

# Proyecto ingeniería

Tema 1. Introducción a  
proyecto de ingeniería

# Tema 1. Introducción a proyecto de ingeniería

## Índice

Objetivos de aprendizaje .....	3
1. Presentación .....	3
2. Aprendizaje basado en proyectos (PBL) .....	3
2.1. ¿Por qué PBL? .....	4
2.2. El rol del estudiante en PBL.....	5
2.3. El rol del profesor en PBL .....	6
3. Internet de las cosas (IoT) .....	6
3.1. Definición de Internet de las cosas .....	7
4. Programación orientada a objetos: Java .....	9
4.1. Propiedades de la programación orientada a objetos.....	10
4.2. Propiedades del lenguaje Java .....	10
5. Microcontroladores programables: Arduino .....	11
5.1. ¿Qué es Arduino? .....	12
6. Herramientas para el desarrollo de proyectos .....	13
6.1. Desarrollo <i>software</i> .....	13
6.2. Gestión de proyectos.....	14
7. Resumen .....	15
Referencias bibliográficas.....	16

# Tema 1. Introducción a proyecto de ingeniería



## Objetivos de aprendizaje

Los objetivos que se pretenden alcanzar en este recurso son los siguientes:

- Entender en qué consiste el aprendizaje basado en proyectos.
- Comprender el concepto de Internet de las cosas.
- Refrescar el paradigma de programación orientado a objetos.
- Comprender la utilidad de microcontroladores programables.
- Comprender la utilidad de las herramientas para desarrollo de proyectos.
- Comprender la utilidad de las herramientas para gestión de proyectos.

### 1. Presentación

En la asignatura Proyecto de Ingeniería se propone a los alumnos desarrollar un proyecto completo de ingeniería en equipos, aplicando los conocimientos adquiridos en las asignaturas previas para desarrollar un sistema *hardware-software* acorde a los tiempos que corren.

Se presentan los últimos avances en algunas de las áreas tecnológicas de mayor impacto dentro del sector TIC, como, sin duda, son los microcontroladores programables, cableado de circuitos, lenguajes de programación orientados a objetos, IoT (*Internet Of Things* o Internet de las cosas), entre otros.

Todo este recorrido ha pretendido realizarse desde el aprendizaje basado en la realización de un proyecto integrador, sello de identidad de la Universidad Europea. Integrador de conocimientos, con trabajo en equipo y posibles colaboraciones de empresas externas. Todo esto es clave en el desarrollo profesional del ingeniero que partiendo de una base común mínima ha de ser capaz de proponer soluciones concretas a proyectos específicos. Por ello, la actividad central de la asignatura consiste en conceptualizar y desarrollar un sistema compuesto de varios módulos HW y SW para diferentes propósitos.

La asignatura enlaza con conceptos ya desarrollados en otras asignaturas como son Fundamentos de Programación, Programación Orientada a Objetos, Fundamentos Físicos para Informáticos, entre otras.

### 2. Aprendizaje basado en proyectos (PBL)

En la asignatura Proyecto de Ingeniería se va a trabajar de forma intensiva según la metodología de Aprendizaje Basado en Proyectos (*Project Based Learning-PBL*). Dicha metodología pretende conseguir el aprendizaje mediante el diseño, ejecución y evaluación de un proyecto por parte de los alumnos con la participación del docente

## Tema 1. Introducción a proyecto de ingeniería

como guía. El estudiante aprende en equipo y lidera su aprendizaje. El equipo docente guía y evalúa el proceso y el producto final asociado al proyecto.

### 2.1. ¿Por qué PBL?

Se ha comprobado que el aprendizaje resulta más atractivo, divertido y gratificante. Con todo ello se mejora el aprendizaje al lograr un conocimiento más profundo que permite al alumno recordar más y por más tiempo. Todo esto puede apreciarse en la pirámide de aprendizaje, donde se refleja que cuanto más se involucran los estudiantes, más asimilan y el conocimiento adquirido es más profundo.



Figura 1. Pirámide de aprendizaje.

Los proyectos propuestos por el equipo docente se encuentran orientados a un producto final, de manera que conecta a los estudiantes con el mundo real al tratarse de una forma de trabajar similar a la realizada en la empresa, donde se encuentran con problemas reales con múltiples posibles soluciones.

La metodología promueve la equidad educativa dado que cada alumno puede buscar su hueco y elegir el papel que desempeña en la comunidad que mejor se adapta a su manera de aprender. El conocimiento es compartido entre los miembros del equipo y distribuido otros equipos de la clase.

Además, se promueven diversas habilidades transversales, como son iniciativa, responsabilidad, autoconfianza, autonomía, resolución de problemas, trabajo en equipo, transmisión de ideas o autogestión.

## Tema 1. Introducción a proyecto de ingeniería



Figura 2. Los roles clásicos para el profesor y el alumno cambian.

Para que todo esto sea posible, Los roles clásicos del profesor y el alumno se modifican, como queda recogido en la siguiente cita de *Aalborg University*:



Teacher role: The teachers support the students during their work with the PBL activity as facilitators and/or tutors for the groups. The teachers' most important task is to facilitate the group processes and to help students work systematically according to the structure but the teachers need not be experts on the academic topics that students work with.

### 2.2. El rol del estudiante en PBL

¿Qué se espera de los estudiantes?

- Tiene un papel activo en su aprendizaje.
- Comprende el problema.
- Conoce los objetivos del proyecto.
- Se compromete con los compañeros de equipo para el logro de un aprendizaje efectivo.
- Busca la información necesaria, aprovechando los recursos disponibles, de manera autónoma.
- Comparte información y conocimientos.

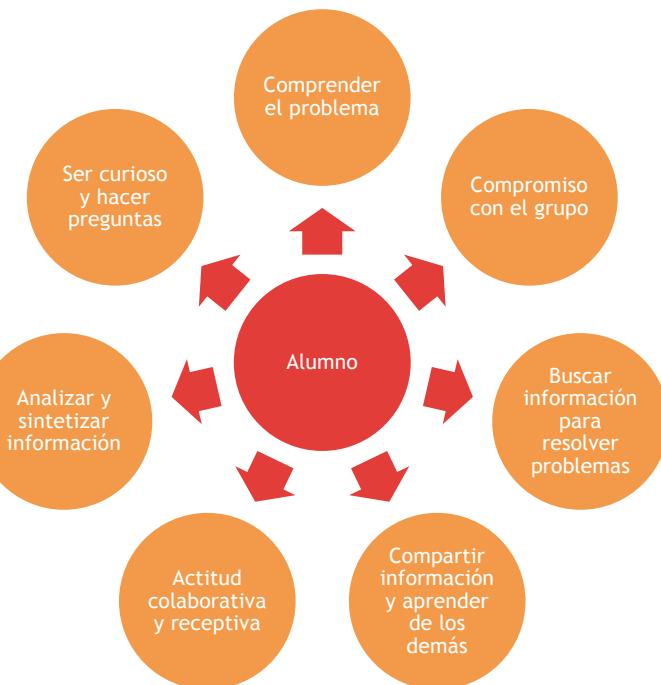


Figura 3. Alumno.

## Tema 1. Introducción a proyecto de ingeniería

- Aprende de los demás.
- Trabaja colaborativamente, con actitud receptiva.
- Aplica las habilidades de análisis y síntesis de la información con visión crítica.
- Es curioso y hace preguntas cuando lo necesita.

### 2.3. El rol del profesor en PBL

**¿Qué pueden esperar los estudiantes del profesor?**

- Diseña el proyecto y conoce la realidad social para proponer problemas reales.
- Da un papel protagonista al alumno en la construcción de su aprendizaje.
- Actúa como guía del aprendizaje.
  - No solo solucionar problema, hay que aprender.
  - No es experto técnico, sino *project manager*.

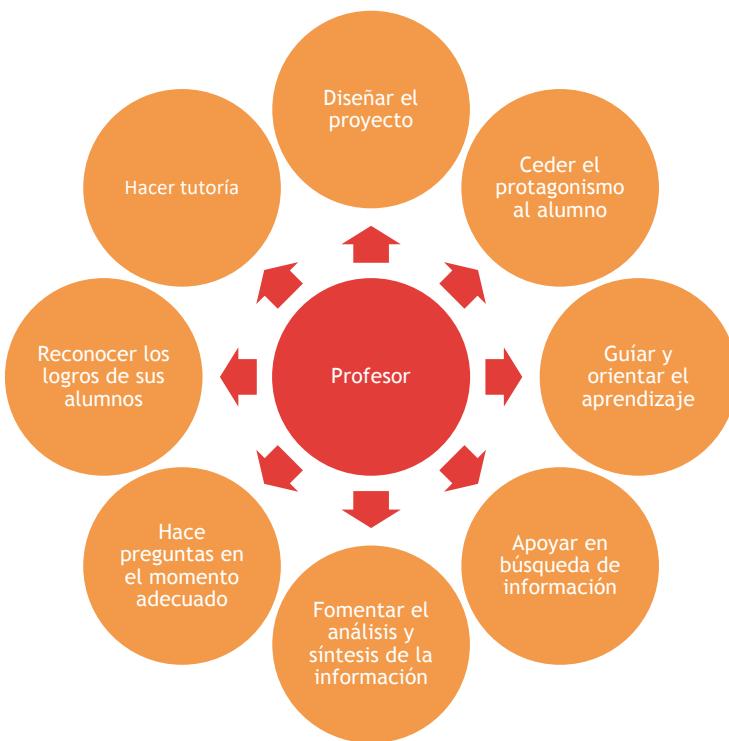


Figura 4. Profesor.

- Hace seguimiento del proyecto y realiza tutorías.
- Da apoyo en la búsqueda de información y fomenta el análisis y síntesis de la información haciendo las preguntas importantes en el momento adecuado.
- Reconoce los logros de sus alumnos.
- Persigue el plagio y los “mochileros”.

## 3. Internet de las cosas (IoT)

El concepto de Internet de las cosas (*Internet of Things-IoT*) fue propuesto por Kevin Ashton en el Auto-ID Center del MIT en 1999, donde se realizaban investigaciones en el campo de la identificación por radiofrecuencia en red (RFID) y tecnologías de sensores.

# Tema 1. Introducción a proyecto de ingeniería

## 3.1. Definición de Internet de las cosas

Existen múltiples definiciones para el término Internet de las cosas, pero todas tienen el mismo fondo. Se han recopilado varias definiciones que se exponen a continuación:



- Se refiere a la interconexión digital de objetos cotidianos con Internet. Alternativamente, Internet de las cosas es la conexión de Internet con más “cosas u objetos” que personas (...). Si objetos de la vida cotidiana tuvieran incorporadas etiquetas de radio, podrían ser identificados y gestionados por otros equipos, de la misma manera que si lo fuesen por seres humanos.
- La consolidación a través de Internet de una red que alojase una gran multitud de dispositivos, es decir, poder tener conectada a esta todas las cosas de este mundo como podrían ser vehículos, electrodomésticos, dispositivos mecánicos o simplemente objetos tales como calzado, muebles, maletas, dispositivos de medición, biosensores o cualquier objeto que nos podamos imaginar.
- Tecnología que establece una comunicación inteligente entre las cosas, facilitando la recogida de datos en línea, el control remoto y la automatización de procesos; todo ello permite reducir costes y proporcionar un mejor servicio, a la vez que mejorar la calidad de vida.

La idea subyacente es que cualquier objeto del mundo cotidiano es susceptible de convertirse, mediante la incorporación de elementos tecnológicos, en un dispositivo conectado. Dicha conexión a través de la red (Internet) le permite comunicarse con otros dispositivos, así como con los humanos. Tanto para su operación como para recoger los datos e información que es capaz de recolectar acerca de su entorno.

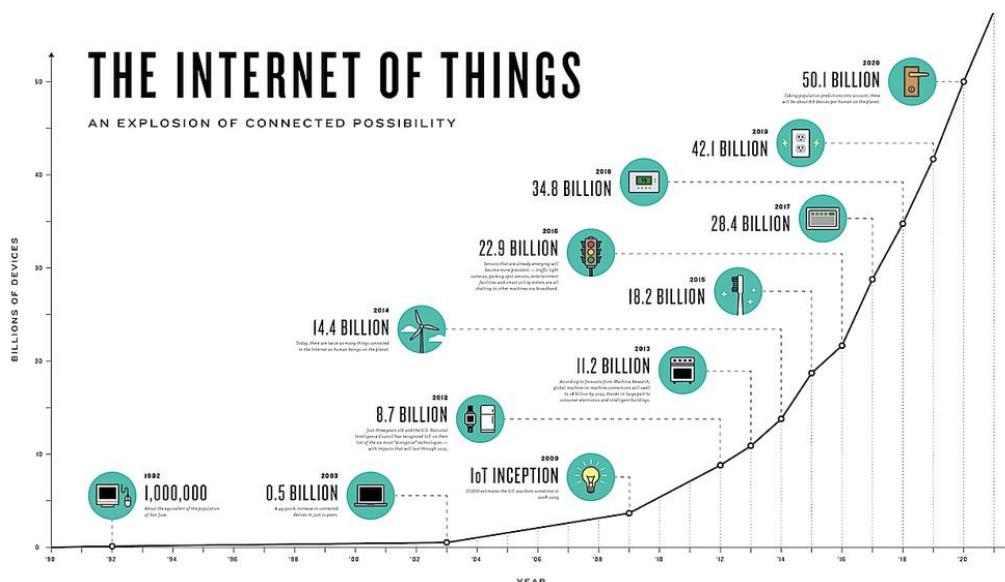


Figura 5. Evolución del número de dispositivos IoT conectados.

## Tema 1. Introducción a proyecto de ingeniería

Las posibles implicaciones y el impacto que esta tecnología iba a suponer ya las adelantaba Kevin Ashton en su artículo “Esa cosa del Internet de las cosas” para el diario RFID en 2009:



Los ordenadores actuales —y, por tanto, internet— son prácticamente dependientes de los seres humanos para recabar información. Una mayoría de los casi 50 *petabytes* de datos disponibles en internet fueron inicialmente creados por humanos, a base de teclear, presionar un botón, tomar una imagen digital o escanear un código de barras. Los diagramas convencionales de internet dejan fuera a los *routers* más importantes de todos: las personas. El problema es que las personas tienen un tiempo, una atención y una precisión limitados, y no se les da muy bien conseguir información sobre cosas en el mundo real. Y eso es un gran obstáculo. Somos cuerpos físicos, al igual que el medio que nos rodea. No podemos comer bits, ni quemarlos para resguardarnos del frío, ni meterlos en tanques de gas. Las ideas y la información son importantes, pero las cosas cotidianas tienen mucho más valor. Aunque, la tecnología de la información actual es tan dependiente de los datos escritos por personas que nuestros ordenadores saben más sobre ideas que sobre cosas. Si tuviéramos ordenadores que supieran todo lo que tuvieran que saber sobre las “cosas”, mediante el uso de datos que ellos mismos pudieran recoger sin nuestra ayuda, nosotros podríamos monitorizar, contar y localizar todo a nuestro alrededor, de esta manera se reducirían increíblemente gastos, pérdidas y costes. Sabríamos cuando reemplazar, reparar o recuperar lo que fuera, así como conocer si su funcionamiento estuviera siendo correcto. El IoT tiene el potencial para cambiar el mundo tal y como hizo la revolución digital hace unas décadas. Tal vez incluso hasta más.

De lo que exponía Ashton, resulta muy significativa la idea de que los dispositivos conectados son y serán los que poblarán Internet de información y datos de forma autónoma, cuando esta labor la realizaban los humanos con su esfuerzo y dedicación de forma manual antes de la explosión del IoT.

### 3.1.1. Casos de uso de Internet de las cosas

Los usos y aplicaciones de los dispositivos IoT son innumerables. Se utilizan en una gran cantidad de sectores, tanto empresariales como domésticos y la tendencia es que vayan a más, prácticamente duplicando el número de dispositivos conectados cada 2 años, como puede observarse en la figura anterior.

## Tema 1. Introducción a proyecto de ingeniería

Algunos de los sectores que hacen un uso intensivo de la tecnología son:

- Industria
  - Fabricación
  - Maquinaria
  - Distribución
- Energía
  - Red inteligente
  - Distribución
- Transporte en todos sus sectores
- Medicina
- Domótica
- Aplicaciones de consumo



Figura 6. Robot aspirador conectado a IoT.

## 4. Programación orientada a objetos: Java

La programación orientada a objetos (POO) es uno de los paradigmas de programación más populares hoy en día. Ello se debe a que posee una serie de propiedades que le confieren gran potencia y flexibilidad, y que fomentan la calidad y reutilización de código.

```
1  /**
2  * @author Javier Sánchez Soriano
3  * @category Introducción
4  */
5 public class Hola {
6
7     public static void main(String[] args) {
8         // Mi primer programa Java
9         System.out.println("Hola Mundo!");
10    }
11
12 }
13 }
```

Hoy en día existen millones de programadores en Java, y la inmensa mayoría de lenguajes de programación previamente existentes han incorporado la orientación a objetos en forma de extensiones para sacar partido de sus propiedades.



**Java es el lenguaje de programación que se va a utilizar para el desarrollo en la asignatura.**

## Tema 1. Introducción a proyecto de ingeniería

### 4.1. Propiedades de la programación orientada a objetos

A modo de repaso se quieren destacar las propiedades más importantes del paradigma de programación orientado a objetos:

- **Modelos orientados a objetos:** identifican las entidades relevantes en una aplicación particular, sus responsabilidades, y cómo interaccionan entre ellas para obtener el comportamiento deseado en la aplicación.
- **Abstracción:** consistente en la capacidad de descomponer un programa en subprogramas o funciones más sencillas, y un tipo de datos en otros más sencillos.
- **Encapsulamiento:** consiste en que los datos llevan asociados las funciones para manejarlos.
- **Ocultación de datos:** los datos no se pueden manipular directamente, sino solo a través de las operaciones o funciones asociadas a los mismos.
- **Clases:** definen las propiedades o estados de tipos de objetos o entidades, y las operaciones admitidas para manipularlos, es decir, su comportamiento.
- **Programas orientados a objetos:** consisten en un conjunto de clases que definen los tipos de los objetos que van a existir en tiempo de ejecución, junto con un programa principal que define cuál es el primer objeto que se crea y cuyos métodos se ejecutan.
- **Herencia:** mecanismo por el cual los objetos de una clase, denominada subclase, disponen de los mismos atributos y métodos de otra clase, denominada superclase.
- **Polimorfismo:** propiedad que permite que el tipo de una referencia varíe durante la ejecución de un programa orientado a objetos.
- **Vinculación dinámica:** consiste en la selección de la implementación de un método durante la ejecución de un programa orientado a objetos, y no durante su compilación.

### 4.2. Propiedades del lenguaje Java

Adicionalmente, se quieren recordar las propiedades más importantes de Java:

- **Totalmente orientado a objetos.** Encapsulación, herencia, polimorfismo, etc., presentes en Java.
- **Multitud de bibliotecas.** Disponibles que permiten realizar cualquier tipo de aplicación.
- **Lenguaje simple.** Con una curva de aprendizaje muy rápida. Más sencillo que C++ al eliminar ciertas características, como los punteros.

## Tema 1. Introducción a proyecto de ingeniería

- **Distribuido.** Proporciona clases para su uso en aplicaciones de red (*sockets*) facilitando la creación de aplicaciones distribuidas.
- **Interpretado y compilado a la vez.** Java es compilado en una especie de código máquina (*bytecodes*, semejantes a las instrucciones de ensamblador) que a su vez se interpretan sobre cualquier máquina virtual de Java (*run-time*).
- **Robusto.** Realiza numerosas comprobaciones en compilación y en tiempo de ejecución. Sus características de memoria liberan a los programadores de una familia entera de errores (la aritmética de punteros), ya que se ha prescindido por completo de los punteros, y la recolección de basura elimina la necesidad de liberación explícita de memoria.
- **Independiente de la arquitectura.** Gracias a la capa intermedia de abstracción entre los programas desarrollados en el lenguaje y el sistema operativo: la máquina virtual de Java (JVM).
- **Portable.** Por ser independiente de la arquitectura.
- **Alto rendimiento.**
- **Multihilo.** Soporta sincronización de múltiples hilos (*multithreading*) a nivel de lenguaje.
- **Dinámico.** El lenguaje Java y su sistema de ejecución en tiempo real son dinámicos en la fase de enlazado. Las clases solo se enlazan a medida que son necesitadas. Se pueden enlazar nuevos módulos de código bajo demanda, procedente de fuentes muy variadas, incluso desde la red.

### 5. Microcontroladores programables: Arduino

Los microcontroladores consisten en circuitos integrados con la capacidad de ser programados. Es por ello que son multipropósito. Su reducido tamaño los hace ideales para multitud de usos y generalmente se integran en placas con todo tipo de circuitería. Son auténticos computadores en miniatura, que contienen todas las unidades funcionales típicas como son CPU, memoria y gestión de entrada/salida.

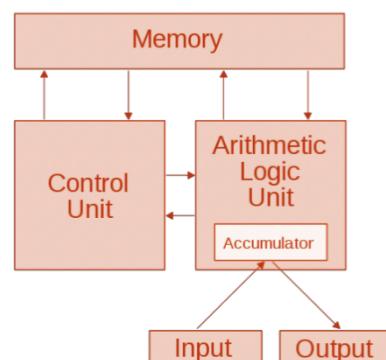
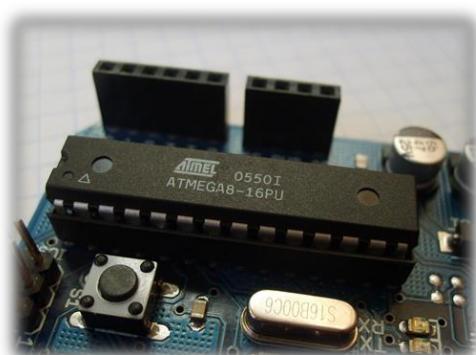


Figura 7. Microcontrolador ATMEGA8 y unidades funcionales de un computador.

# Tema 1. Introducción a proyecto de ingeniería

Este tipo de dispositivos son complejos y requieren de equipo específico (y caro) para hacerlos funcionar y sacarles partido. Ahora bien, desde hace años existen placas sencillas orientadas a todo tipo de público que permiten aprovechar las características de estos microcontroladores (integrados en la placa) de una forma más cómoda y económica. Para ello se rodean los microcontroladores con todo lo necesario para interactuar con sus entradas y salidas, así como para modificar su código de forma amable y sin necesidad de grandes conocimientos de física y electrónica. Un caso bien conocido es la familia de placas Arduino.

## 5.1. ¿Qué es Arduino?

Arduino consiste en una plataforma de código abierto basada en *hardware* y *software* asequible, así como sencillo de usar. Las placas Arduino permiten leer entradas (humedad o temperatura en un sensor, cierre de un interruptor, caudal de un conducto o incluso información de APIs Web) para procesarlas y convertirlas en salidas (activar relé, encender un LED o guardar datos en una tarjeta de memoria SD).

Para programar estas placas se utiliza el lenguaje de programación Arduino (basado en *Wiring*, muy parecido a Java en lo referente a su sintaxis) y el IDE Arduino, basado en Processing.

Existe una gran comunidad de usuarios que comparten y contribuyen al desarrollo del ecosistema Arduino, en torno al cual existen multitud de placas y variantes, así como otros competidores de concepto (*Parallax Basic Stamp*, *BX-24 de Netmedia*, *Phidgets*, *Handyboard de MIT*, *Raspberry Pi*, etc.).



Figura 8. Placa Arduino UNO.

ENTRY LEVEL	UNO   LEONARDO   101   ESPLORA   MICRO   NANO   MINI   MKR2UNO ADAPTER
ENHANCED FEATURES	MEGA   ZERO   DUE   MEGA ADK   MO   MO PRO   MKR ZERO   MOTOR SHIELD USB HOST SHIELD   PROTO SHIELD   MKR PROTO SHIELD   4 RELAYS SHIELD MEGA PROTO SHIELD   MKR RELAY PROTO SHIELD   ISP   USB2SERIAL MICRO USB2SERIAL CONVERTER
INTERNET OF THINGS	YUN   ETHERNET   TIAN   INDUSTRIAL 101   LEONARDO ETH   MKR FOX 1200 MKR WAN 1300   MKR GSM 1400   MKR1000   YUN MINI   YUN SHIELD   WIRELESS SD SHIELD WIRELESS PROTO SHIELD   ETHERNET SHIELD V2   CSM SHIELD V2   MKR IoT BUNDLE
EDUCATION	CTC 101
WEARABLE	GEMMA   LILYPAD ARDUINO USB   LILYPAD ARDUINO MAIN BOARD   LILYPAD ARDUINO SIMPLE LILYPAD ARDUINO SIMPLE SNAP
3D PRINTING	MATERIA 101

BOARDS | MODULES | SHIELDS | KITS | ACCESSORIES | COMING NEXT

Figura 9. Familia de placas y accesorios oficiales del ecosistema Arduino.

## Tema 1. Introducción a proyecto de ingeniería

Las placas Arduino utilizan microcontroladores diferentes en función de sus necesidades (número de puertos de E/S, capacidad de memoria, etc.). Por ejemplo, Arduino *Serial* lleva un ATMEGA8, Arduino *Diecimila* lleva un ATMEGA168, Arduino *Duemilanove* lleva un ATMEGA1280 o el ATMEGA328 (dos versiones de la placa), Arduino *UNO* lleva un ATMEGA 328P, Arduino *Leonardo* lleva un ATMEGA32u4, Arduino *Mega* lleva un ATMEGA 2460, etc.

### 5.1.1. Ventajas de Arduino frente a otras placas

Algunas de las ventajas que ofrece Arduino para los desarrolladores son las siguientes:

- **Económico:** en comparación con otras plataformas de microcontroladores. La versión clónica más barata compatible con Arduino cuesta en torno a los 10 €. También se puede ensamblar a mano.
- **Multiplataforma:** IDE Arduino es compatible con Windows, Macintosh OSX y Linux. La mayoría de los sistemas de microcontroladores están limitados a Windows.
- **IDE sencillo y claro:** fácil de usar y suficientemente flexible.
- **SW de código abierto:** publicado como herramientas de código abierto, disponibles para extensión por programadores experimentados.
- **SW extensible:** puede expandirse mediante bibliotecas codificadas en C++.
- **HW de código abierto:** los diseños de las placas Arduino se publican bajo una licencia *Creative Commons*, por lo que los diseñadores experimentados de circuitos pueden crear su propia versión del módulo, ampliarlo y mejorarlo.

## 6. Herramientas para el desarrollo de proyectos

Para el desarrollo del proyecto (que va a suponer el eje central en la asignatura Proyecto de Ingeniería) se van a utilizar 4 herramientas básicas, divididas en dos categorías: desarrollo de *software* y gestión de proyectos. Además de estas herramientas principales, durante el curso se introducirán otras aplicaciones complementarias, así como extensiones de estas.

En lo relativo a la gestión, se quiere introducir a los alumnos en el uso de herramientas ágiles utilizadas por la industria. El propósito es comprender que su uso resulta vital dado que ayuda a los miembros del equipo a conseguir alcanzar los hitos del proyecto en tiempo y forma. En futuros cursos, el alumno adquirirá más experiencia en la necesidad de definir y utilizar procesos en el desarrollo de *software*, que son la base de toda un área de la profesión: la ingeniería del *software*.

### 6.1. Desarrollo *software*

Al trabajar con dos lenguajes de programación distintos se van a requerir dos entornos diferentes:

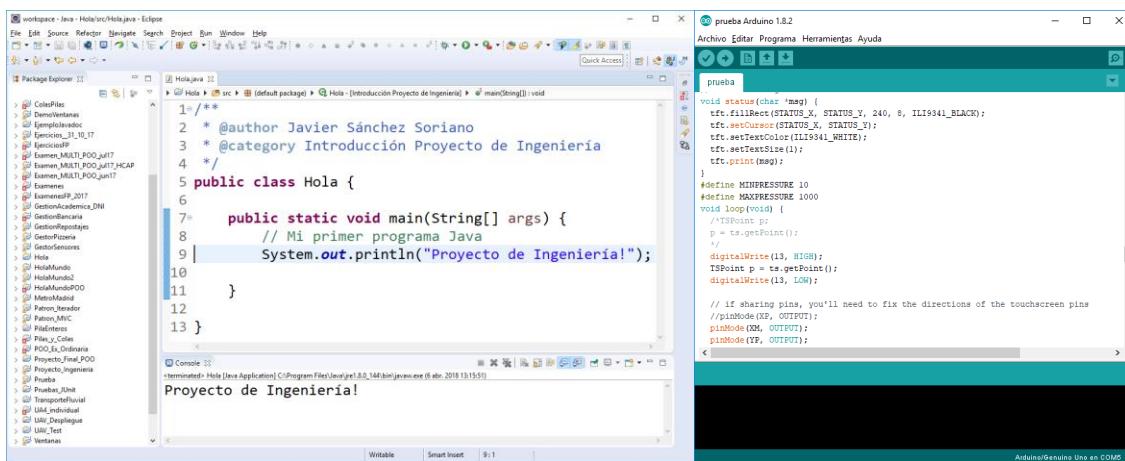
# Tema 1. Introducción a proyecto de ingeniería

## 6.1.1. Eclipse IDE

Eclipse es un entorno de desarrollo integrado para desarrollar en Java (aunque mediante extensiones también permite trabajar con C, C++, PHP, Python, ...). Tiene un mercado de aplicaciones/complementos llamado Eclipse Marketplace, entre otras características interesantes. Además, ha sido utilizado en las asignaturas previas de programación: Fundamentos de Programación y Programación Orientada a Objetos.

## 6.1.2. Arduino IDE

Arduino IDE es un *software* de código abierto que permite editar código, compilarlo y cargarlo en las placas Arduino. Es compatible con Windows, Mac OS X y Linux. Está escrito en Java y está basado en Processing, que es también *software* de código abierto. Resulta muy sencillo y con muy pocas opciones, de modo que tiene una curva de aprendizaje muy corta.



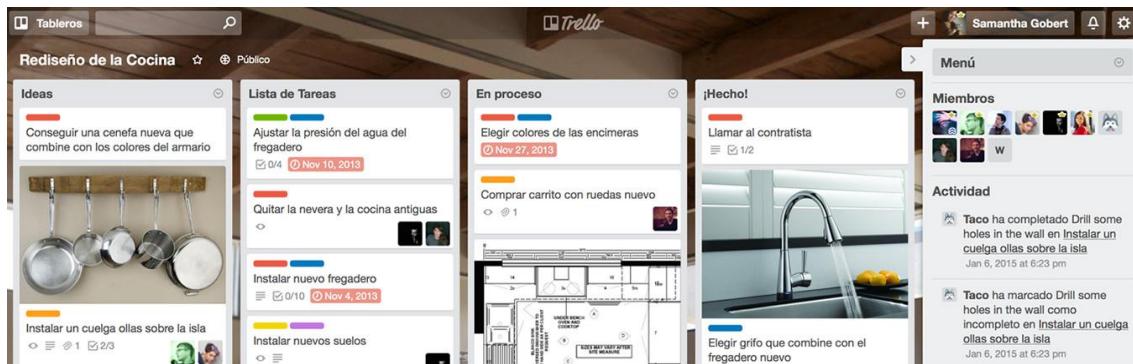
## 6.2. Gestión de proyectos

Como se ha adelantado, es importante utilizar metodologías que garanticen una comunicación y reparto de tareas entre los miembros del equipo. Para ello se propone utilizar las herramientas siguientes:

### 6.2.1. Trello

Trello es un tablón en el que se pueden colgar ideas, tareas, imágenes o enlaces. Es apropiado para cualquier tipo de tarea que requiera organizar información y trabajar en equipo. Emplea el sistema Kanban para el registro de actividades mediante tarjetas virtuales que permiten organizar tareas, agregar listas, adjuntar archivos, etiquetar eventos, agregar comentarios y compartir tableros.

## Tema 1. Introducción a proyecto de ingeniería



### 6.2.2. GitHub

GitHub es una plataforma de desarrollo colaborativo que permite alojar proyectos utilizando el sistema de control de versiones Git. Se utiliza principalmente para la creación de código fuente. El código de los proyectos alojados en GitHub se almacena típicamente de forma pública, aunque utilizando una cuenta de pago o una con fines educativos también permite hospedar repositorios privados.

## 7. Resumen

En este tema introductorio se han repasado las tecnologías y herramientas fundamentales que permitan al alumno afrontar el reto de desarrollar un sistema compuesto por elementos *hardware* y *software* destinado a resolver un problema real en forma de proyecto de ingeniería en la industria del IoT. También se ha hecho hincapié en la metodología de trabajo que se va a seguir: aprendizaje basado en proyectos, donde los alumnos van a liderar su aprendizaje acompañados por el equipo docente.

En concreto se han abordado los siguientes aspectos:

- Las bases del aprendizaje basado en proyectos (PBL) o por qué resulta interesante.
  - Qué hace cada uno de los roles: alumno y profesor.
- El concepto de Internet de las cosas (IoT).

## Tema 1. Introducción a proyecto de ingeniería

- Repasar los principales conceptos de la programación orientada a objetos.
- Comprender la utilidad de microcontroladores programables.
  - En concreto de la familia Arduino.
  - Con vistas a permitir realizar desarrollos IoT.
- La utilidad de las herramientas para desarrollo de proyectos.
  - Eclipse IDE para desarrollar en Java.
  - Arduino IDE para desarrollar en las placas Arduino.
- La utilidad de las herramientas para gestión de proyectos.
  - Trello para gestión de tareas.
  - GitHub como repositorio para compartir código.

### Referencias bibliográficas

Schildt, H. (2014). *Java: a beginner's guide*. Nueva York: McGraw-Hill Education. Disponible en [https://descubre-uem.bibliocrai.universidadeuropea.es/cgi-bin/koha/opac-detail.pl?biblionumber=86989&query\\_desc=kw%2Cwrdl%3A%20java](https://descubre-uem.bibliocrai.universidadeuropea.es/cgi-bin/koha/opac-detail.pl?biblionumber=86989&query_desc=kw%2Cwrdl%3A%20java)

Banzi, M. (2009). *Getting started with Arduino*. Pekín: O'Reilly. Disponible en [https://descubre-uem.bibliocrai.universidadeuropea.es/cgi-bin/koha/opac-detail.pl?biblionumber=11901&query\\_desc=kw%2Cwrdl%3A%20arduino](https://descubre-uem.bibliocrai.universidadeuropea.es/cgi-bin/koha/opac-detail.pl?biblionumber=11901&query_desc=kw%2Cwrdl%3A%20arduino)

Oxer, J. y Blemings, H. (2009). *Practical Arduino: cool projects for open source hardware*. Nueva York: Apress. Disponible en [https://descubre-uem.bibliocrai.universidadeuropea.es/cgi-bin/koha/opac-detail.pl?biblionumber=11902&query\\_desc=kw%2Cwrdl%3A%20arduino](https://descubre-uem.bibliocrai.universidadeuropea.es/cgi-bin/koha/opac-detail.pl?biblionumber=11902&query_desc=kw%2Cwrdl%3A%20arduino)

Reese, G. (2001). *Programación de bases de datos con JDBC y JAVA*. Madrid: Anaya multimedia. Disponible en [https://descubre-uem.bibliocrai.universidadeuropea.es/cgi-bin/koha/opac-detail.pl?biblionumber=10664&query\\_desc=kw%2Cwrdl%3A%20bases%20datos%20java](https://descubre-uem.bibliocrai.universidadeuropea.es/cgi-bin/koha/opac-detail.pl?biblionumber=10664&query_desc=kw%2Cwrdl%3A%20bases%20datos%20java)



© Todos los derechos de propiedad intelectual de esta obra pertenecen en exclusiva a la Universidad Europea de Madrid, S.L.U. Queda terminantemente prohibida la reproducción, puesta a disposición del público y en general cualquier otra forma de explotación de toda o parte de la misma.

La utilización no autorizada de esta obra, así como los perjuicios ocasionados en los derechos de propiedad intelectual e industrial de la Universidad Europea de Madrid, S.L.U., darán lugar al ejercicio de las acciones que legalmente le correspondan y, en su caso, a las responsabilidades que de dicho ejercicio se deriven.