

Memoria del Proyecto de Innovación Docente

APRENDIZAJE ACTIVO DE PROGRAMACIÓN CON LEGO MINDSTORMS NXT



Profesor responsable

Juan Carlos Matos Franco

Departamento de Informática y Automática
Escuela Politécnica Superior de Zamora

Entidad financiadora

Universidad de Salamanca

Periodo de desarrollo

Julio/2010 – Junio/2011

Dirigido a

Vicerrectorado de Docencia y Convergencia Europea

ÍNDICE

1. INTRODUCCIÓN.....	3
2. JUSTIFICACIÓN DEL PROYECTO Y UTILIDAD.....	3
3. OBJETIVOS Y METODOLOGÍA.....	4
4. PLAN DE TRABAJO SEGUIDO.....	6
5. VALORACIÓN DE RESULTADOS.....	8
6. CONCLUSIONES Y LÍNEAS DE TRABAJO FUTURAS.....	9
7. REFERENCIAS	10

1. INTRODUCCIÓN

Los *Legó Mindstorms NXT* son unos robots de tipo educativo que están basados en el popular juego de construcción *Legó*, integrados por diferentes partes mecánicas y electromecánicas controladas por un bloque programable. Este bloque es un computador integrado (NXT) que puede ser programado de forma flexible, ya que admite diferentes lenguajes de programación (además del que trae por defecto), de forma que el robot puede recibir información del entorno a través de distintos sensores y realizar una serie de operaciones con sus actuadores (Figura 1).



Figura 1. Bloque NXT, sensores y actuadores disponibles en el paquete educativo.

2. JUSTIFICACIÓN DEL PROYECTO Y UTILIDAD

El propósito principal de la aplicación de este proyecto ha sido que el estudiante emplease diferentes metodologías de programación sobre un dispositivo *físico* (en concreto programarlo con los lenguajes C y Java), y que observase lo que ocurría cuando éste interactuaba con el mundo real. Con los *Legó Mindstorms NXT* el alumno puede solucionar problemas de diversa índole, algunos de los cuales ya se han resuelto de forma teórica en el aula, observando además que puede tomar datos del entorno, tratarlos y actuar en consecuencia. Con ello se ha pretendido aumentar la comprensión de las asignaturas de programación y la motivación de los alumnos en el aprendizaje de la misma, mediante una aplicación práctica.

Otro propósito del proyecto fue potenciar el aprendizaje colaborativo mediante la creación de grupos de trabajo organizados de forma que cada alumno se encargase de una parte del trabajo y tomase decisiones por su cuenta (selección de los elementos del robot, diseño de funcionalidades, búsqueda de soluciones, etc.), integrándose posteriormente todo en el proyecto global.

Por último se planteó que cada alumno realizase un informe, de forma individual, exponiendo sus aportaciones al trabajo realizado e indicando qué dificultades se había encontrado en su desarrollo y qué conclusiones había alcanzado tras la realización de cada una de las prácticas.

3. OBJETIVOS Y METODOLOGÍA

Los objetivos del proyecto fueron ligeramente diferentes según la asignatura en la que se aplicaron:

- En la asignatura “Fundamentos de programación” se aprende a programar utilizando una programación estructurada, para la cual se emplea el lenguaje C. Los *Leg Mindstorms* pueden programarse con extensiones del lenguaje C (*RobotC*, *Not Quite C*, etc.) que cuentan con entornos integrados de desarrollo multiplataforma (con versiones para Windows, Linux y OSX) y con potentes herramientas adicionales. Las prácticas que se realizaron para esta asignatura se encaminaron a la realización de funciones y procedimientos específicos para controlar los robots, empleando para ello elementos de programación del lenguaje C (datos, estructuras de control, etc.). Aunque inicialmente se intentó emplear *Not Quite C*, dado su carácter libre, al final se utilizó la extensión comercial *RobotC for Mindstorms 2.26.1* (Figura 2), que incluye también un entorno de desarrollo integrado (IDE), dado que *Not Quite C* no funciona actualmente con los modelos NXT (sólo lo hace con los antiguos robots RCX).
- En las asignaturas de “Programación Orientada a Objetos”, tanto de Ingeniería Técnica en Informática de Gestión como del Grado en Ingeniería Informática en Sistemas de Información, se emplea un enfoque orientado a objetos. La forma de programar cambia radicalmente y para mostrar esta diferencia se propuso inicialmente la resolución de los mismos problemas que en la asignatura

“Fundamentos de programación”, sólo que desde un punto de vista diferente, empleando clases, métodos y propiedades (a través de un interfaz de programación de aplicaciones, API), en lugar de funciones y procedimientos. Posteriormente se desarrollaron también los conceptos propios de la programación orientada a objetos, como herencia y polimorfismo. El lenguaje utilizado en esta asignatura fue Java, mediante el *framework Lejos NXT*, en su versión 0.8.5 y, en las últimas prácticas, con la nueva versión 0.9.0. Dado que los alumnos programan básicamente utilizando el entorno de desarrollo integrado educativo *BlueJ*, se incorporó una extensión a este IDE para que les resultara familiar el sistema de trabajo y pudiesen centrarse únicamente en las características especiales de programación del robot (Figura 3).

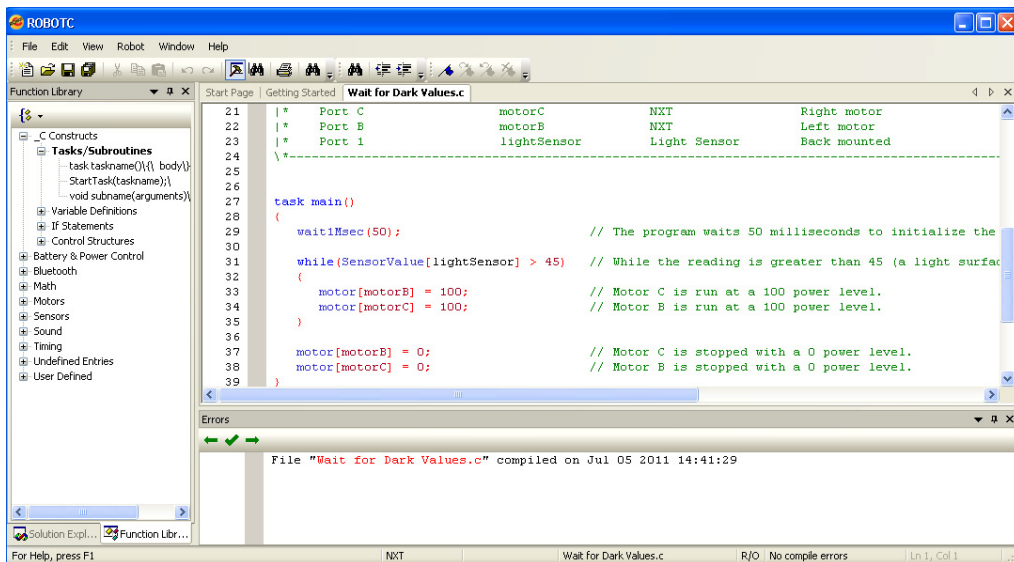


Figura 2. Ventana del IDE de *RobotC*.

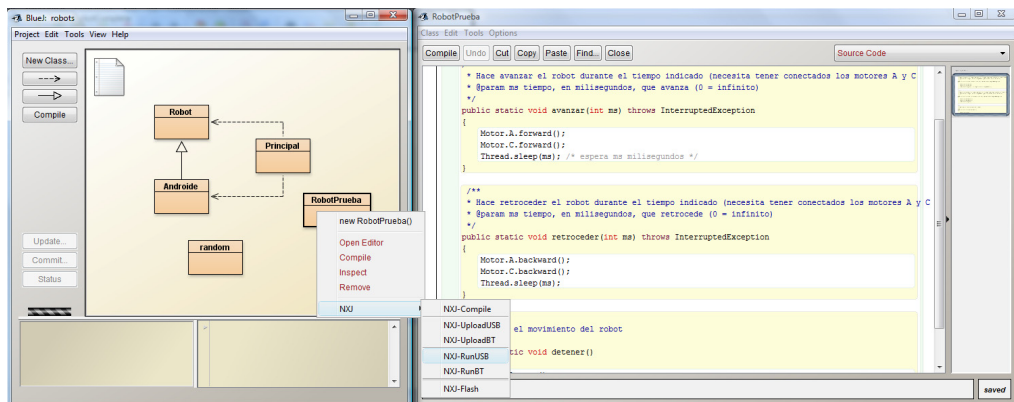


Figura 3. Ventana del IDE de *BlueJ* con la extensión para *Legos NXT*.

Las asignaturas concretas en las que se aplicó la metodología de trabajo desarrollada se detallan a continuación:

- Fundamentos de Programación
(1º curso del Grado en Ingeniería Informática en Sistemas de Información).
- Métodos de Programación
(2º curso de Ingeniería de Materiales).
- Programación Orientada a Objetos
(3º curso de Ingeniería Técnica en Informática de Gestión).
- Programación Orientada a Objetos
(1º curso del Grado en Ingeniería Informática en Sistemas de Información).

4. PLAN DE TRABAJO SEGUIDO

En primer lugar se efectuaron varias reuniones de coordinación entre los profesores del grupo para definir la estructura de las prácticas y su ubicación en los programas de las distintas asignaturas. En ellas también se hizo una planificación temporal de la realización y entrega de cada uno de los ejercicios prácticos.

Previo a la realización de las prácticas en cada uno de los semestres se impartió un seminario que explicaba el funcionamiento básico de los robots NXT: estructura, construcción, comunicación con el exterior a través de sensores y actuadores y por último su programación. Para que los estudiantes puedan crear un determinado robot que resuelva un problema propuesto, es necesario que sepan determinar qué componentes deben utilizar antes de programarlo.

En cada una de las asignaturas se establecieron tres ejercicios prácticos. Para estas prácticas se definieron los grupos de trabajo, se repartieron los roles y se les facilitó un espacio de trabajo privado en el curso virtual de la asignatura en la plataforma *Studium* (compuesto por un gestor de documentos, un foro, un chat y un wiki), para que así los alumnos pudieran trabajar en línea de forma colaborativa. Las prácticas se fueron realizando de forma paralela al desarrollo de la asignatura, con

distintas sesiones de control tutorizadas, y los resultados obtenidos se presentaron ante el profesor de la asignatura. Además cada alumno, de forma individual, realizó un informe de su trabajo.

Los estudiantes cumplieron también un pequeño cuestionario sobre su participación, exponiendo las dificultades que se habían encontrado en su trabajo e indicando las observaciones que estimaron oportunas (dificultades, propuestas de mejora, modificaciones, criterios de calificación, etc.).

Al término de los dos semestres se realizó la valoración global de la actividad y se extrajeron las conclusiones finales del proyecto, definiendo las actuaciones necesarias para ampliar y mejorar la docencia en los próximos cursos.

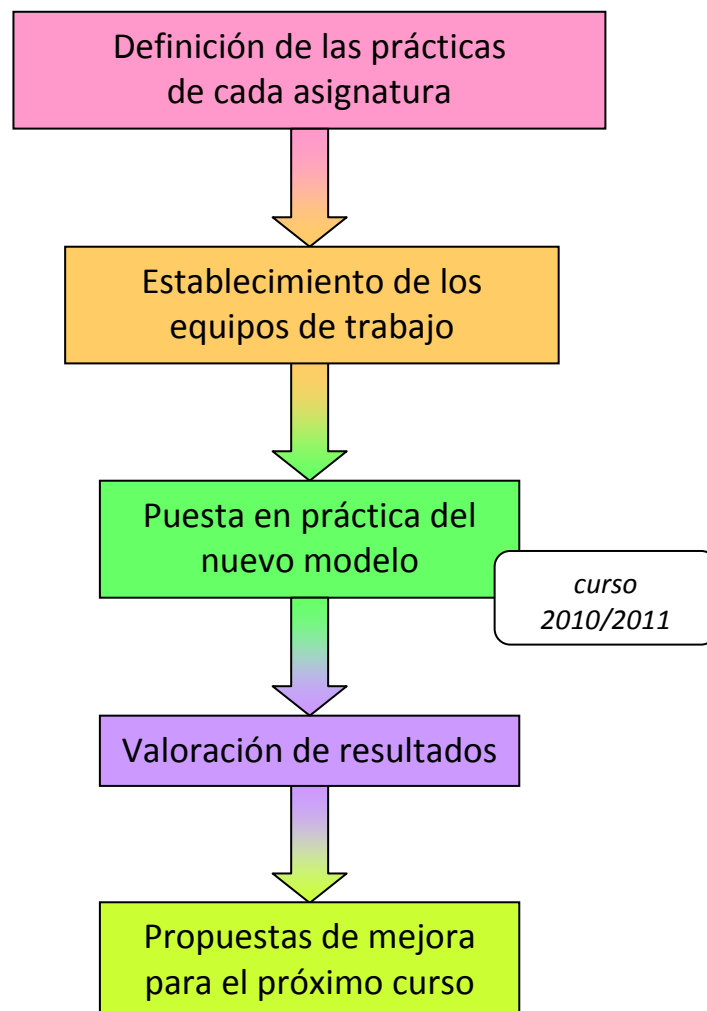


Figura 4. Etapas del desarrollo del proyecto.

En la Figura 4 se muestran de forma resumida las etapas generales cubiertas en el proyecto. Como punto de partida se establecieron las prácticas de cada una de las asignaturas propuestas y se establecieron los grupos de trabajo. Con el análisis de cada una de ellas se determinaron los aspectos que se podían tratar de manera virtual y los que era necesario realizar en el aula (como el trabajo con los robots), así como la puntuación que se iba a dar a cada parte del trabajo. Una vez realizadas las prácticas se valoraron las ventajas e inconvenientes de las mismas, identificando aspectos que podían mejorarse de cara al próximo curso.

5. VALORACIÓN DE RESULTADOS

Los alumnos presentaron un informe de cada práctica realizada, detallando la solución empleada y las decisiones tomadas, así como los problemas que se encontraron durante su desarrollo. También se realizaron encuestas a los alumnos para conocer la valoración que hacían del nuevo modelo de trabajo propuesto. Como resultado del análisis de los informes y de las respuestas a los problemas encontrados se extrajeron las siguientes conclusiones:

- Los alumnos valoran positivamente la experiencia de trabajo con el robot. Más del 80% de los alumnos se mostraron muy entusiasmados con la realización de estas prácticas, el 70% de los alumnos consideró útil o muy útil este sistema y al 98% le pareció muy positiva la realización de las prácticas de forma grupal y coordinada entre diferentes grupos.
- La plataforma virtual facilita y agiliza la gestión de los trabajos por parte de los alumnos. Resulta sencillo fijar plazos de entrega y comunicar a los alumnos el resultado de sus prácticas. Aunque siempre hay algún alumno que realiza la entrega tarde, en general se ha observado que el conjunto se ajusta a las fechas impuestas. El hecho de que la plataforma impida entregar trabajos fuera de plazo (o lo permita, pero se lo advierta tanto al alumno como al profesor) contribuye a que los alumnos se organicen y dosifiquen su esfuerzo.
- Las principales dificultades con las que se encontraron los alumnos fueron el aprendizaje de la utilización de las librerías y de los APIs de programación, ya que son bastantes extensos debido a la cantidad de componentes y funcionalidades que tienen

los robots. Otra dificultad adicional fue la comprensión del funcionamiento físico del robot (mecánica, movimientos, rotaciones, sensores, etc.), ya que algunos de los alumnos de las titulaciones de Informática nunca habían trabajado con ellos, por lo que se les proporcionó un montaje básico de un robot con el que empezar a trabajar (Figura 5), así como un sistema de anclajes básicos para los distintos tipos de sensores.



Figura 5. Montaje básico del robot.

6. CONCLUSIONES Y LÍNEAS DE TRABAJO FUTURAS

En el desarrollo de este proyecto se han utilizado robots educativos *Legó Mindstorms NXT* para el aprendizaje de diferentes lenguajes (y metodologías) de Programación. Se han realizado una serie de aplicaciones donde el estudiante ha puesto en práctica los conceptos teóricos aprendidos en las distintas asignaturas, de forma que ha visto el resultado de lo que programa sobre un dispositivo físico y cómo interactúa con el mundo real. De esta manera se potencia y facilita el aprendizaje, pues se crea un entorno en el que los estudiantes pueden desempeñar actividades propias de una Ingeniería.

Los resultados obtenidos con este proyecto corroboran la utilidad de la forma de trabajo introducida. Los alumnos han aceptado el modelo de prácticas propuesto, lo han utilizado de forma intensiva y lo han valorado positivamente. Por todo ello el próximo

curso se seguirá aplicando el nuevo modelo en las asignaturas incluidas en este proyecto y se hará extensible a algunas más, coordinadas por los profesores que han participado en este proyecto. En concreto se trabajará con *Microsoft Robotics Studio* para programar los robots con los lenguajes *C#* y *Visual Basic .NET* en las asignaturas de programación del Grado en Ingeniería Mecánica, que además incorpora la posibilidad de realizar simulaciones 3D del proceso de trabajo.

7. REFERENCIAS

AI and Robots Using Lego Mindstorms NXT.

<http://www.computing.northampton.ac.uk/~scott/robot_nxt_ai/default.html>

Última modificación: 9/3/2011. Consultado el 30/6/2011.

Barnes, D.J. (2009). *Programming LEGO robots with BlueJ*. SIGCSE 2009. Disponible online: <<http://www.bluej.org/bluej-greenfoot-day/content/bluej-nxt.pdf>>. Consultado el 30/6/2011.

Griffin, T. (2010). *The Art of Lego Mindstorms NXT-G Programming*. No Starch Press.

Kelly, J.F. (2010). *Lego Mindstorms NXT-G Programming Guide, 2ed.* Apress.

LeJOS, Java for Lego Mindstorms.

<<http://lejos.sourceforge.net/>>

Última modificación: 16/5/2011. Consultado el 30/6/2011.

LejOS NXJ extension for BlueJ.

<<http://homepages.feis.herts.ac.uk/~comqdhb/lego/bluej.php>>

Última modificación: N/D. Consultado el 30/6/2011.

Programar el Lego NXT desde cero con RobotC.

<<http://www.agustin.mx/Blog.php?TutorialRobotC>>

Última modificación: 22/6/2011. Consultado el 29/6/2011.

ROBOTC, a C Programming Language for Robotics.

<<http://www.robotc.net/>>

Última modificación: 17/6/2011. Consultado el 30/6/2011.

Schueller, A.W. (2010). *Programming with Robots*. Whitman College, USA. Disponible online: <<http://carrot.whitman.edu/Robots/notes.pdf>>. Consultado el 25/6/2011.