Linux x86/x64 o Mac OS X.
* Java Runtime Environment 7.
* Un puerto USB libre.
* Servidor Web Apache u otro similar (no es necesario instalar PHP).
* Librerías nativas rxtx 2.1.7, para la comunicación con el puerto USB.
* La placa Arduino debe de tener alojado en su memoria el programa con el que se trabajará.

1. **Proceso de instalación**

Lo primero que se debe hacer, es copiar la aplicación cliente hacia el directorio raíz del servidor web. Esto es necesario porque la versión usada de jWebSocket, necesita que el cliente este bajo una dirección web, dado que esta validado por defecto, en la configuración del servidor que la aplicación cliente este bajo la URL: http://localhost.

Luego se procede a indicarle a la máquina virtual de Java, que ficheros debe de usar para que la aplicación pueda manejar el puerto, para ello se deben de copiar los ficheros de las librerías nativas de control de puerto serial para la carpeta de binarios del JRE (Java Runtime Environment):

* Windows: Copiar los ficheros rxtxParallel.dll y rxtxSerial.dll en la ubicación C:/Program Files/Java/jre7/bin/
* Linux: Copiar el fichero librxtxSerial.so en la ubicación /jre/lib/, por debajo de la carpeta de la Máquina Virtual de Java.
* Mac OS X: Copiar el fichero librxtxSerial.jnilib en la carpeta /Library/Java/Extensions/

Seguidamente se debe de conectar el circuito Arduino a un puerto USB, verificar una vez conectado, que todos los indicadores de tierra (GND) estén encendidos. Para el desarrollo de la solución se usó Arduino Mega ADK, sin embargo para otros tipos de placa de la plataforma Arduino, el comportamiento sería igual. Es necesario disponer de 4 LEDs de colores azul, rojo, verde y amarillo que se conectarán a los pines de entrada/salida (I/O) 12, 8, 7 y 4 respectivamente además se debe tener un joystick de dos conexiones correspondientes a las coordenadas (x, y), el cual se conectará al pin de entrada 0 (I0) el conector x y al pin de entrada 1(I1) el conector y. Una vez conectado el joystick verificar que los indicadores tierra (GND) estén encendidos. Para facilitar las conexiones sería muy bueno disponer de un TinkerKit, consiste en un agrego que se le acopla al circuito Arduino para facilitar las conexiones con los LEDs y el joystick.

Para echar a andar la aplicación no es necesario tener el código fuente del proyecto, o sea se dispondrá de un fichero de tipo jar, que será ejecutado mediante la consola. Este fichero luego de ser descargado puede ser copiado en cualquier directorio.

**3.1 Hardware de la aplicación**

La aplicación tiene como requisito fundamental la existencia de un micro-controlador Arduino Mega ADK. Dicho dispositivo se conecta a la PC mediante el puerto USB, por lo que es necesario instalar drivers para que la aplicación pueda manejarlo.

**Instalación de drivers en Windows**

Una vez que la placa se haya conectado en Windows, el sistema mostrara el asistente para añadir nuevo hardware. Aquí le indicamos que no se conecte a Windows Update, y después le damos siguiente.

Nota: solo se especificara la instalación en Windows, para instalar Arduino en otros sistemas operativos, documentarse en el sitio <http://arduino.cc>.



Figura 3.1: Asistente para Nuevo Hardware en MS-Windows. Paso 1.

En la siguiente ventana seleccionar: *Instalar desde una lista o ubicación especificada (Avanzado)* y luego click en siguiente.



Figura 3.2: Asistente para Nuevo Hardware en MS-Windows. Paso 2.

Marcar buscar los drivers desde una ubicación en específica. Esta ubicación será el directorio donde se encuentra el driver descargado.



Figura 3.3: Asistente para Nuevo Hardware en MS-Windows. Paso 3.

El asistente indicara que encontró Arduino Mega ADK, luego dar clic en finalizar.

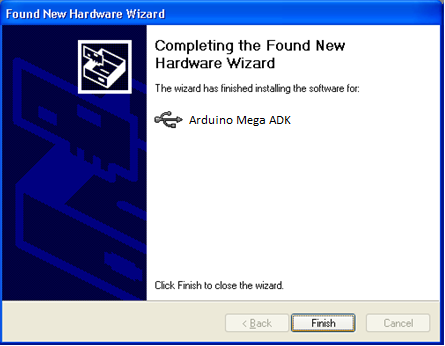


Figura 3.4: Asistente para Nuevo Hardware en MS-Windows. Paso 4.

Luego buscar en el administrador de dispositivos cual es el nombre del puerto que tiene asignado Arduino Mega ADK.

1. **Opciones de configuración**

|  |
| --- |
| *$JWEBSOCKET\_HOME/conf/EventsPlugIn/rc-application/app-plugins/rc.xml* |

La aplicación solo tiene un punto para su configuración, consiste en un fichero XML, o sea el fichero asociado al plug-in: *rc.xml* en el cual se especifica el puerto donde se deben enviar y recibir los datos. Este fichero se encuentra ubicado en la dirección:

Al abrir el fichero con un editor de texto se debe de modificar la línea 8 especificando cual será el puerto donde se encuentra conectado Arduino, ejemplo:

|  |
| --- |
| *<constructor-arg><value>****COM3****</value></constructor-arg>* |

En Windows el puerto sería: COM0,. . ., COM3; en Linux: /dev/tty/USB0, /dev/tty/USB1 o similar y en MAC probablemente sea: /dev/tty.usbserial-1B1.

**5. Puesta en funcionamiento**

Después de que el programa se encuentre en la placa y haber configurado el puerto, se procede a ejecutar el servidor por la consola mediante el siguiente comando:

|  |
| --- |
| *java -jar $JWEBSOCKET\_HOME/libs/jWebSocketServer-Bundle-1.0.jar* |

Una vez que el servidor se esté ejecutando, se procede a iniciar la aplicación cliente desde su URL local, mediante un navegador web que tenga soporte para websocket. En la parte superior derecha debe indicar que se estableció la conexión con el servidor, si esto ocurre ya la aplicación esta lista para ser probada.

**5. Administración de la aplicación**

Para la administración de la aplicación Arduino Remote Control Demo se procede de la misma forma en que se configura EventsPlugIn en el servidor jWebSocket. En caso de ocurrir cualquier problema durante el proceso de instalación y/o configuración de la aplicación puede consultar a:

**Dariel Noa Graverán**

**Correo:** dariel87@gmail.com