

Programación en Python con Robots

JUICa 2012

Grupo de Desarrollo Lihuen - Fernando López - Sofía Martín

Noviembre 2012



Programación en Python con Robots

Temario

- El proyecto: **Entre Robots, juegos y lenguajes: una experiencia de aprendizaje real en las aulas**
- Los Robots
- Introducción a Python

Primer curso de programación usando robots y Python

Objetivos

- Incentivar el interés de los niños y jóvenes en el uso de la tecnología con fines educativos
- Trabajar en mecanismos de resolución de problemas

Más específicamente ...

- Difundir y trabajar con Software Libre
- Enseñar aspectos de programación utilizando Python

Primer curso de programación usando robots y Python

Objetivos

- Incentivar el interés de los niños y jóvenes en el uso de la tecnología con fines educativos
- Trabajar en mecanismos de resolución de problemas

Más específicamente ...

- Difundir y trabajar con Software Libre
- Enseñar aspectos de programación utilizando Python

Entre Robots, juegos y lenguajes: una experiencia de aprendizaje real en las aulas

El proyecto

- 2012: Subsidiado por la Fundación YPF
- Equipamiento en las escuelas
 - 20 robots
 - 1 netbook
 - 1 filmadora
- Capacitación para docentes y alumnos
- Acompañamiento para su implementación

El proyecto de robots

En las escuelas

Características

- 10 escuelas.
- 10/15 docentes por escuela.
- 5 alumnos por escuela.
- Conocimientos previos desiguales.

El proyecto de robots

En las escuelas

Características

- 10 escuelas.
- 10/15 docentes por escuela.
- 5 alumnos por escuela.
- Conocimientos previos desiguales.

El proyecto de robots

En las escuelas

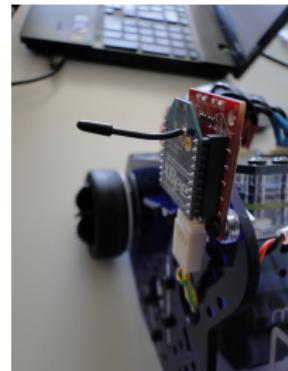
Características

- 10 escuelas.
- 10/15 docentes por escuela.
- 5 alumnos por escuela.
- Conocimientos previos desiguales.

Programando con Python y Robots

Recursos Necesarios

- Plataforma Linux
- Robots inalámbricos



Programando con Python y Robots

Contenidos Mínimos

- Conceptos básicos de programación.
- Paradigma...
- Introducción al lenguaje Python.
- Implementación de algoritmos sencillos.
- Funciones para manipulación del robot.
- Manual de guía.

Programando con Python y Robots

Contenidos Mínimos

- Conceptos básicos de programación.
- Paradigma...
- Introducción al lenguaje Python.
- Implementación de algoritmos sencillos.
- Funciones para manipulación del robot.
- Manual de guía.

Programando con Python y Robots

Contenidos Mínimos

- Conceptos básicos de programación.
- Paradigma...
- Introducción al lenguaje Python.
- Implementación de algoritmos sencillos.
- Funciones para manipulación del robot.
- Manual de guía.

Programando con Python y Robots

Actividades

Se organizan actividades lúdicas (juegos, carreras, obras de teatro, bailes, etc.)



Donde se intenta que los alumnos presenten sus trabajos en una jornada (interna o externa)

Vídeo de los chicos practicando

El proyecto de software libre

En las escuelas

Características

- Aplicaciones educativas
- Conexión con el proyecto Conectar Igualdad.
- Jornadas en las escuelas.
- Aplicación del uso de la aplicación al área de estudio.

El proyecto de software libre

En las escuelas

Características

- Aplicaciones educativas
- Conexión con el proyecto Conectar Igualdad.
- Jornadas en las escuelas.
- Aplicación del uso de la aplicación al área de estudio.

El proyecto de software libre

En las escuelas

Características

- Aplicaciones educativas
- Conexión con el proyecto Conectar Igualdad.
- Jornadas en las escuelas.
- Aplicación del uso de la aplicación al área de estudio.

El proyecto de software libre

En las escuelas

Características

- Aplicaciones educativas
- Conexión con el proyecto Conectar Igualdad.
- Jornadas en las escuelas.
- Aplicación del uso de la aplicación al área de estudio.

Los Robots

Características Básicas

Los Robots

Los Robots

Características Básicas

- Basados en **hardware y software libre**
- Inalámbricos
- 3 Baterías AA
- Varios Sensores
- Nacionales - Empresa RobotGroup



Los Robots

Detalle

- Forman parte del sistema constructivo Multiplo de RobotGroup.
- Pueden programarse en C++ o con bloques (MiniBloq).
- Las piezas plásticas pueden replicarse con una cortadora láser.
- El controlador del robot es una placa Arduino con algunas modificaciones.

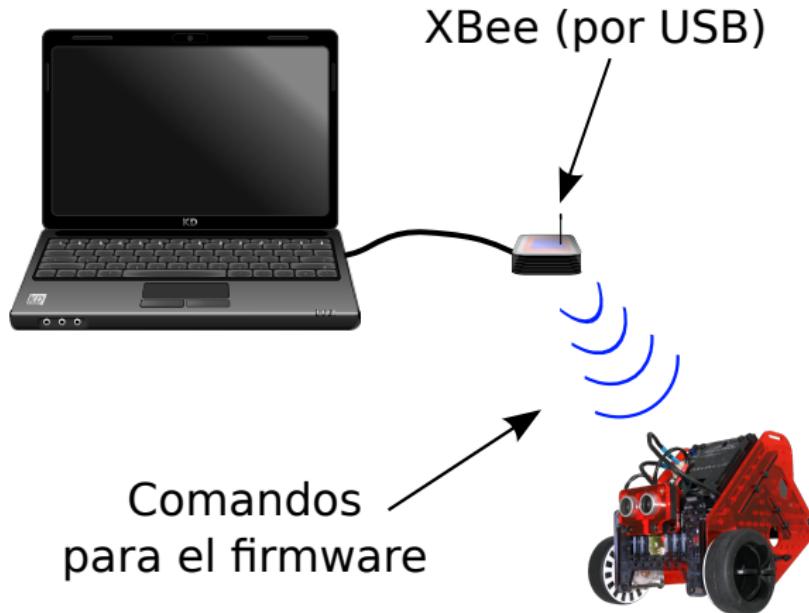
Los Robots

Robots utilizados en el curso

- Son versiones **modificadas** del N6.
- Cuentan con un módulo XBee para la comunicación.
- Tienen un firmware basado en Firmata que permite controlarlos de forma remota.
- API en Python desarrollada en conjunto entre el LINTI y RobotGroup.
 - Clases de alto nivel → en conjunto.
 - PyFirmata → por terceros, modificada.

Los Robots

Esquema de conexión



Sensores

Sensores utilizados en el curso

- **Sensor ultrasónico**

- Funcionamiento
- ping()
- getObstacle()
- Alternativa más económica...

- **Sensores de línea / cuenta vueltas**

- Funcionamiento
- getLine()
- getWheels()

Otros sensores

- Fotoresistencias
- Cámara fotográfica → se podría suplir...
- Un switch para detectar obstáculos
- Micrófono
 - Intensidad de ruido ambiente
 - Intensidad en determinada frecuencia
- Para algunos sería práctico ampliar la API

Python

Características Básicas

Introducción a Python

Python

Características Básicas

- Es interactivo
- Multi plataforma
- Permite una programación prolífica e intuitiva
- Es posible programar aplicaciones complejas
- Muy usado
- Es **software libre**



Python

El Intérprete

Desde el shell de Python

```
Archivo Editor Ver Terminal Ayuda
claudia@lilithan:~$ python
Python 2.6.6 (r266:80292, Dec 26 2010, 22:31:48)
[GCC 4.4.5] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> [REDACTED]
```

Utilizando un IDE: Geany

```
Archieve Editar Desarrollar Vie Documento Proyecto Configuración Herramientas Ayuda

Símbolos | Funciones | Clases | Campos | Métodos | Propiedades | Eventos | Entradas | Salidas | Entradas Salidas

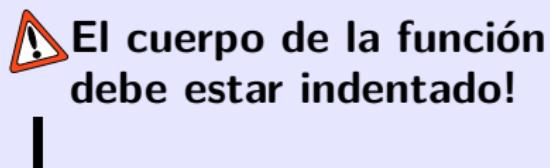
relativepath X
1 import pygame
2
3 from pygame import sys
4
5 from pygame.locals import *
6
7
8 def dibuja_rectangulo(ventana):
9     pygame.draw.rect(ventana,(255, 0, 0),(190, 300, 180, 20),0)
10    pygame.draw.rect(ventana,(255, 255, 0),(290, 300, 180, 20),0)
11    pygame.draw.rect(ventana,(0, 0, 255),(390, 300, 180, 20),0)
12
13    pygame.draw.rect(ventana,(0, 255, 0),(490, 300, 180, 20),0)
14
15    pygame.draw.rect(ventana,(255, 150, 150),(590, 300, 180, 20),0)
16
17    pygame.draw.rect(ventana,(0, 0, 0),(690, 300, 180, 20),0)
18
19    pygame.draw.rect(ventana,(0, 200, 0),(790, 300, 180, 20),0)
20
21    pygame.draw.rect(ventana,(0, 255, 255),(890, 300, 180, 20),0)
22
23    pygame.draw.rect(ventana,(255, 0, 255),(990, 300, 180, 20),0)
24
25
26 def dibuja_círculo(ventana):
27     pygame.draw.circle(ventana,(0, 255, 0),(125, 150),150,0)
28
29     pygame.draw.circle(ventana,(255, 255, 0),(225, 150),150,0)
30
31     pygame.draw.circle(ventana,(0, 0, 255),(325, 150),150,0)
32
33     pygame.draw.circle(ventana,(0, 255, 255),(425, 150),150,0)
34
35     pygame.draw.circle(ventana,(255, 0, 0),(525, 150),150,0)
36
37     pygame.draw.circle(ventana,(0, 0, 0),(625, 150),150,0)
38
39     pygame.draw.circle(ventana,(0, 200, 0),(725, 150),150,0)
40
41     pygame.draw.circle(ventana,(0, 255, 150),(825, 150),150,0)
42
43     pygame.draw.circle(ventana,(255, 150, 150),(925, 150),150,0)
44
45
46 def dibuja_lineas(ventana):
47     pygame.draw.line(ventana,(255, 255, 255),(500, 400),(500, 600),10)
48
49     pygame.draw.line(ventana,(255, 255, 255),(500, 400),(700, 400),10)
50
51     pygame.draw.line(ventana,(255, 255, 255),(500, 600),(700, 600),10)
52
53     pygame.draw.line(ventana,(255, 255, 255),(500, 400),(700, 600),10)
54
55
```

Características Generales del Lenguaje Python

Funciones

- Una función agrupa un conjunto de sentencias.
- Puede tener argumentos
- Sintaxis:

```
def nombre(parametros):  
    sentencial  
    sentencia2  
    sentencia3
```



Características Generales del Lenguaje Python

Estructuras de Control

Sentencia if

```
if expresion:  
    sentencial  
    sentencia2
```

Sentencia while

```
while expresion:  
    sentencial  
    sentencia2
```

Sentencia for

```
for var in lista:  
    sentencial  
    sentencia2
```

Materiales adicionales

- <http://robots.linti.unlp.edu.ar/>

Programando con ROBOTS

LINTI | Facultad de Informática UNLP

Página principal

Programando con Robots en la Semana de la ciencia

La Facultad de Informática participará, a través del programa Mundo Nuevo de la Universidad Nacional de La Plata, de la Semana Nacinal de la Ciencia y la Tecnología que se desarrollará del 11 al 22 de junio en la República de Los Niños.

Con el objetivo de acercar el conocimiento científico y tecnológico a la sociedad, el programa Mundo Nuevo llevará a cabo talleres, actividades lúdicas, espectáculos, muestras interactivas, dinámicas y jornadas recreativas. La Facultad de Informática se hará presente para acompañar la iniciativa y brindar información acerca de las actividades científicas y tecnológicas que realiza.

Estará presente el Proyecto de Extensión "Programando con robots y software libre" dirigido por la Lic. Claudia Bianchi, que fue la máxima atracción de la Muestra de la TEC 2012. De esta manera las escuelas y el público en general podrán disfrutar nuevamente de los robots y sus actividades programadas en Python.

También expondrá su trabajo el [Proyecto L-Buscar](#) cuyo objetivo fundamental es la concientización ambiental y el compromiso social. Este proyecto recibe equipamiento informático en desuso, lo recupera y lo dona a organizaciones sociales que lo necesitan, haciendo un aporte importante a la reducción de la brecha digital.

Asimismo, el martes 19 de junio a las 14:30 horas, integrantes de la línea de trabajo Concentración en Seguridad de la Información del Laboratorio de Investigación en Nuevas tecnologías Informáticas (LINTI) brindarán la charla "Distrutando de Internet en forma segura" para las escuelas que visiten la muestra. La propuesta impulsada por la Secretaría de Planeamiento y Políticas del Ministerio de Ciencia, Tecnología e Innovación Productiva de la Nación, busca ofrecer actividades gratuitas y abiertas a todo público para conocer más de cerca el mundo de la tecnología. Además, apunta a "generar un espacio de difusión y apropiación social del conocimiento para que chicos y grandes redescubran la ciencia en los grandes inventos y en la vida cotidiana".

De esta manera, la Facultad de Informática junto a Mundo Nuevo se sumará a las instituciones y organizaciones como museos, universidades, centros de investigación, observatorios y otras, con el objetivo de acercar el conocimiento científico y tecnológico a la comunidad.

12:15 2 M 2012 (APT)

Destacada participación del Proyecto Programando con Robots y Software libre en TEC La Plata

El proyecto Programando con Robots y Software libre tuvo una extensa participación en la muestra de ciencia y tecnología organizada por la Municipalidad de La Plata -TEC La Plata-, que se llevó a cabo del 19 de abril al 1 de mayo, siendo uno de los stands más visitados que cautivó la atención de los chicos y también de los adultos.

Emarcado dentro de la propuesta que realizó la Facultad de Informática, el proyecto expuso cómo se puede aprender programación de una manera didáctica y muy entretenida, apuntando a incentivar a los jóvenes en el uso de la tecnología. De

Primeros Pasos

Instalando

Primeros pasos

Primeros Pasos

Conectando el robot y la computadora

- Se conecta el módulo de comunicaciones a un puerto USB de la computadora.
- Luego se enciende el robot para que reciba las órdenes de la computadora.

Primeros Pasos

Conectando el robot y la computadora

- Se conecta el módulo de comunicaciones a un puerto USB de la computadora.
- Luego se enciende el robot para que reciba las órdenes de la computadora.



Primeros Pasos

Conectando el robot y la computadora

- Se conecta el módulo de comunicaciones a un puerto USB de la computadora.
- Luego se enciende el robot para que reciba las órdenes de la computadora.

Primeros Pasos

Conectando el robot y la computadora

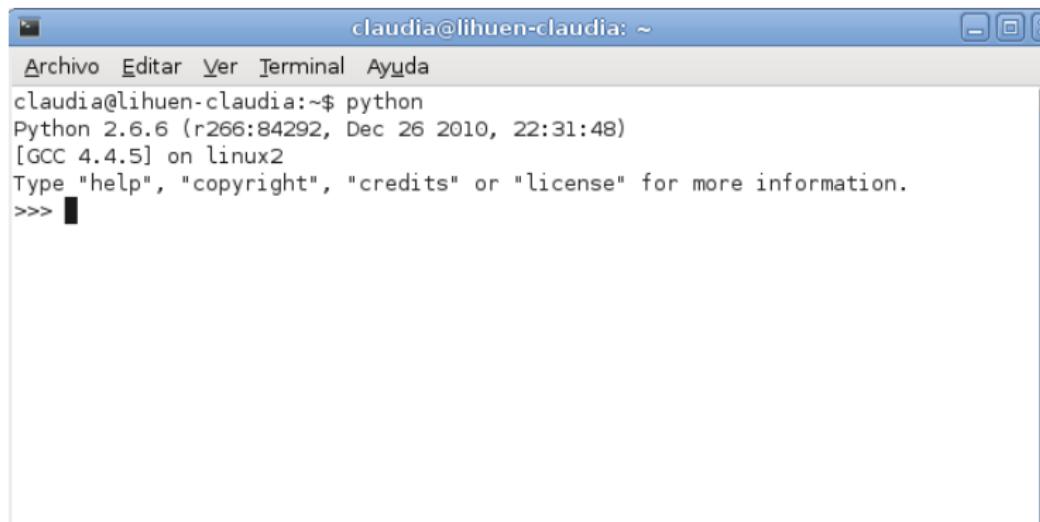
- Se conecta el módulo de comunicaciones a un puerto USB de la computadora.
- Luego se enciende el robot para que reciba las órdenes de la computadora.



Primeros Pasos

Conectando el robot y la computadora

- Importamos la librería para trabajar con el robot:
from duinobot import *
- Encender el robot



A screenshot of a terminal window titled "claudia@lihuen-claudia: ~". The window has a standard Linux-style title bar with icons for minimize, maximize, and close. Below the title bar is a menu bar with options: Archivo, Editar, Ver, Terminal, Ayuda. The main area of the terminal shows a Python session:

```
claudia@lihuen-claudia:~$ python
Python 2.6.6 (r266:84292, Dec 26 2010, 22:31:48)
[GCC 4.4.5] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> 
```

Primeros Pasos

Iniciando la actividad

```
from duinobot import *
b = Board("/dev/ttyUSB0")
b.report()
mi_robot = Robot(b, 0)
mi_robot.setId(3)
```

Primeros Pasos

¿Movemos el robot?

Hacia Adelante

forward(velocidad, tiempo)

Ejemplos:

`forward()`

`forward(1, 0.5)`

Hacia Atrás

backward(velocidad, tiempo)

Ejemplos:

`backward()`

`backward(1, 0.5)`

Probemos ...

`backward(-10)`

`forward(-10)`

Primeros Pasos

¿Movemos el robot?

Hacia Adelante

forward(velocidad, tiempo)

Ejemplos:

forward()

forward(1, 0.5)

Hacia Atrás

backward(velocidad, tiempo)

Ejemplos:

backward()

backward(1, 0.5)

Probemos ...

backward(-10)

forward(-10)

Primeros Pasos

¿Movemos el robot?

Girando a la Izquierda

turnLeft(velocidad, tiempo)

Ejemplos:

turnLeft()

turnLeft(1, 0.5)

Girando a la Derecha

turnRight(velocidad, tiempo)

Ejemplos:

turnRight()

turnRight(1, 0.5)

Primeros Pasos

Usando los sensores

¿Cómo obtenemos los valores de los sensores?

- `senses()`
- `ping()`
- `getObstacle()`

Demo en vivo con sensores

Primeros Pasos

Ejemplo con sensores

Evita obstáculos

```
robot.forward()
obstaculos = 0
while obstaculos < 10:
    if robot.getObstacle(15):
        robot.backward(40, 1)
        robot.turnLeft(50, 0.5)
        robot.forward()
        obstaculos = obstaculos + 1
```

Ampliando el manejo del robot

Motivación

- Incentivo a la programación usando teléfonos Android.
- Innovación en nuevas tecnologías.

Integración

- Los robots ejecutando algún programa
- Joystick
- Celulares

Ampliando el manejo del robot

Motivación

- Incentivo a la programación usando teléfonos Android.
- Innovación en nuevas tecnologías.

Integración

- Los robots ejecutando algún programa
- Joystick
- Celulares

Programando en Android

¿Qué es remotebot?

- Aplicación cliente-servidor.
- Está compuesto de:
 - Un servidor Python que se ejecuta en una PC con GNU/Linux.
 - Un cliente en Java que se ejecuta en Android.
- Permite controlar robots simples desde dispositivos Android.
- Contiene un Wrapper completo de la clase Robot.

Programando en Android

El código

```
from duinobot import *
b=Board("/dev/ttyUSB0")
r=Robot(b, 1)

r.forward(50, 2)
r.turnLeft(25, 2)
r.backward(50, 1)
r.forward(100, 1)
r.stop()

b.exit()
```

```
import ar.edu.unlp.linti.robot.*;
import ar.edu.unlp.linti.robot.exceptions
.*;
/* ... */
try {
    Board b = new Board(
        "http://10.0.0.1:8000",
        "/dev/ttyUSB0");
    Robot r = new Robot(b, 1);
    r.forward(50, 2);
    r.turnLeft(25, 2);
    r.backward(50, 1);
    r.forward(100, 1);
    r.stop();
} catch (CommunicationException e) {
    System.out.println("Ups...");
    e.printStackTrace();
}
/* ... */
```

Programando en Android

Método de comunicación

- Acceso a Internet o a la LAN del servidor.
- HTTP y JSON.
- Peticiones del cliente en un HTTP POST.
- Resultado del comando en la respuesta.

Programando en Android

¿Por qué esos protocolos?

- Bluetooth (IEEE 802.15.1) para proyectos similares.
- El N6 usa ZigBee (basado en IEEE 802.15.4).
- HTTP tiene mucho overhead pero...
 - Librerías para hacer clientes en cualquier lenguaje.
 - Servidor en Python BaseHTTPRequestHandler.
 - Mensajes JSON en texto plano.
 - HTTP + JSON → Javascript.
 - Posibilidad a futuro de robots remotos.

Programando en Android

¿Por qué esos protocolos?

- Bluetooth (IEEE 802.15.1) para proyectos similares.
- El N6 usa ZigBee (basado en IEEE 802.15.4).
- HTTP tiene mucho overhead pero...
 - Librerías para hacer clientes en cualquier lenguaje.
 - Servidor en Python BaseHTTPRequestHandler.
 - Mensajes JSON en texto plano.
 - HTTP + JSON → Javascript.
 - Posibilidad a futuro de robots remotos.

Programando en Android

¿Por qué esos protocolos?

- Bluetooth (IEEE 802.15.1) para proyectos similares.
- El N6 usa ZigBee (basado en IEEE 802.15.4).
- HTTP tiene mucho overhead pero...
 - Librerías para hacer clientes en cualquier lenguaje.
 - Servidor en Python BaseHTTPRequestHandler.
 - Mensajes JSON en texto plano.
 - HTTP + JSON → Javascript.
 - Posibilidad a futuro de robots remotos.

Programando en Android

¿Por qué esos protocolos?

- Bluetooth (IEEE 802.15.1) para proyectos similares.
- El N6 usa ZigBee (basado en IEEE 802.15.4).
- HTTP tiene mucho overhead pero...
 - Librerías para hacer clientes en cualquier lenguaje.
 - Servidor en Python BaseHTTPRequestHandler.
 - Mensajes JSON en texto plano.
 - HTTP + JSON → Javascript.
 - Posibilidad a futuro de robots remotos.

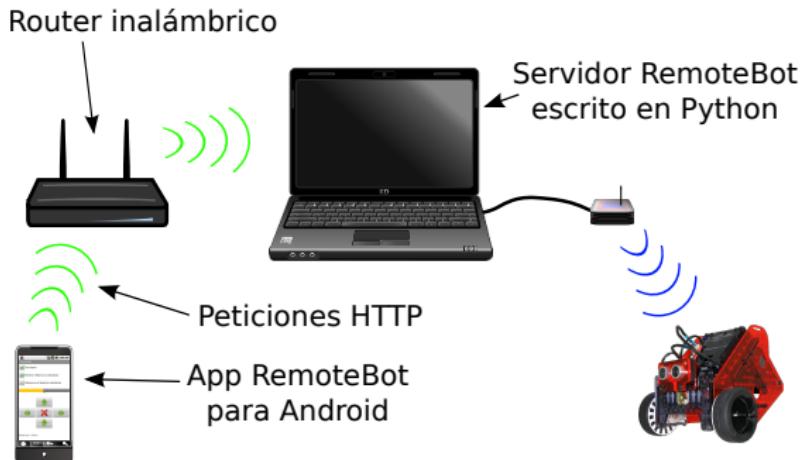
Programando en Android

¿Por qué esos protocolos?

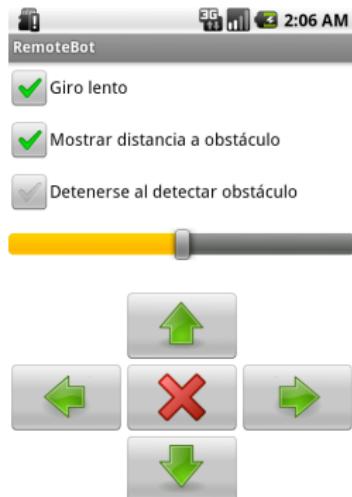
- Bluetooth (IEEE 802.15.1) para proyectos similares.
- El N6 usa ZigBee (basado en IEEE 802.15.4).
- HTTP tiene mucho overhead pero...
 - Librerías para hacer clientes en cualquier lenguaje.
 - Servidor en Python BaseHTTPRequestHandler.
 - Mensajes JSON en texto plano.
 - HTTP + JSON → Javascript.
 - Posibilidad a futuro de robots remotos.

Programando en Android

Arquitectura de Remotebot



Demo



Obstáculo a 100cm

Programando en Android

Protocolo en JSON

- 3 entidades agrupan métodos y funciones:
 - robot
 - board
 - module
- 3 tipos de mensajes:
 - Peticiones del cliente.
 - Valores retornados.
 - Excepciones con stacktrace.

Programando en Android

Ejemplo del protocolo

Para instanciar una Board un cliente remotebot envía mensaje JSON:

```
[ {  
    "target": "board",  
    "board": {"device": "/dev/ttyUSB0"},  
    "command": "__init__"  
}]
```

Si el dispositivo no existiera se produciría una excepción en Python y la respuesta del servidor sería:

```
{  
    "type": "exception",  
    "name": "SerialException(u\"could not open port /dev/  
        ttyUSB0...\")",  
    "stacktrace": "Traceback (most recent call last):..."  
}
```

Para leer el sensor de distancia “ping” se usa el mensaje:

```
[{
  "target": "robot",
  "board": { "device": "/dev/ttyUSB0" },
  "id": 1,
  "command": "ping",
  "args": []
}]
```

El resultado tiene la forma:

```
{
  "type": "returnvalues",
  "values": [100]
}
```

¡Gracias a la UNCa y a la AESI por la
invitación!

Página proyecto: <http://robots.linti.unlp.edu.ar>

Repositorio duinobot: <http://lihuen.linti.unlp.edu.ar>

Blog de software libre: <http://sl.linti.unlp.edu.ar>

Código remotebot: <https://github.com/fernandolopez>

Contacto: soportelihuen [AT] linti.unlp.edu.ar

Twitter: [@SL_Informatica](https://twitter.com/@SL_Informatica)

Facebook: Lihuen GNU/Linux