

XRemoteBot

Un servicio para programar robots en forma remota



Fernando López

Facultad de Informática
Universidad Nacional de La Plata

2015

Indice

Introducción

- Producto de la tesina

- Motivación

- Control de robots a través de Internet

XRemoteBot

- Diseño

- Implementación

Ejemplos

- Movimientos básicos

- Esquivar obstáculos

- Valores de los sensores en la vista Web

- Control remoto

Trabajos futuros

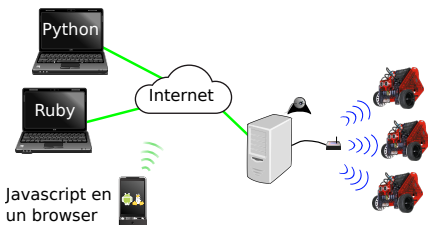
Producto de la tesina

- XRemoteBot
- Robots reales de forma remota
- Programación

The screenshot shows the XRemoteBot web interface. At the top, there are navigation tabs for 'XRemoteBot', 'Javascript', 'Documentación', 'Python', and 'Ruby'. The main content area is divided into three sections:

- Section A:** A code editor containing JavaScript code for interacting with a robot server. The code includes functions for getting robots, fetching a robot, and controlling its movement (forward, turn, backward) and logging.
- Section B:** A terminal window showing the output of the code execution, including log messages and JSON responses from the server.
- Section C:** A live camera feed showing a small red robot in a white enclosure.

Additional elements include a 'Ejecutar' (Execute) button, a 'Habilitar modo debug' (Enable debug mode) checkbox, and a note: 'Todos los métodos de los robots usan Promesas, agrégué a usar la función en la documentación'.



Motivación

Proyecto

- Desde el año 2009 proyecto “Programando con robots y software libre”
- Parallax Scribbler (2009)
- Múltiplo N6 de RobotGroup (2011)
- Proyecto con YPF y Dirección de Escuelas Técnicas de la Provincia de Buenos Aires

Motivación

Proyecto

- Desde el año 2009 proyecto “Programando con robots y software libre”
- Parallax Scribbler (2009)
- Múltiplo N6 de RobotGroup (2011)
- Proyecto con YPF y Dirección de Escuelas Técnicas de la Provincia de Buenos Aires

Motivación

Proyecto

- Desde el año 2009 proyecto “Programando con robots y software libre”
- Parallax Scribbler (2009)
- Multiplo N6 de RobotGroup (2011)
- Proyecto con YPF y Dirección de Escuelas Técnicas de la Provincia de Buenos Aires

Motivación

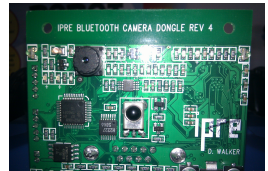
Proyecto

- Desde el año 2009 proyecto “Programando con robots y software libre”
- Parallax Scribbler (2009)
- Multiplo N6 de RobotGroup (2011)
- Proyecto con YPF y Dirección de Escuelas Técnicas de la Provincia de Buenos Aires

Motivación

Scribblers

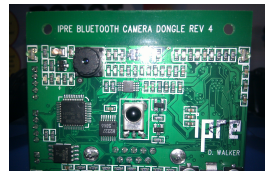
- Origen
- Biblioteca Myro
- Fluke
- Sensores
- Cómo los usamos
- Problemas:
 - Reconexión
 - Pilas
 - Compra



Motivación

Scribblers

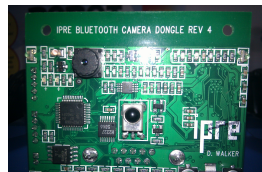
- Origen
- Biblioteca Myro
- Fluke
- Sensores
- Cómo los usamos
- Problemas:
 - Reconexión
 - Pilas
 - Compra



Motivación

Scribblers

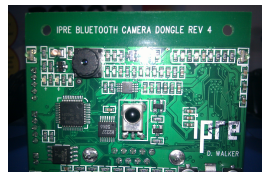
- Origen
- Biblioteca Myro
- Fluke
- Sensores
- Cómo los usamos
- Problemas:
 - Reconexión
 - Pilas
 - Compra



Motivación

Scribblers

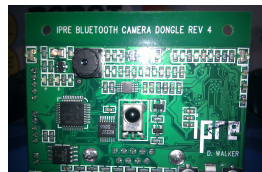
- Origen
- Biblioteca Myro
- Fluke
- Sensores
- Cómo los usamos
- Problemas:
 - Reconexión
 - Pilas
 - Compra



Motivación

Scribblers

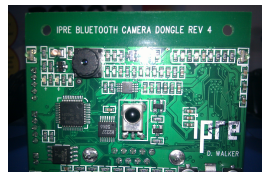
- Origen
- Biblioteca Myro
- Fluke
- Sensores
- Cómo los usamos
- Problemas:
 - Reconexión
 - Pilas
 - Compra



Motivación

Scribblers

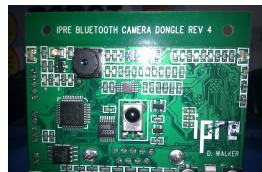
- Origen
- Biblioteca Myro
- Fluke
- Sensores
- Cómo los usamos
- Problemas:
 - Reconexión
 - Pilas
 - Compra



Motivación

Scribblers

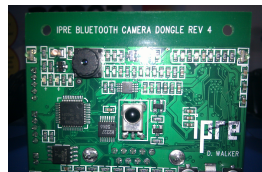
- Origen
- Biblioteca Myro
- Fluke
- Sensores
- Cómo los usamos
- Problemas:
 - Reconexión
 - Pilas
 - Compra



Motivación

Scribblers

- Origen
- Biblioteca Myro
- Fluke
- Sensores
- Cómo los usamos
- Problemas:
 - Reconexión
 - Pilas
 - Compra



Motivación

Scribblers

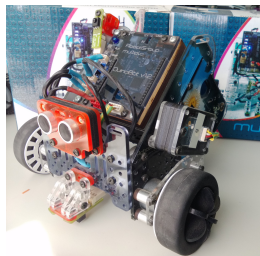
- Origen
- Biblioteca Myro
- Fluke
- Sensores
- Cómo los usamos
- Problemas:
 - Reconexión
 - Pilas
 - Compra



Motivación

Robots N6

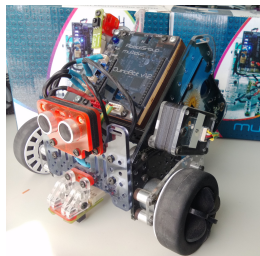
- Origen
- Biblioteca DuinoBot
- XBee
- Sensores
- Cómo los usamos
- Problemas:
 - Costos: XBee
 - Pilas



Motivación

Robots N6

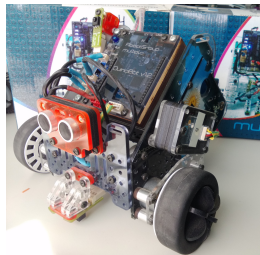
- Origen
- Biblioteca DuinoBot
- XBee
- Sensores
- Cómo los usamos
- Problemas:
 - Costos: XBee
 - Pilas



Motivación

Robots N6

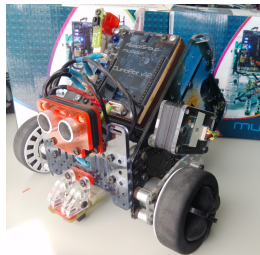
- Origen
- Biblioteca DuinoBot
- XBee
- Sensores
- Cómo los usamos
- Problemas:
 - Costos: XBee
 - Pilas



Motivación

Robots N6

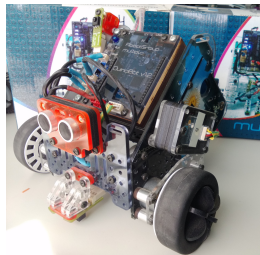
- Origen
- Biblioteca DuinoBot
- XBee
- Sensores
- Cómo los usamos
- Problemas:
 - Costos: XBee
 - Pilas



Motivación

Robots N6

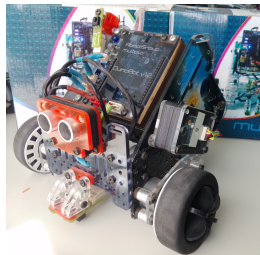
- Origen
- Biblioteca DuinoBot
- XBee
- Sensores
- Cómo los usamos
- Problemas:
 - Costos: XBee
 - Pilas



Motivación

Robots N6

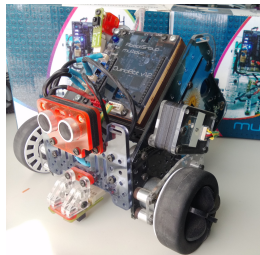
- Origen
- Biblioteca DuinoBot
- XBee
- Sensores
- Cómo los usamos
- Problemas:
 - Costos: XBee
 - Pilas



Motivación

Robots N6

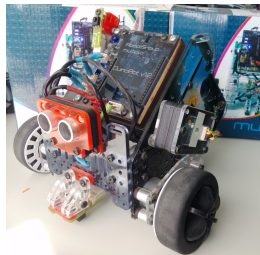
- Origen
- Biblioteca DuinoBot
- XBee
- Sensores
- Cómo los usamos
- Problemas:
 - Costos: XBee
 - Pilas



Motivación

Robots N6

- Origen
- Biblioteca DuinoBot
- XBee
- Sensores
- Cómo los usamos
- Problemas:
 - Costos: XBee
 - Pilas



Motivación

Trabajo final RemoteBot

- Laboratorio de software (2012)
- Aplicación Android en Java
 - GUI
 - Uso de sensores
 - Uso de red
- Incompatibilidad XBee
- Necesidad de un puente entre dispositivos:
 - Python
 - SimpleHTTPServer
 - DuinoBot



Motivación

Trabajo final RemoteBot

- Laboratorio de software (2012)
- Aplicación Android en Java
 - GUI
 - Uso de sensores
 - Uso de red
- Incompatibilidad XBee
- Necesidad de un puente entre dispositivos:
 - Python
 - SimpleHTTPServer
 - DuinoBot



Motivación

Trabajo final RemoteBot

- Laboratorio de software (2012)
- Aplicación Android en Java
 - GUI
 - Uso de sensores
 - Uso de red
- Incompatibilidad XBee
- Necesidad de un puente entre dispositivos:
 - Python
 - SimpleHTTPServer
 - DuinoBot



Motivación

Trabajo final RemoteBot

- Laboratorio de software (2012)
- Aplicación Android en Java
 - GUI
 - Uso de sensores
 - Uso de red
- Incompatibilidad XBee
- Necesidad de un puente entre dispositivos:
 - Python
 - SimpleHTTPServer
 - DuinoBot



Motivación

Trabajo final RemoteBot

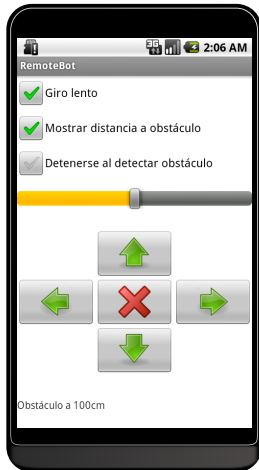
- Laboratorio de software (2012)
- Aplicación Android en Java
 - GUI
 - Uso de sensores
 - Uso de red
- Incompatibilidad XBee
- Necesidad de un puente entre dispositivos:
 - Python
 - SimpleHTTPServer
 - DuinoBot



Motivación

Trabajo final RemoteBot

- Laboratorio de software (2012)
- Aplicación Android en Java
 - GUI
 - Uso de sensores
 - Uso de red
- Incompatibilidad XBee
- Necesidad de un puente entre dispositivos:
 - Python
 - SimpleHTTPServer
 - DuinoBot



Motivación

Trabajo final RemoteBot

- Laboratorio de software (2012)
- Aplicación Android en Java
 - GUI
 - Uso de sensores
 - Uso de red
- Incompatibilidad XBee
- Necesidad de un puente entre dispositivos:
 - Python
 - SimpleHTTPServer
 - DuinoBot



Motivación

Trabajo final RemoteBot

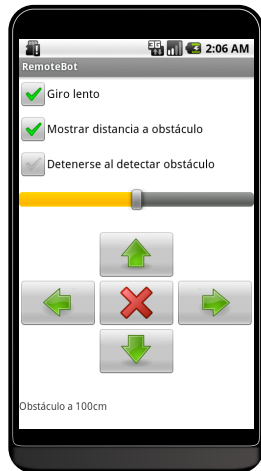
- Laboratorio de software (2012)
- Aplicación Android en Java
 - GUI
 - Uso de sensores
 - Uso de red
- Incompatibilidad XBee
- Necesidad de un puente entre dispositivos:
 - Python
 - SimpleHTTPServer
 - DuinoBot



Motivación

Trabajo final RemoteBot

- Laboratorio de software (2012)
- Aplicación Android en Java
 - GUI
 - Uso de sensores
 - Uso de red
- Incompatibilidad XBee
- Necesidad de un puente entre dispositivos:
 - Python
 - SimpleHTTPServer
 - DuinoBot



Motivación

Trabajo final RemoteBot

- Laboratorio de software (2012)
- Aplicación Android en Java
 - GUI
 - Uso de sensores
 - Uso de red
- Incompatibilidad XBee
- Necesidad de un puente entre dispositivos:
 - Python
 - SimpleHTTPServer
 - DuinoBot



Motivación

Experiencia posterior de la materia

- JDance: Otros alumnos intectuando con mi servidor
- Uso de toda la API JSON expuesta (métodos bloqueantes)
- Acceso concurrente

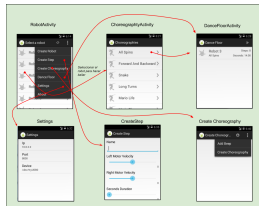


Figure: Una de las implementaciones de JDance

Fuente:

<https://github.com/mclo/JDance>

Motivación

Experiencia posterior de la materia

- JDance: Otros alumnos intectuando con mi servidor
- Uso de toda la API JSON expuesta (métodos bloqueantes)
- Acceso concurrente

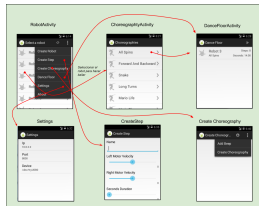


Figure: Una de las implementaciones de JDance

Fuente:

<https://github.com/mclo/JDance>

Motivación

Experiencia posterior de la materia

- JDance: Otros alumnos intectuando con mi servidor
- Uso de toda la API JSON expuesta (métodos bloqueantes)
- Acceso concurrente

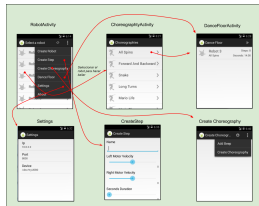


Figure: Una de las implementaciones de JDance

Fuente:

<https://github.com/mclo/JDance>

Motivación

Casos de uso posibles para una versión extendida

- Acceso de los alumnos a los robots desde sus casas
- Compartir un solo XBee entre varios robots
- Permitir el uso de los robots en dispositivos móviles

Control de robots a través de Internet

Posibilidades

- Experimentar sin miedo desde el hogar
- Afrontar los problemas de controlar un dispositivo real
- Acercar el proyecto a más gente
- Permitir el uso de otros lenguajes
- Otro enfoque posible: Ahorrar placas XBee
- Soportar otros robots con la misma biblioteca

Control de robots a través de Internet

Posibilidades

- Experimentar sin miedo desde el hogar
- Afrontar los problemas de controlar un dispositivo real
- Acercar el proyecto a más gente
- Permitir el uso de otros lenguajes
- Otro enfoque posible: Ahorrar placas XBee
- Soportar otros robots con la misma biblioteca

Control de robots a través de Internet

Posibilidades

- Experimentar sin miedo desde el hogar
- Afrontar los problemas de controlar un dispositivo real
- Acercar el proyecto a más gente
- Permitir el uso de otros lenguajes
- Otro enfoque posible: Ahorrar placas XBee
- Soportar otros robots con la misma biblioteca

Control de robots a través de Internet

Posibilidades

- Experimentar sin miedo desde el hogar
- Afrontar los problemas de controlar un dispositivo real
- Acercar el proyecto a más gente
- Permitir el uso de otros lenguajes
- Otro enfoque posible: Ahorrar placas XBee
- Soportar otros robots con la misma biblioteca

Control de robots a través de Internet

Posibilidades

- Experimentar sin miedo desde el hogar
- Afrontar los problemas de controlar un dispositivo real
- Acercar el proyecto a más gente
- Permitir el uso de otros lenguajes
- Otro enfoque posible: Ahorrar placas XBee
- Soportar otros robots con la misma biblioteca

Control de robots a través de Internet

Posibilidades

- Experimentar sin miedo desde el hogar
- Afrontar los problemas de controlar un dispositivo real
- Acercar el proyecto a más gente
- Permitir el uso de otros lenguajes
- Otro enfoque posible: Ahorrar placas XBee
- Soportar otros robots con la misma biblioteca

Control de robots a través de Internet

Opciones existentes de control de robots de forma remota

- Control de dispositivos:
 - VCar
 - Tele Toyland
- APIs y plataformas:
 - Educabot
 - Gobot, Cylon y Artoo (The Hybrid Group)
 - cppy-io

Control de robots a través de Internet

Opciones existentes de control de robots de forma remota

- Control de dispositivos:
 - VCar
 - Tele Toyland
- APIs y plataformas:
 - Educabot
 - Gobot, Cylon y Artoo (The Hybrid Group)
 - cppy-io

Control de robots a través de Internet

Opciones existentes de control de robots de forma remota

- Control de dispositivos:
 - VCar
 - Tele Toyland
- APIs y plataformas:
 - Educabot
 - Gobot, Cylon y Artoo (The Hybrid Group)
 - cppy-io

Control de robots a través de Internet

Opciones existentes de control de robots de forma remota

- Control de dispositivos:
 - VCar
 - Tele Toyland
- APIs y plataformas:
 - Educabot
 - Gobot, Cylon y Artoo (The Hybrid Group)
 - cPPP-io

Control de robots a través de Internet

Opciones existentes de control de robots de forma remota

- Control de dispositivos:
 - VCar
 - Tele Toyland
- APIs y plataformas:
 - Educabot
 - Gobot, Cylon y Artoo (The Hybrid Group)
 - cPPP-io

Control de robots a través de Internet

Opciones existentes de control de robots de forma remota

- Control de dispositivos:
 - VCar
 - Tele Toyland
- APIs y plataformas:
 - Educabot
 - Gobot, Cylon y Artoo (The Hybrid Group)
 - cppp-io

Control de robots a través de Internet

Opciones existentes de control de robots de forma remota

- Control de dispositivos:
 - VCar
 - Tele Toyland
- APIs y plataformas:
 - Educabot
 - Gobot, Cylon y Artoo (The Hybrid Group)
 - cPPP-io

Diseño

XRemoteBot

- Nombre XRemoteBot
- Inspirado en RemoteBot
- Reescrito
- Cliente-Servidor

Diseño

XRemoteBot

- Nombre XRemoteBot
- Inspirado en RemoteBot
- Reescrito
- Cliente-Servidor

Diseño

XRemoteBot

- Nombre XRemoteBot
- Inspirado en RemoteBot
- Reescrito
- Cliente-Servidor

Diseño

XRemoteBot

- Nombre XRemoteBot
- Inspirado en RemoteBot
- Reescrito
- Cliente-Servidor

Diseño

Planteo de un servidor nuevo

- 2 modos de operación
- Variable `public_server`:

True: Expone una copia de los datos
Manda los datos como "reservados"
Permite para ser visto a través de internet

False: No expone información
Cualquier usuario puede usar cualquier robot
Permite para ser usado en un aula

Diseño

Planteo de un servidor nuevo

- 2 modos de operación
- Variable `public_server`:
 - `True`:
 - Requiere autenticación de los clientes
 - Marca los robots como "reservados"
 - Pensado para ser usado a través de Internet
 - `False`:
 - No requiere autenticación
 - Cualquier usuario puede usar cualquier robot
 - Pensado para ser usado en un aula

Diseño

Planteo de un servidor nuevo

- 2 modos de operación
- Variable `public_server`:
 - True:**
 - Requiere autenticación de los clientes
 - Marca los robots como “reservados”
 - Pensado para ser usado a través de Internet
 - False**
 - No requiere autenticación
 - Cualquier usuario puede usar cualquier robot
 - Pensado para ser usado en un aula

Diseño

Planteo de un servidor nuevo

- 2 modos de operación
- Variable `public_server`:
 - True:**
 - Requiere autenticación de los clientes
 - Marca los robots como “reservados”
 - Pensado para ser usado a través de Internet
 - False**
 - No requiere autenticación
 - Cualquier usuario puede usar cualquier robot
 - Pensado para ser usado en un aula

Diseño

Planteo de un servidor nuevo

- 2 modos de operación
- Variable `public_server`:
 - True:**
 - Requiere autenticación de los clientes
 - Marca los robots como “reservados”
 - Pensado para ser usado a través de Internet
 - False**
 - No requiere autenticación
 - Cualquier usuario puede usar cualquier robot
 - Pensado para ser usado en un aula

Diseño

Planteo de un servidor nuevo

- 2 modos de operación
- Variable `public_server`:
 - True:**
 - Requiere autenticación de los clientes
 - Marca los robots como “reservados”
 - Pensado para ser usado a través de Internet
 - False**
 - No requiere autenticación
 - Cualquier usuario puede usar cualquier robot
 - Pensado para ser usado en un aula

Diseño

Planteo de un servidor nuevo

- 2 modos de operación
- Variable `public_server`:
 - True:**
 - Requiere autenticación de los clientes
 - Marca los robots como “reservados”
 - Pensado para ser usado a través de Internet
 - False**
 - No requiere autenticación
 - Cualquier usuario puede usar cualquier robot
 - Pensado para ser usado en un aula

Diseño

Planteo de un servidor nuevo

- 2 modos de operación
- Variable `public_server`:
 - True:**
 - Requiere autenticación de los clientes
 - Marca los robots como “reservados”
 - Pensado para ser usado a través de Internet
 - False**
 - No requiere autenticación
 - Cualquier usuario puede usar cualquier robot
 - Pensado para ser usado en un aula

Diseño

Planteo de un servidor nuevo

- 2 modos de operación
- Variable `public_server`:
 - True:**
 - Requiere autenticación de los clientes
 - Marca los robots como “reservados”
 - Pensado para ser usado a través de Internet
 - False**
 - No requiere autenticación
 - Cualquier usuario puede usar cualquier robot
 - Pensado para ser usado en un aula

Diseño

Planteo de un servidor nuevo

- 2 modos de operación
- Variable `public_server`:
 - True:**
 - Requiere autenticación de los clientes
 - Marca los robots como “reservados”
 - Pensado para ser usado a través de Internet
 - False**
 - No requiere autenticación
 - Cualquier usuario puede usar cualquier robot
 - Pensado para ser usado en un aula

Diseño

Planteo de un nuevo protocolo

- Transporte: Websockets
- Análisis de JSON, CBOR y BSON:
 - Velocidad
 - Bytes por mensaje
 - Sin diferencias significativas en Python
 - Diferencias en la velocidad en Javascript
 - Se prioriza el soporte para el browser: JSON

Diseño

Planteo de un nuevo protocolo

- Transporte: Websockets
- Análisis de JSON, CBOR y BSON:
 - Velocidad
 - Bytes por mensaje
 - Sin diferencias significativas en Python
 - Diferencias en la velocidad en Javascript
 - Se prioriza el soporte para el browser: JSON

Diseño

Planteo de un nuevo protocolo

- Transporte: Websockets
- Análisis de JSON, CBOR y BSON:
 - Velocidad
 - Bytes por mensaje
 - Sin diferencias significativas en Python
 - Diferencias en la velocidad en Javascript
 - Se prioriza el soporte para el browser: JSON

Diseño

Planteo de un nuevo protocolo

- Transporte: Websockets
- Análisis de JSON, CBOR y BSON:
 - Velocidad
 - Bytes por mensaje
 - Sin diferencias significativas en Python
 - Diferencias en la velocidad en Javascript
 - Se prioriza el soporte para el browser: JSON

Diseño

Planteo de un nuevo protocolo

- Transporte: Websockets
- Análisis de JSON, CBOR y BSON:
 - Velocidad
 - Bytes por mensaje
 - Sin diferencias significativas en Python
 - Diferencias en la velocidad en Javascript
 - Se prioriza el soporte para el browser: JSON

Diseño

Planteo de un nuevo protocolo

- Transporte: Websockets
- Análisis de JSON, CBOR y BSON:
 - Velocidad
 - Bytes por mensaje
 - Sin diferencias significativas en Python
 - Diferencias en la velocidad en Javascript
 - Se prioriza el soporte para el browser: JSON

Diseño

Planteo de un nuevo protocolo

- Transporte: Websockets
- Análisis de JSON, CBOR y BSON:
 - Velocidad
 - Bytes por mensaje
 - Sin diferencias significativas en Python
 - Diferencias en la velocidad en Javascript
 - Se prioriza el soporte para el browser: JSON

Diseño

API JSON

- Protocolo simple y extensible
- Mensajes del cliente:
 - `entity`
 - `method`
 - `args`
- Respuestas del servidor:
 - `response`
 - `value`
 - `message`
- `msg_id`: Opcional

Diseño

API JSON

- Protocolo simple y extensible
- Mensajes del cliente:
 - `entity`
 - `method`
 - `args`
- Respuestas del servidor:
 - `response`
 - `value`
 - `message`
- `msg_id`: Opcional

Diseño

API JSON

- Protocolo simple y extensible
- Mensajes del cliente:
 - `entity`
 - `method`
 - `args`
- Respuestas del servidor:
 - `response`
 - `value`
 - `message`
- `msg_id`: Opcional

Diseño

API JSON

- Protocolo simple y extensible
- Mensajes del cliente:
 - `entity`
 - `method`
 - `args`
- Respuestas del servidor:
 - `response`
 - `value`
 - `message`
- `msg_id`: Opcional

Diseño

API JSON

- Protocolo simple y extensible
- Mensajes del cliente:
 - `entity`
 - `method`
 - `args`
- Respuestas del servidor:
 - `response`
 - `value`
 - `message`
- `msg_id`: Opcional

Diseño

API JSON

- Protocolo simple y extensible
- Mensajes del cliente:
 - `entity`
 - `method`
 - `args`
- Respuestas del servidor:
 - `response`
 - `value`
 - `message`
- `msg_id`: Opcional

Diseño

API JSON

- Protocolo simple y extensible
- Mensajes del cliente:
 - `entity`
 - `method`
 - `args`
- Respuestas del servidor:
 - `response`
 - `value`
 - `message`
- `msg_id`: Opcional

Diseño

API JSON

- Protocolo simple y extensible
- Mensajes del cliente:
 - `entity`
 - `method`
 - `args`
- Respuestas del servidor:
 - `response`
 - `value`
 - `message`
- `msg_id`: Opcional

Diseño

API JSON

- Protocolo simple y extensible
- Mensajes del cliente:
 - entity
 - method
 - args
- Respuestas del servidor:
 - response
 - value
 - message
- msg_id: Opcional

Diseño

API JSON

- Protocolo simple y extensible
- Mensajes del cliente:
 - entity
 - method
 - args
- Respuestas del servidor:
 - response
 - value
 - message
- msg_id: Opcional

Diseño

Características comunes entre el N6 y el Scribblor

- Posibilidades de movimiento
- Sensores principales
- Lenguaje de las bibliotecas
- Interfaz de programación similar
- Módulos de Python específicos para cada uno
- Posibilidad de unificar las APIs

Diseño

Características comunes entre el N6 y el Scribblor

- Posibilidades de movimiento
- Sensores principales
- Lenguaje de las bibliotecas
- Interfaz de programación similar
- Módulos de Python específicos para cada uno
- Posibilidad de unificar las APIs

Diseño

Características comunes entre el N6 y el Scribblor

- Posibilidades de movimiento
- Sensores principales
- Lenguaje de las bibliotecas
- Interfaz de programación similar
- Módulos de Python específicos para cada uno
- Posibilidad de unificar las APIs

Diseño

Características comunes entre el N6 y el Scribblor

- Posibilidades de movimiento
- Sensores principales
- Lenguaje de las bibliotecas
- Interfaz de programación similar
- Módulos de Python específicos para cada uno
- Posibilidad de unificar las APIs

Diseño

Características comunes entre el N6 y el Scribbler

- Posibilidades de movimiento
- Sensores principales
- Lenguaje de las bibliotecas
- Interfaz de programación similar
- Módulos de Python específicos para cada uno
- Posibilidad de unificar las APIs

Diseño

Características comunes entre el N6 y el Scribbler

- Posibilidades de movimiento
- Sensores principales
- Lenguaje de las bibliotecas
- Interfaz de programación similar
- Módulos de Python específicos para cada uno
- Posibilidad de unificar las APIs

Implementación

Características de XRemoteBot

- Soporte de robots Scribblers y Multiplo con la misma API
- Sistema de reservas de los robots
- Servidor con soporte de concurrencia
- Extensible a otros robots (u otros dispositivos)
- Protocolo accesible desde Javascript
- Selección de tecnologías que habilita el uso de Javascript desde un navegador
- Clientes simples en general

Implementación

Características de XRemoteBot

- Soporte de robots Scribblers y Multiplo con la misma API
- Sistema de reservas de los robots
- Servidor con soporte de concurrencia
- Extensible a otros robots (u otros dispositivos)
- Protocolo accesible desde Javascript
- Selección de tecnologías que habilita el uso de Javascript desde un navegador
- Clientes simples en general

Implementación

Características de XRemoteBot

- Soporte de robots Scribblers y Multiplo con la misma API
- Sistema de reservas de los robots
- Servidor con soporte de concurrencia
- Extensible a otros robots (u otros dispositivos)
- Protocolo accesible desde Javascript
- Selección de tecnologías que habilita el uso de Javascript desde un navegador
- Clientes simples en general

Implementación

Características de XRemoteBot

- Soporte de robots Scribblers y Multiplo con la misma API
- Sistema de reservas de los robots
- Servidor con soporte de concurrencia
- Extensible a otros robots (u otros dispositivos)
- Protocolo accesible desde Javascript
- Selección de tecnologías que habilita el uso de Javascript desde un navegador
- Clientes simples en general

Implementación

Características de XRemoteBot

- Soporte de robots Scribblers y Multiplo con la misma API
- Sistema de reservas de los robots
- Servidor con soporte de concurrencia
- Extensible a otros robots (u otros dispositivos)
- Protocolo accesible desde Javascript
- Selección de tecnologías que habilita el uso de Javascript desde un navegador
- Clientes simples en general

Implementación

Características de XRemoteBot

- Soporte de robots Scribblers y Multiplo con la misma API
- Sistema de reservas de los robots
- Servidor con soporte de concurrencia
- Extensible a otros robots (u otros dispositivos)
- Protocolo accesible desde Javascript
- Selección de tecnologías que habilita el uso de Javascript desde un navegador
- Clientes simples en general

Implementación

Características de XRemoteBot

- Soporte de robots Scribblers y Multiplo con la misma API
- Sistema de reservas de los robots
- Servidor con soporte de concurrencia
- Extensible a otros robots (u otros dispositivos)
- Protocolo accesible desde Javascript
- Selección de tecnologías que habilita el uso de Javascript desde un navegador
- Clientes simples en general

Instalación actual



Implementación

Clientes de XRemoteBot

- Python
- Ruby
- Javascript
 - API de WebSockets
 - Promises
- Interfaz Web ([video](#))

Implementación

Clientes de XRemoteBot

- Python
- Ruby
- Javascript
 - API de WebSockets
 - Promises
- Interfaz Web ([video](#))

Implementación

Clientes de XRemoteBot

- Python
- Ruby
- Javascript
 - API de WebSockets
 - Promises
- Interfaz Web ([video](#))

Implementación

Cientes de XRemoteBot

- Python
- Ruby
- Javascript
 - API de WebSockets
 - Promises
- Interfaz Web ([video](#))

Implementación

Clientes de XRemoteBot

- Python
- Ruby
- Javascript
 - API de WebSockets
 - Promises
- Interfaz Web ([video](#))

Implementación

Cientes de XRemoteBot

- Python
- Ruby
- Javascript
 - API de WebSockets
 - Promises
- Interfaz Web ([video](#))

Ejemplos

Movimientos básicos

Listing 1: Python

```
#!/usr/bin/env python
import xremotebot

key = '8628cd03-3d4e-4465-aa3d-7
b6352f26279'

server = xremotebot.Server('ws
://163.10.10.85:8000/api', key)

robot = xremotebot.Robot(server,
server.reserve('n6', 3))

for i in range(3):
    robot.forward(50, 1)
    robot.turnLeft(20, 1.25)
```

Listing 2: Ruby

```
#!/usr/bin/env ruby
require 'xremotebot'
key = '8628cd03-3d4e-4465-aa3d-7
b6352f26279'

server = XRemoteBot::Server.new('
163.10.10.85', 8000, 'api', key)

robot = XRemoteBot::Robot.new(server,
server.reserve('n6', 3))

4.times do
    robot.forward(50, 1)
    robot.turnLeft(20, 1.25)
end
```

Ejemplos

Movimientos básicos

Listing 3: Javascript

```
var key = '8628cd03-3d4e-4465-aa3d-7b6352f26279';
var server = new Server('ws://163.10.10.85:8000/api', key);
server.onConnect(function(){
    server.reserve('n6', 3).then(function(robot_obj){
        var robot = new Robot(server, robot_obj);
        for (var i = 0; i < 4; i++){
            robot.forward(50, 1);
            robot.turnLeft(20, 1.25);
        }
    });
});
```


Ejemplos

Esquivar obstáculos

Listing 4: Python

```
#!/usr/bin/env python
import xremotobot
from random import random
key = '8628cd03-3d4e-4465-aa3d-7
      b6352f26279'

server = xremotobot.Server('ws
    ://163.10.10.85:8000/api', key)

robot = xremotobot.Robot(server,
    server.reserve('n6', 3))

robot.forward(40)
while True:
    if robot.ping() < 15:
        robot.backward(30, 1)
    if random() < 0.5:
        robot.turnLeft(30, 1)
    else:
        robot.turnRight(30, 1)
    robot.forward(40)
```

Listing 5: Ruby

```
#!/usr/bin/env ruby
require 'xremotobot'
key = '8628cd03-3d4e-4465-aa3d-7
      b6352f26279'

server = XRemoteBot::Server.new('
    163.10.10.85', 8000, 'api', key)

robot = XRemoteBot::Robot.new(server,
    server.reserve('n6', 3))

robot.forward(40)
loop do
    if robot.ping < 15
        robot.backward(30, 1)
        if rand < 0.5
            robot.turnLeft(30, 1)
        else
            robot.turnRight(30, 1)
        end
        robot.forward(40)
    end
end
```

Ejemplos

Esquivar obstáculos

Listing 6: Javascript

```
var key = '8628cd03-3d4e-4465-aa3d-7b6352f26279';
var server = new Server('ws://163.10.10.85:8000/api', key);
server.onConnect(function(){
  server.reserve('n6', 3).then(function(robot_obj){
    var robot = new Robot(server, robot_obj);

    function esquivar(){
      robot.ping().then(function(ping){
        if (ping < 15){
          robot.backward(30, 1);
          if (Math.random() < 0.5){
            robot.turnLeft(30, 1);
          }
          else{
            robot.turnRight(30, 1);
          }
          robot.forward(40).then(function(){
            setTimeout(esquivar, 200);
          });
        }
        else{
          setTimeout(esquivar, 200);
        }
      });
    }
  });
});
```

Ejemplos

Sensores

Listing 7: Javascript

```
var key = '8628cd03-3d4e-4465-aa3d-7b6352f26279';
var server = new Server('ws://163.10.10.85:8000/api', key);

var video = $('#videoCanvas');
if ($('#s_ping').length == 0){
    video.after('<div id="senses"><p>Sensores:<ul><li id="s_ping"></li>'
        + '<li id="s_line"></li>'
        + '<li id="s_obstacle"></li></div>');
}

server.onConnect(function(){
    server.reserve('n6', 3).then(function(robot_obj){
        var robot = new Robot(server, robot_obj);
        function update(){
            Promise.all([
                robot.ping(),
                robot.getLine(),
                robot.getObstacle(),
            ])
        }
    })
})
```

([video](#))

Ejemplos

Control remoto con botones

Listing 8: Javascript

```
var server = new Server('ws://163.10.10.85:8000/api',
                        '8628cd03-3d4e-4465-aa3d-7b6352f26279');
server.onConnect(function(){
  server.reserve('n6', 3).then(function(robot_obj){
    var robot = new Robot(server, robot_obj);
    if ($('#izq').length == 0){
      $('#videoCanvas').after("<div>\
        <input id='izq' type='button' value='Izquierda'>\
        <input id='atr' type='button' value='Atr&aacute;s'>\
        <input id='ade' type='button' value='Adelante'>\
        <input id='der' type='button' value='Derecha'>\
        </div>");
      $('#izq').click(function(){ robot.turnLeft(50, 0.5); });
      $('#atr').click(function(){ robot.backward(100, 0.5); });
      $('#ade').click(function(){ robot.forward(100, 0.5); });
      $('#der').click(function(){ robot.turnRight(50, 0.5); });
    }
  });
});
```

(video)

Trabajos futuros

- Brython, Opal, Blockly
- Integración con moodle
- Almacenamiento de programas con la API LocalStorage o en el servidor
- Streaming: UStream, Youtube, etc...

Trabajos futuros

- Brython, Opal, Blockly
- Integración con moodle
- Almacenamiento de programas con la API LocalStorage o en el servidor
- Streaming: UStream, Youtube, etc...

Trabajos futuros

- Brython, Opal, Blockly
- Integración con moodle
- Almacenamiento de programas con la API LocalStorage o en el servidor
- Streaming: UStream, Youtube, etc...

Trabajos futuros

- Brython, Opal, Blockly
- Integración con moodle
- Almacenamiento de programas con la API LocalStorage o en el servidor
- Streaming: UStream, Youtube, etc...

Links

Servidor: <https://github.com/fernandolopez/xremotebot.git>

Clientes: <https://github.com/fernandolopez/xremotebot-clients.git>



Esta obra está licenciada bajo la Licencia Creative Commons Atribución 4.0 Internacional. Para ver una copia de esta licencia, visita <http://creativecommons.org/licenses/by/4.0/>.