

Antes de poner en funcionamiento a Justina, debemos conocer los software con los que trabaja y los requerimientos para su correcto funcionamiento.

Como primer paso debemos conocer el software necesario para el funcionamiento de Justina.

1.1. Software necesario

Se requiere lo siguiente:

- Ubuntu 14.04.1 (This is the tested version)
- ROS Indigo desktop full
- OpenNI + PrimeSense drivers
- OpenCV 2.4.8 or 2.4.9. Compiled with OpenNi, WITHOUT OpenCL, WITHOUT Cuda, with Eigen
- PCL 1.6

Para conocer la forma de instalar ROS, OpenNI, los drivers PrimeSense y OpenCV 2.4.9 por favor acude al apéndice B (software).

Como siguiente paso obtener el software de Justina, para esto debemos descargar todas las carpetas con las que se ha trabajado Justina.

1.2. Obtención de la carpeta de Git hub

Todos los repositorios del software de Justina se encuentran en Git hub (así como este manual y es de donde podrás descargar futuras versiones). Existen dos formas para obtener la carpeta contenedora con todo lo necesario para empezar:

La primera es ir a la dirección "<https://github.com/RobotJustina/JUSTINA/tree/develop>" descargarlo con el botón color verde que dice `clone or download`".

te saldrá una opción para seleccionar la ubicación en la que desees guardar el archivo .zip

Busca la carpeta contenedora y descomprime el archivo. Al descomprimirlo obtendrás una carpeta llamada "JUSTINA-develop" la cual contiene todo lo necesario para utilizar a Justina.

La segunda opción

1.2.1. ROS Introducción

ROS es un *middleware* de código abierto (open source) que provee la funcionalidad comúnmente necesaria en el desarrollo de software para robots móviles autónomos, como paso de mensajes y manejo de paquetes. La robot Justina utiliza ROS como plataforma de desarrollo.

ROS puede describirse en dos niveles conceptuales: el sistema de archivos y el grafo de procesos.

El sistema de archivos. Se refiere al modo en que están organizados los recursos en disco:

- **Workspace:** Se refiere a las carpetas que contienen paquetes de ROS.
- **Paquete:** Es la principal unidad de organización de software en ROS. Pueden contener nodos, bibliotecas, datasets, archivos de configuración y otros.
- **Manifiesto:** Definido por el archivo `package.xml` en cada paquete. Provee meta-datos acerca de cada paquete.

- **Mensaje:** Archivos con extensión `.msg`. Definen estructuras de datos para el paso de mensajes en ROS.
- **Servicio:** Archivos con extensión `.srv`. Definen estructuras de tipo request-response. Utilizan mensajes para dicha definición.

Grafo de procesos. Es una red *peer-to-peer* de procesos. Los componentes básicos son:

- **Roscore:** Inicializa el sistema ROS: un master + rosout + un servidor de parámetros.
- **Nodos:** Es simplemente un ejecutable que usa ROS para comunicarse con otros nodos.
- **Tópicos:** Algo similar a una variable cuyo contenido puede ser compartido entre todos los nodos mediante un patrón de publicación y suscripción.
- **Servicios:** Otra forma de comunicar nodos pero con un patrón de petición y respuesta.
- **Servidor de parámetros:** Es un diccionario compartido. Todos los nodos pueden leer y escribir parámetros en tiempo de ejecución.

Si no cuenta con ROS o no conoce la manera de instalarlo en ubuntu diríjase al apéndice "software".

1.3. Instalación completa del software de Justina

Una vez instalado ROS procedemos a instalar el software de Justina, para esto abrir una terminal y seguir las siguientes instrucciones.

1. meterse a la carpeta JUSTINA-develop y ejecutar JustinaSetup.sh

2. Aceptar cada que pregunte. Esto nos llevara varios minutos.
3. Una vez instalado el software, debemos habilitar el uso de los puertos USB para ROS, para esto nos metemos al directorio "JUSTINA-develop/ToInstall /USB (si quieres seguir las instrucciones más detalladamente, en la misma dirección abrir el archivo "instructions") una vez dentro de la carpeta ejecutar el siguiente comando "sudo cp 80-justinaRobot.rules /etc/udev/rules.d/"
4. Te pedirá la contraseña. Una vez termines de ejecutar el comando, se debe ejecutar el siguiente:

Listo, ya tienes instalado el software de Justina.

1.4. Cómo compilar los repositorios de Justina

Para compilar los repositorios de Justina simplemente ve al directorio "JUSTINA-develop/catkin_ws" en este directorio ejecutamos el siguiente comando `catkin_make`". Esto nos llevara varios minutos.

Una vez compilados todos los repositorios ejecutar el siguiente comando dentro de la misma carpeta "source devel/setup.bash".

Listo, ahora el software de Justina está instalado y los repositorios compilados y listos para usarse.

1.5. RViz y GUI de Justina

Para probar el funcionamiento del hardware y software de Justina utilizaremos RViz y la GUI. Para ejecutar estos programas utilizamos el comando `roslaunch surge_et_ambula justina.launch`".

1.6. Simulación en el RViz y GUI de Justina

Cuando no tenemos conectado el robot Justina a nuestras laptops lo único que podemos hacer es simular a Justina en nuestras laptops, para esto ejecutamos el siguiente código - `roslaunch surge_et_ambula justina_simul.launch`".