

# DESARROLLO DE UNA PLATAFORMA EDUCATIVA PARA LA INVESTIGACIÓN EN FÚTBOL DE ROBOTS

Lic. Gonzalo Zabala

[gonzalo.zabala@vaneduc.edu.ar](mailto:gonzalo.zabala@vaneduc.edu.ar)

Ricardo Morán

[richi.moran@gmail.com](mailto:richi.moran@gmail.com)

Sebastián Blanco

[sebastiangabrielblanco@gmail.com](mailto:sebastiangabrielblanco@gmail.com)

Centro de Altos Estudios en Tecnología Informática

Facultad de Tecnología

Universidad Abierta Interamericana

TE / FAX: (5411) 43015240

## 1. Resumen:

*El objetivo de la investigación consiste en crear una plataforma educativa que brinde un contexto amigable para la enseñanza y potente para la investigación mediante un ambiente de objetos derivado de Smalltalk llamado "Squeak". Se utilizará como ejemplo el fútbol de robots (físico y simulado) debido a que vincula desafíos cognitivos inherentes a la robótica situada como captación del ambiente, autonomía, navegación, inteligencia artificial, comportamiento colaborativo y respuestas en tiempo real con el interés que genera una actividad lúdica.*

## 1. Abstract:

*The aim of this investigation is to create an educative platform that provides a friendly context for teaching and powerful for investigating using an environment of objects derived from Smalltalk called "Squeak". Physical and simulated robot soccer will be used as an example because it is related to different concepts such as environment capture, autonomy, navigation, artificial intelligence, collaborative behavior and real-time response combined with the interest that is generated by a recreational activity.*

## 2. Palabras clave:

Robótica Situada — Inteligencia Artificial — Squeak — Simulador RobotSoccer — Doraemon — Lego — Entorno — Estrategia — Proxy — Morphs

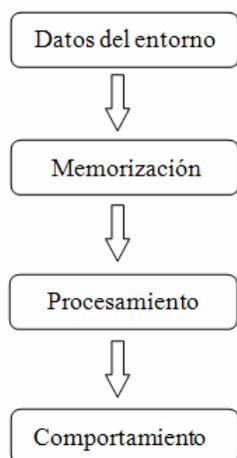
## 2. Keywords:

Situated Robotics — Artificial Intelligence — Squeak — RobotSoccer Simulator — Doraemon — Lego — Environment — Strategy — Proxy — Morphs

## 3. Introducción:

La robótica situada se caracteriza porque el robot toma los datos de un entorno y, en base a los mismos, toma decisiones y actúa consecuentemente en tiempo real utilizando una heurística eficaz y rápida.

Básicamente se puede resumir la idea en un esquema de la siguiente manera:



Dentro de la robótica situada, el **Fútbol de Robots** es la actividad que más investigadores ha congregado en todo el mundo porque, aparte de tener reglas conocidas y ser atractivo a nivel mundial, presenta todos los desafíos clásicos que este tipo de robótica contiene:

- Percepción del ambiente: ¿Cómo representar simbólicamente la realidad para actuar en consecuencia?
- Autonomía: ¿Cómo hacer que un robot tome decisiones sin estar constantemente indicándole qué hacer mediante un periférico?
- Navegación: ¿Cómo moverse hacia un determinado lugar de forma precisa y veloz considerando obstáculos (pathplanning)? ¿Cómo dirigirse hacia un objeto móvil con un comportamiento predecible o impredecible?
- Inteligencia artificial: ¿Cómo representar un comportamiento análogo al de un ser humano en una circunstancia determinada?
- Comportamiento colaborativo: ¿Cómo potenciar el comportamiento grupal para lograr un objetivo en base a la colaboración de cada agente individual?
- Respuestas en tiempo real: ¿Cómo responder en tiempo y forma a un entorno completamente dinámico?

Desde el punto de vista pedagógico, el fútbol de robots ofrece un marco para la enseñanza de la robótica y la programación sumamente atractivo ya que añade un elemento concreto a la abstracción propuesta por el desarrollo de software. Además, de por sí el fútbol y la competencia entre robots denotan una atracción lúdica apta para todo público.

Lamentablemente, los ambientes de desarrollo para fútbol de robots existentes adolecen de las siguientes limitaciones que ponen obstáculos al aprendizaje y dificultan la creación de un equipo eficiente:

- Implican conocimientos profundos de electrónica para realizar los robots.
- Implican conocimientos de lenguajes complejos para iniciarse en el mundo de la programación (como es el caso de C++).
- No permiten la modificación del comportamiento del equipo en tiempo real dificultando la evolución de la estrategia.
- No facilitan la visualización en tiempo real de los datos del partido dificultando el análisis con precisión del comportamiento del equipo.

Lo que se presenta en esta oportunidad es el desarrollo de un ambiente educativo (fusionando las plataformas Squeak y Lego) que brinda un contexto amigable para la enseñanza y potente para la investigación debido a que no sufre las desventajas antes citadas.

#### 4. Recursos utilizados

Cabe destacar que hay otras dificultades que van más allá del comportamiento de los robots pero son esenciales a la hora de tener un sistema de este tipo funcionando:

- Entorno de desarrollo apropiado para la implementación de una estrategia.

- Elección de un campo de acción y robots aptos para el mismo.
- Tecnología de comunicación para los robots.

El entorno en el cual se implementó el desarrollo es Squeak<sup>1</sup>, un ambiente de objetos derivado de Smalltalk<sup>2</sup>. Las ventajas que brinda Squeak son las siguientes:

- Conexión por puerto serie: de esta manera se puede comunicar con los robots debido a que Windows representa al Bluetooth como un puerto COM.
- Herramientas de programación en objetos: Squeak, como entorno de programación, responde al paradigma de objetos de manera eficiente y posee todas las herramientas para hacerlo altamente productivo tanto para su uso como para la enseñanza.
- Portabilidad: trabaja sobre una máquina virtual que ha sido portada a casi cualquier sistema operativo consumiendo recursos moderados.
- Open-source: es completamente abierto, con una comunidad activa, y permite hacer todo tipo de modificaciones inclusive en tiempo de ejecución.

Estas características hacen de Squeak un ambiente que potencia la creatividad y la imaginación permitiendo al usuario trabajar con completa libertad.

El campo de acción elegido para fútbol simulado es el software RobotSoccer v1.5a<sup>3</sup> (simulador oficial de la Fira (Federation of International Robot-Soccer Association) para las competencias “SimuroSot”) asociado a un proxy para hacer la comunicación con Squeak.

---

<sup>1</sup> Más información acerca de Squeak en <http://squeak.educarex.es/Squeakpolis> y <http://www.squeakland.org/>

<sup>2</sup> Más información acerca de Smalltalk en <http://www.squeak.org/Smalltalk/>

<sup>3</sup> Puede descargarse en <http://www.fira.net/soccer/simurosot/overview.html>

En cambio, para fútbol físico se eligió una cancha de pasto sintético de dimensiones acordadas en competencias a nivel internacional. Para conocer la posición de los robots y de la pelota se utilizó un sistema de visión con un software de procesamiento de imágenes open-source llamado Doraemon.

Los robots fueron armados con el kit educativo Lego Mindstorms Nxt<sup>4</sup> de la empresa Lego y se comunican utilizando el protocolo Bluetooth.

Los motivos por los cuales se eligió el kit de Lego son los siguientes:

- No se necesitan conocimientos avanzados de electrónica para armarlos. De esta manera se hace mayor hincapié en conceptos vinculados con la programación del comportamiento.
- Es rápida la creación de robots posibilitando la prueba de distintos diseños en poco tiempo.
- Se puede comunicar fácilmente vía Bluetooth al Squeak.
- Es un producto que está en constante evolución debido a que la empresa va actualizando los kits frecuentemente.
- Hay diversos tutoriales en Internet tanto para neófitos como para avanzados en el tema.

## 5. Descripción del proyecto

El proyecto consta básicamente, de 3 partes:

- Una estrategia modelo.
- Una capa de comunicaciones que varía según se trate de robótica física o simulada.
- Un visor del partido en tiempo real.

---

<sup>4</sup> Más información acerca de Lego Mindstorms Nxt en <http://mindstorms.lego.com/>

## 5.1 Desarrollo de la estrategia modelo

Como ayuda para el usuario, se desarrolló una estrategia básica con comportamientos de navegación fundamentales para que actúe como cimiento de estrategias que, posteriormente, sean perfeccionadas. Aunque funciona de una manera muy primitiva, marca las pautas básicas a seguir para el desarrollo de una estrategia efectiva.

Para el modelado de la estrategia se tuvo en cuenta la creación de 4 clases fundamentales:

- **Environment:** Representa el entorno del partido y cumple la función de comunicarse con el proxy. Posee las siguientes variables de instancia:
  - **home:** una colección con los jugadores de nuestro equipo.
  - **opponent:** una colección con los jugadores del equipo contrario.
  - **currentBall:** representa la pelota actual.
  - **predictedBall:** representa la pelota en su posición siguiente.
  - **lastBall:** representa la pelota en su última posición.
  - **fieldBounds:** representa los límites de la cancha.
  - **goalBounds:** representa los límites de los arcos.
  - **whosBall:** indica quién tiene la pelota.
  - **team:** Indica el color de nuestro equipo.
- **Robot:** Representa un robot de nuestro equipo y posee las siguientes variables de instancia:
  - **rotation:** Indica el ángulo de rotación del robot.
  - **position:** Representa un punto indicando su posición.
  - **velocityRight:** Indica la velocidad de la rueda derecha.

- **velocityLeft:** Indica la velocidad de la rueda izquierda.
- **env:** Representa al Environment en el que se juega el partido.
- **OpponentRobot:** Representa un robot del equipo contrario y posee las siguientes variables de instancia:
  - **rotation:** Indica el ángulo de rotación del robot.
  - **position:** Representa un punto indicando su posición.
- **Ball:** Representa la pelota y posee la siguiente variable de instancia:
  - **position:** Representa un punto indicando su posición.

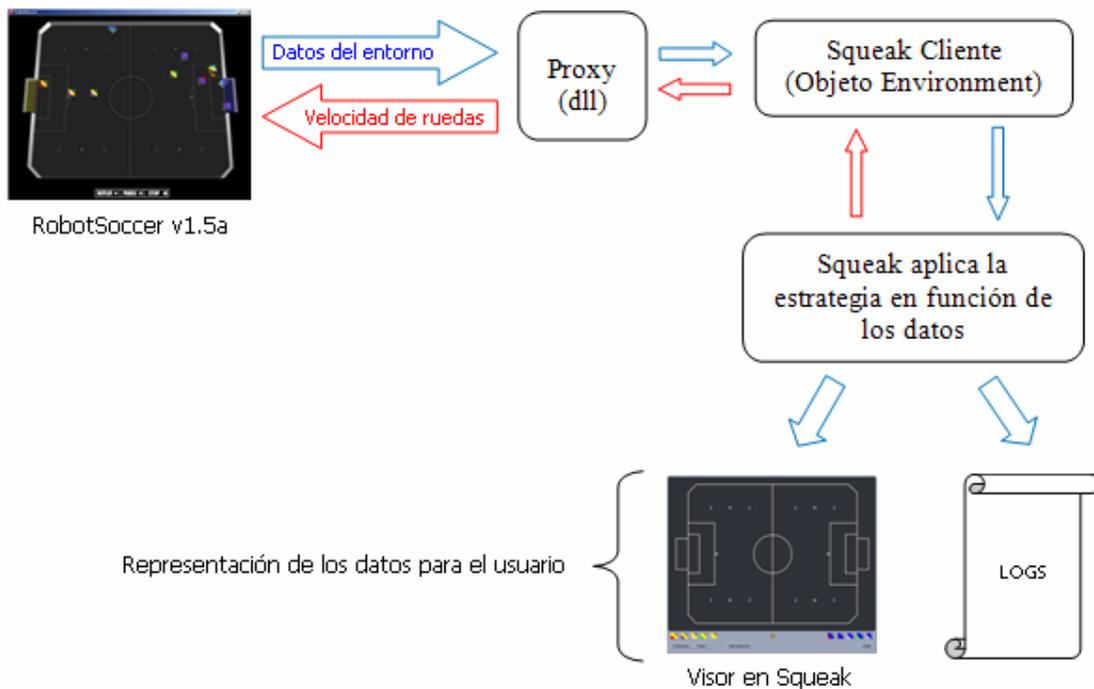
Los métodos de una estrategia se pueden dividir en 4 categorías esenciales:

- **De cálculo:** Obtienen información a partir de los datos del partido. Por ej.: Calcular el ángulo que hay que rotar para mirar un objeto determinado (la pelota, el arco, etcétera), predecir el movimiento de la pelota, etcétera.
- **Referentes a roles:** Especifican comportamientos particulares que dinámicamente se asignan a los jugadores. Por ej.: arquero, defensor, etcétera.
- **Referentes a estrategia:** Conforman patrones de juego y reglas a seguir por el equipo en su totalidad. Por ej.: ofensiva, defensiva, etcétera.
- **De navegación:** Especifican la forma en que un robot debe moverse de manera óptima. Por ej.: moverse a una zona de la cancha, moverse a un punto fijo, moverse a un punto móvil, etcétera.

La estrategia modelo desarrollada consta de métodos básicos para algunas de las categorías mencionadas. Sin embargo, se espera que el usuario se interroge acerca de las problemáticas que plantean a fin de perfeccionar la estrategia y hacerla tan compleja y eficiente como guste.

## 5.2 Aplicación en Robótica Simulada

La robótica simulada permite ver cómo se desempeñan los robots en un ambiente completamente virtual. Si bien las condiciones son ideales debido a que los robots nunca se dañarán o sufrirán algún cambio debido a alguna variable física real específica, es interesante para probar diferentes estrategias y técnicas en poco tiempo, y puede servir para trabajar cuando la situación económica es limitada.



- En líneas generales, el ejemplo de robótica simulada tratado en la investigación puede dividirse en dos elementos esenciales: Un simulador y un proxy.
- El simulador RobotSoccer1.5a es un software que representa virtualmente un partido de fútbol de robots. Permite cargar estrategias de equipos programadas en una cantidad limitada de lenguajes (C++ y Lingo).

- El proxy es una librería que se carga en el simulador (como si fuera un equipo) y funciona como un intermediario entre el simulador y cualquier plataforma terminando con la limitación a la hora de elegir lenguajes. Codificando los datos en un string y comunicándose por sockets, el proxy envía la posición y rotación de cada robot al objeto Environment y recibe de él (luego de haber aplicado la estrategia) la nueva velocidad de las ruedas izquierda y derecha para actualizar el estado de los robots del simulador. El proxy utilizado fue creado por la Universidad del Comahue con modificaciones posteriores de nuestro equipo.

## 5.3 Aplicación en Robótica Física

La robótica física se distingue claramente de la simulada ya que ésta se desempeña en circunstancias sometidas a distintas variables que condicionan la autonomía del robot como por ejemplo el rozamiento, los choques, el dinamismo de la

luz, la pérdida de batería, etcétera. Este tipo de variables hace que en algunos casos, lógicamente, los robots no se comporten de la manera deseada. Si bien el grado de complejidad es mayor en el mundo real, el poder implementar soluciones en un marco complejo no debe verse como un obstáculo sino como un desafío que tiene más relación con nuestro entorno y puede resultar sumamente útil en el futuro para mejorar la calidad de vida humana.

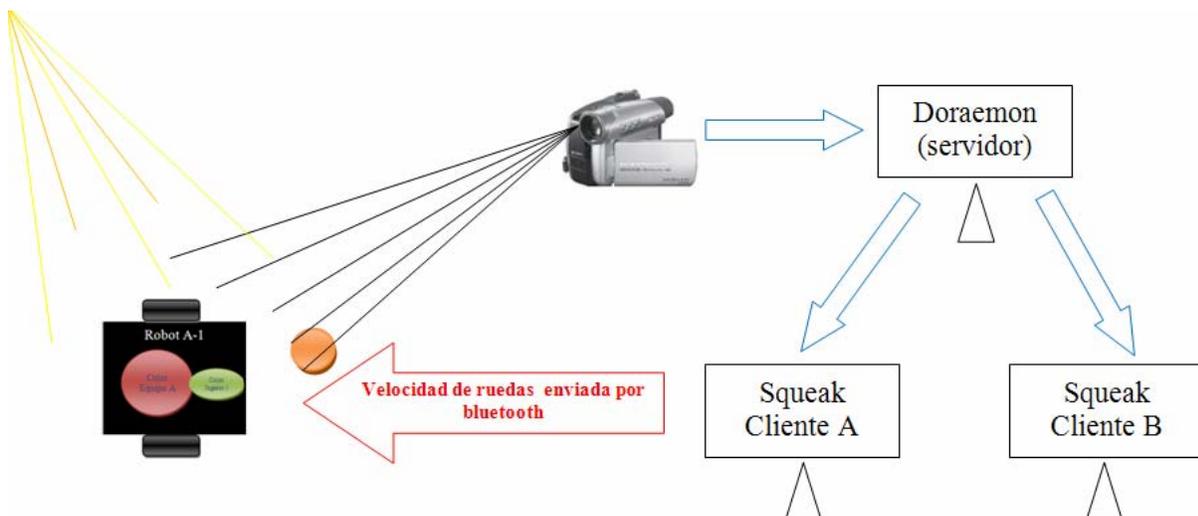
A diferencia del desarrollo realizado en robótica simulada, en robótica física se utiliza para captar el ambiente una cámara que detecta la pelota y unos parches situados en la parte superior de los robots; está asociada a un sistema de visión servidor y se comunica con las máquinas cliente por medio de un socket enviándoles los datos del ambiente. Los parches son de distintos colores para marcar la diferencia entre equipos y entre robots de un mismo bando.

El sistema de visión utiliza para procesar los datos del ambiente el software Doraemon y se ejecuta bajo Linux. Sin entrar en detalles, el Doraemon realiza una correspondencia entre centímetros reales de una zona rectangular a píxeles. Para lograr esto se necesita armar una alfombra calibradora con una cantidad determinada de cuadrados de igual tamaño y separados en intervalos iguales. Una vez hecha la correspondencia, se pasa a calibrar los colores de cada equipo y luego de cada robot.

La calibración de colores se basa en seleccionar zonas determinadas del parche para que guarde un espectro de colores relativamente amplio y así poder diferenciar los robots, la pelota y la cancha sin perder tolerancia a reflejos fugaces. Es indispensable tomar en cuenta factores como la posición y el ángulo de la cámara y sobre todo elegir una iluminación adecuada y que no varíe para no tener problemas ulteriores en el desarrollo del partido. Razón por la cual es conveniente antes de calibrar, elegir primero los colores de los parches que se le colocarán a los robots y el color de la pelota.

Cuando las máquinas cliente reciben los datos del entorno (que en Squeak serán los valores de las variables de instancia del objeto environment), se los procesa en base a la estrategia teniendo como salida la velocidad de las ruedas de cada robot. Estos datos son enviados en tiempo real desde Squeak a los distintos robots Lego vía bluetooth. Cabe destacar que también fue probado en robots armados íntegramente en el Centro de Altos Estudios en Tecnología Informática (CAETI) con la variante que se comunicaban por radio. Esto fue posible porque se tiene un único framework en donde sólo se modifica la capa de comunicaciones posibilitando un grupo más heterogéneo de tipos de robots y protocolos de comunicación.

## 6. Visor en Squeak



Para visualizar el partido del lado del cliente y tener acceso a información rica del entorno y de los robots en tiempo real se optó por hacer un visor del partido en Squeak a base de morphs<sup>5</sup> que representan la cancha, los robots y la pelota. Este visor permite inspeccionar, en cualquier momento del partido, la posición y rotación exacta de cada robot (así como todas las variables particulares que se hayan definido). Dichos valores no se muestran en el simulador RobotSoccer ni tampoco se guardan para poder analizarlos con mayor detenimiento, por lo que tenerlos disponibles de esta forma proporciona una ventaja valiosa a la hora de analizar el comportamiento del equipo. Otra ventaja es que facilita la conexión de diferentes equipos al presentar una interfaz gráfica olvidando por completo el tener que ejecutar líneas de código como se hacía inicialmente.

Los Morphs utilizados son:

- BallMorph: Representa la pelota.
- RobotMorph: Representa un robot de un equipo.
- RobotSoccerMorph: Representa el campo de juego de los robots.
- RobotSoccerViewerMorph: Representa el visor entero (tanto la cancha como los controles para manejar el partido).

## 7. Consideraciones para el uso educativo

Si bien se dejó en claro que el fútbol de robots en Squeak se puede utilizar tanto para robótica física como simulada, es pertinente destacar algunas diferencias entre ambas:

### Robótica Simulada:

---

<sup>5</sup> Básicamente, los morphs son objetos de Squeak que tienen representación gráfica. Para una explicación más detallada del sistema Morphic, véase <http://wiki.squeak.org/squeak/30>

Ventajas:

- ✓ Bajo costo.
- ✓ Mundo ideal en el que se puede optar por omitir variables físicas y focalizar la investigación en otros tópicos.
- ✓ Es fácil de implementar técnicas de navegación y de captura de ambiente.
- ✓ Resultados rápidos.
- ✓ Las pérdidas en caso de que haya un inconveniente en el sistema son menores.
- ✓ Es portable.

Desventajas:

- ❖ Las situaciones son ideales y distan del mundo real perdiendo cierto grado de aplicabilidad concreta.

### Robótica Física:

Ventajas:

- ✓ Acorde al mundo real por lo tanto tiene una aplicabilidad inmediata.

Desventajas:

- ❖ Costo moderado/alto.
- ❖ Requiere algún tipo de capacitación previa en electrónica (exceptuando el caso en el que se elijan kits).
- ❖ Poca reutilización.
- ❖ Se tienen en cuenta determinadas variables físicas que hacen más compleja la implementación de la estrategia.
- ❖ Se deben tratar cuidadosamente los robots para evitar cualquier tipo de daño que sea propenso a inutilizar el sistema.

## 8. Conclusiones

Este proyecto incentiva la investigación en Fútbol de Robots tanto a nivel de educación media como universitaria. Este tipo de investigación convierte al estudiante en el principal protagonista, requiriendo actividades que lo implican cognoscitivamente, promoviendo procesos de indagación, investigación y búsqueda. Se nutre de diversas ramas de la ciencia (como Lógica, Matemática, Física, Inteligencia Artificial, Robótica y Programación) así como también potencia la camaradería y trabajo en grupo. En el caso del fútbol de robots físico también funciona como un vínculo entre el universo virtual de la computadora (que nos permite hacer cualquier tipo de simulaciones) y el mundo real con sus limitaciones físicas. Estas características convierten el aprendizaje en una actividad original y entretenida que genera curiosidad en los estudiantes, lo cual ayuda a solucionar uno de los principales problemas actual en la educación: la falta de motivación.

Por otro lado, el uso de una plataforma que permite y fomenta la experimentación en todos los niveles facilita al estudiante la construcción de su conocimiento situando al docente en el papel de mentor. De acuerdo con la teoría constructorista de Seymour Papert, la computadora en el proceso educativo debe ser concebida como un medio para la construcción del conocimiento, promoviendo la exploración libre de los estudiantes dentro de un marco dado. Basándonos en estos conceptos es que extendemos el enfoque utilitario actual de la computadora: no es sólo una herramienta que puede ayudarnos a ser más productivos y conectarnos con el mundo, sino también un instrumento que permite expresar nuestras ideas en modos diferentes a lo conocido. La combinación entre los desafíos cognitivos del fútbol de robots junto con la motivación que genera en los alumnos dicha actividad, hace de este recurso una plataforma maravillosa para la educación.

1. Alan Kay, "The Real Computer Revolution Hasn't Happened Yet", Viewpoints Research Institute, 2007.
2. Seymour Papert, "Mindstorms: Childrens, Computers, and Powerful Ideas", Brighton Harvester Press, 1980.
3. Squeak: Open Personal Computing and Multimedia, Prentice Hall, 2002.
4. B. J. Allen Conn y Kim Rose, "Powerful ideas in the Classroom", Viewpoints Research Institute, 2003.
5. Diego Gómez Deck y José L. Redrejo Rodríguez, "Squeak en España como parte del Proyecto LinEx", 2003.
6. Stephan Ducasse, "Learning Programming in Squeak", Morgan Kaufman Publisher, 2003.
7. Fernando Fraga y Adriana Gewerc, "Una experiencia interdisciplinar en Ed. Primaria mediante el uso de Squeak", Universidad de Santiago de Compostela, 2004.
8. Ana Pizarro Galán y otros, "Un mundo para aprender", Editorial Editlin, 2005.
9. Gonzalo Zabala, "Squeak, el laboratorio infinito", Revista Novedades Educativas Edición 185, Editorial Novedades Educativas, Mayo 2006.
10. Gonzalo Zabala, "Desarrollo en un entorno educativo de objetos para el control de una interfaz de domótica", Anales de WICC 2007 [668-673], Mayo 2007.
11. The Lego Group, "LEGO MINDSTORMS NXT Bluetooth Developer Kit", 2006.
12. The Lego Group, "LEGO MINDSTORMS NXT Hardware Developer Kit", 2006.

## 9. Referencias: