Propuesta de Trabajo de Inserción Profesional

*Título:*

Integración de tecnología para control y automatización de espacio físico*.*

*Alumno:*

*Martin Alejandro Melo.*

*Director:*

*Ing. José Luis Di Biase.*

*Carrera:*

*Tecnicatura en Programación Informática.*



Contenido

[Introducción 3](#_Toc402720897)

[Contexto 3](#_Toc402720898)

[Problemática 3](#_Toc402720899)

[Solución 3](#_Toc402720900)

[Desarrollo de la solución propuesta 4](#_Toc402720901)

[Descripción/Contexto General 4](#_Toc402720902)

[Automatización y control de los espacios físicos 4](#_Toc402720903)

[Internet Of Things 5](#_Toc402720904)

[Hardware 6](#_Toc402720905)

[Como nace 6](#_Toc402720906)

[¿Qué es? 7](#_Toc402720907)

[¿Qué resuelve? 8](#_Toc402720908)

[Base 8](#_Toc402720909)

[Sensores específicos 9](#_Toc402720910)

[Componentes específicos 9](#_Toc402720911)

[Diagramas de conexión 10](#_Toc402720912)

[Software 11](#_Toc402720913)

[Back-End 12](#_Toc402720914)

[Midle-End 12](#_Toc402720915)

[Front-End 12](#_Toc402720916)

[Firmware 13](#_Toc402720917)

[MQTT 13](#_Toc402720918)

[Mosca 14](#_Toc402720919)

[Forma de comunicación 14](#_Toc402720920)

[Conclusiones 14](#_Toc402720921)

[Resultados Obtenidos 14](#_Toc402720922)

# Introducción

## Contexto

En la actualidad cada vez es mas común ver dispositivos inteligentes en nuestras vidas.

## Problemática

Porque estamos haciendo este TIP. Que viene a solucionar. Tratar el problema de la “Desconexión entre la automatización y la información”

## Solución

Crear un dispositivo electrónico independiente, uso de protocolos de comunicación y desarrollo de Front End de web

# Desarrollo de la solución propuesta

## Descripción/Contexto General

### Automatización y control de los espacios físicos

Es también conocida como Domotica, la cual es un conjunto de sistemas capaces de automatizar una vivienda, aportando servicios de gestión energética, seguridad, bienestar y comunicación, y que pueden estar integrados por medio de redes interiores y exteriores de comunicación, cableadas o inalámbricas, y cuyo control goza de cierta ubicuidad, desde dentro y fuera del hogar. Se podría definir como la integración de la tecnología en el diseño inteligente de un recinto cerrado.

### Internet Of Things



El Internet de las cosas o Internet Of Things es un término que se relacionan mucho con la web. Pero también resulta un concepto un poco abstracto para los no iniciados pese a que ha estado ganando bastante popularidad en el último tiempo. La idea que intenta representar queda bastante bien ilustrada por su nombre, cosas cotidianas que se conectan a Internet, pero que en realidad se trata de mucho más que eso.  
  
Para entender de qué va el internet de las cosas debemos también comprender que sus fundamentos no son en lo absoluto nuevos. Desde hace unos 30 años que se viene trabajando con la idea de hacer un poco más interactivos todos los objetos de uso diario. Planteos como el hogar inteligente o casa del mañana, han evolucionado antes de que nos demos cuenta en el hogar conectado para entrar al internet de las cosas.  
  
El Internet de las cosas potencia objetos que antiguamente se conectaban mediante circuito cerrado, como comunicadores, cámaras, sensores, y demás, y les permite comunicarse globalmente mediante el uso de la red de redes.  
  
Si tuviéramos que dar una definición del internet de las cosas probablemente lo mejor sería decir que se trata de una red que interconecta objetos físicos valiéndose de la red. Los mentados objetos se valen de sistemas embebidos, o lo que es lo mismo, hardware especializado que le permite no solo la conectividad a internet, sino que además programa eventos específicos en función de las tareas que le sean dictadas remotamente (por ejemplo desde nuestro celular mientras trabajamos en la oficina).

## Hardware

El hardware principal estará basado en la plataforma Arduino.

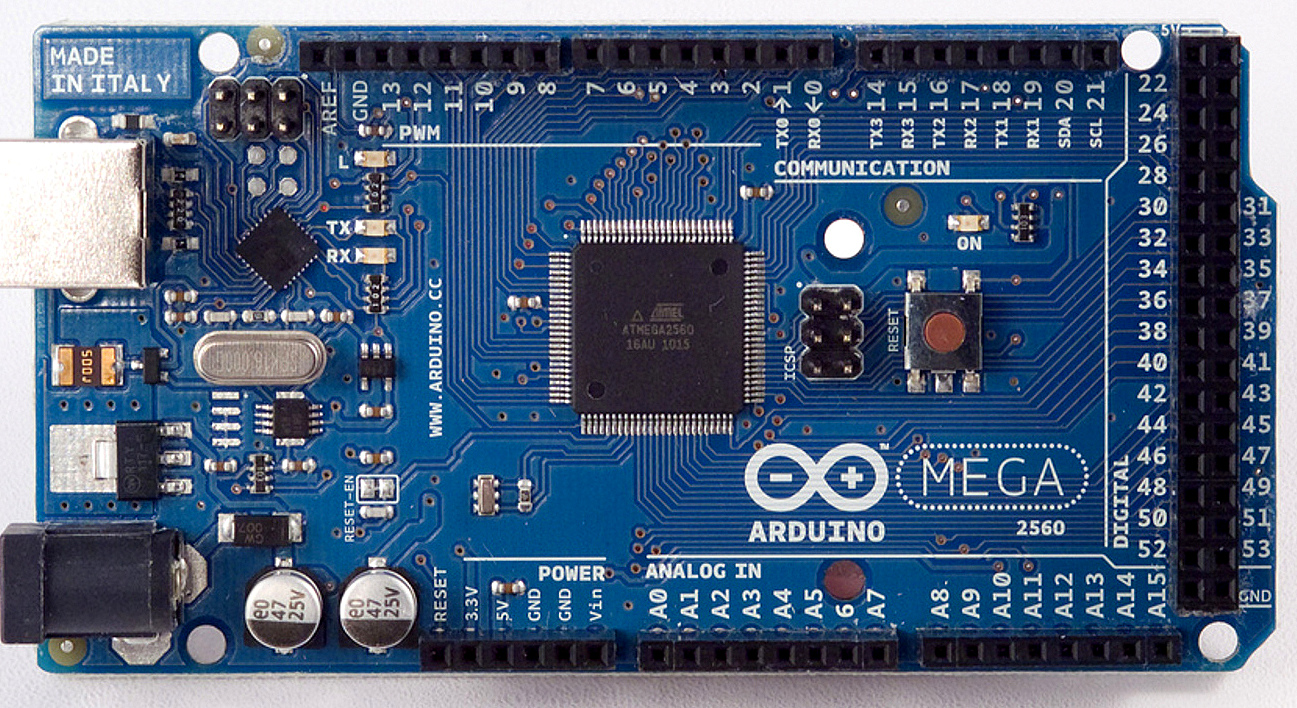
### Como nace

La creación de Arduino se inicio en el año 2005 y fue un proyecto para los estudiantes de un instituto en Ivrea Italia. Este proyecto surgió de la necesidad de contar con un dispositivo de bajo costo y que se pueda utilizar con cualquier sistema operativo.

### ¿Qué es?

*Arduino* es una plataforma de hardware libre, basada en una placa con un microcontrolador y un entorno de desarrollo, diseñada para facilitar el uso de la electrónica en proyectos multidisciplinares.

El hardware consiste en una placa con un microcontrolador [Atmel AVR](http://es.wikipedia.org/wiki/AVR). La siguiente imagen pertenece una placa Arduino Mega2650.



Actualmente existen muchos entornos de desarrollos para crear programas para Arduino, los cuales se encuentran programados en diferentes lenguajes como C, Java, y Python.

La sintaxis del lenguaje de programación Arduino es una versión simplificada de C/C++.

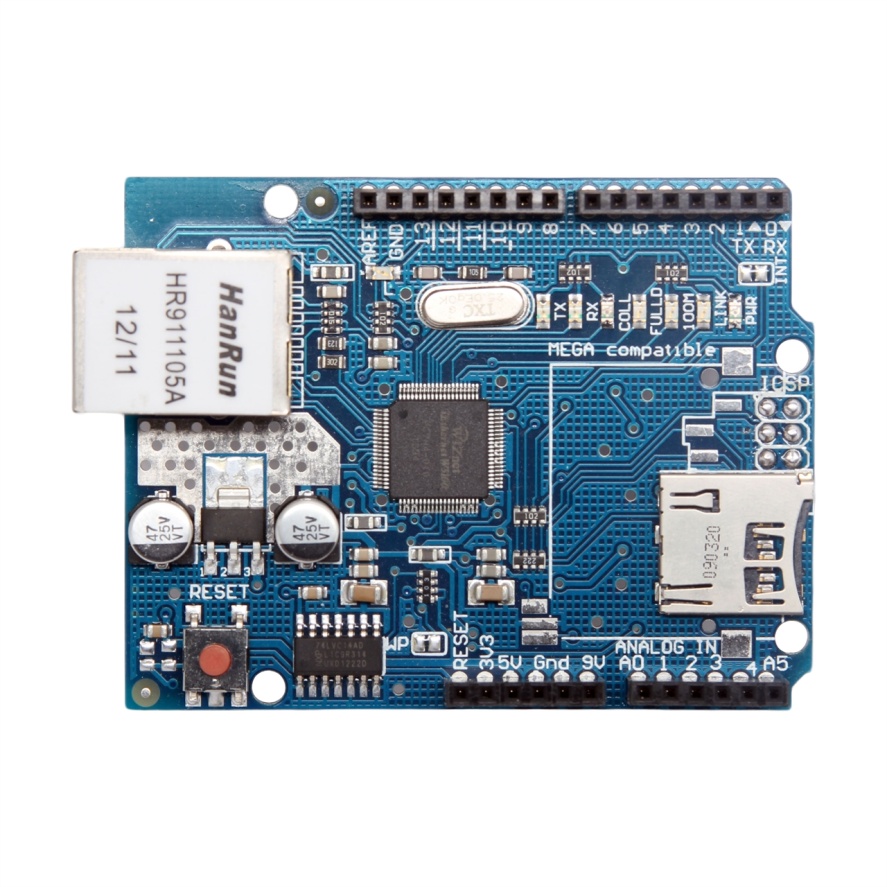
### ¿Qué resuelve?

Se utilizo para crear un dispositivo capaz de poder comunicarse en ambos sentidos con la aplicación que utiliza el usuario y que a su vez se pueda utilizar para efectuar distintas acciones basándose en la información que se le suministre atravez de un protocolo especifico. A su vez, el mismo está conectado a distintos circuitos electrónicos que le proveen efectuar las acciones anteriormente mencionadas y también la obtención de información de los sensores que posea.

### Base

Los componentes de base necesarios para que el sistema funcione son:

* Arduino Mega2560.
* Ethernet Shield con chipset Wiznet W5100.



### Sensores específicos

El sensor de temperatura utilizado en el circuito impreso es un LM35DZ. Cuya hoja de datos es la siguiente: <http://www.ti.com.cn/cn/lit/ds/symlink/lm35.pdf>

### Componentes específicos

Los componentes específicos utilizados son los siguientes:

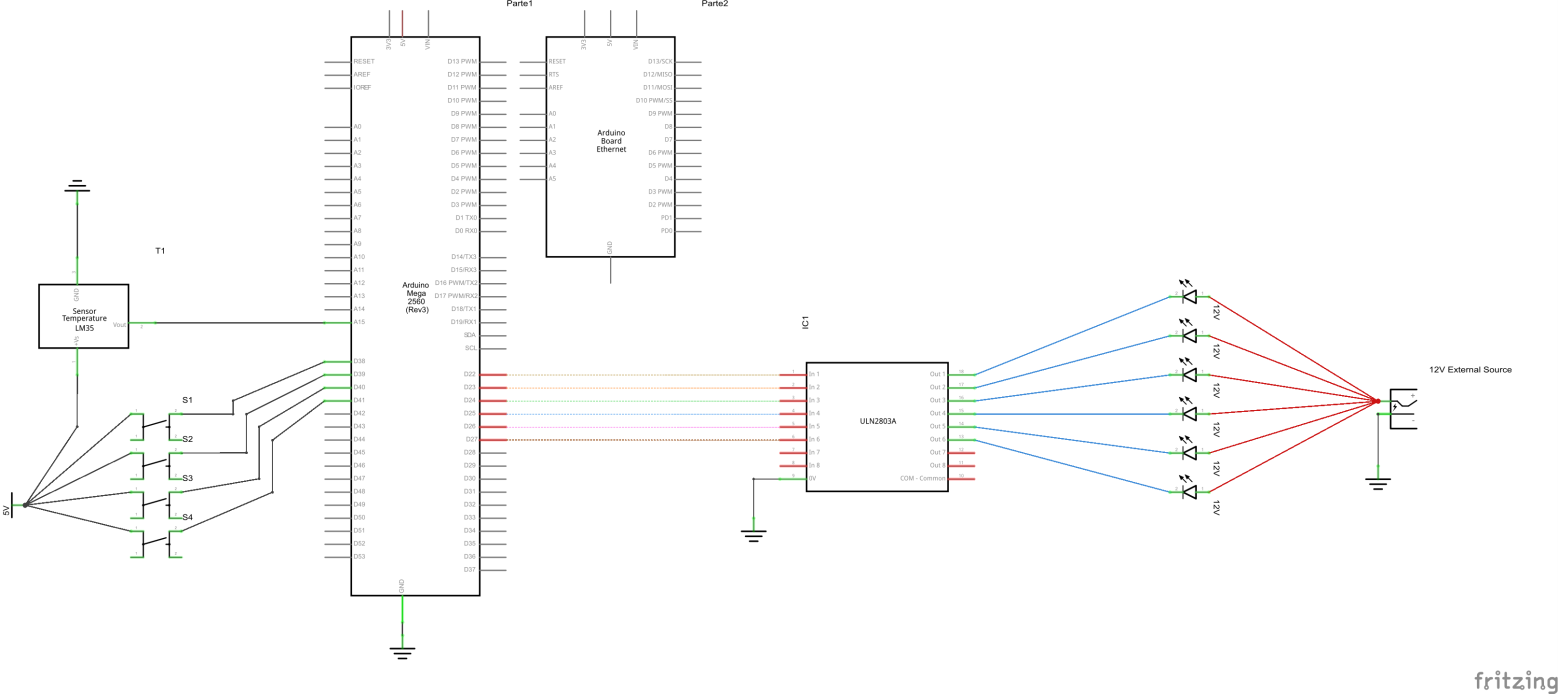
* ULN2803. Es un array de Darlington que se utiliza para controlar las luces que se encuentran conectadas al circuito de las luces. Su hoja de datos es la siguiente: [http://www.ti.com/lit/ds/symlink/uln2803.pdf](http://www.ti.com/lit/ds/symlink/uln2803a.pdf).
* Botones de estado normal abierto. Cada botón se utiliza para simular apertura de puertas.
* Resistencia 1K ohm. Se utilizan para eliminar el efecto "antena" de los cables que se conectan a las entrada del Arduino, estas resistencia van junto a los botones.

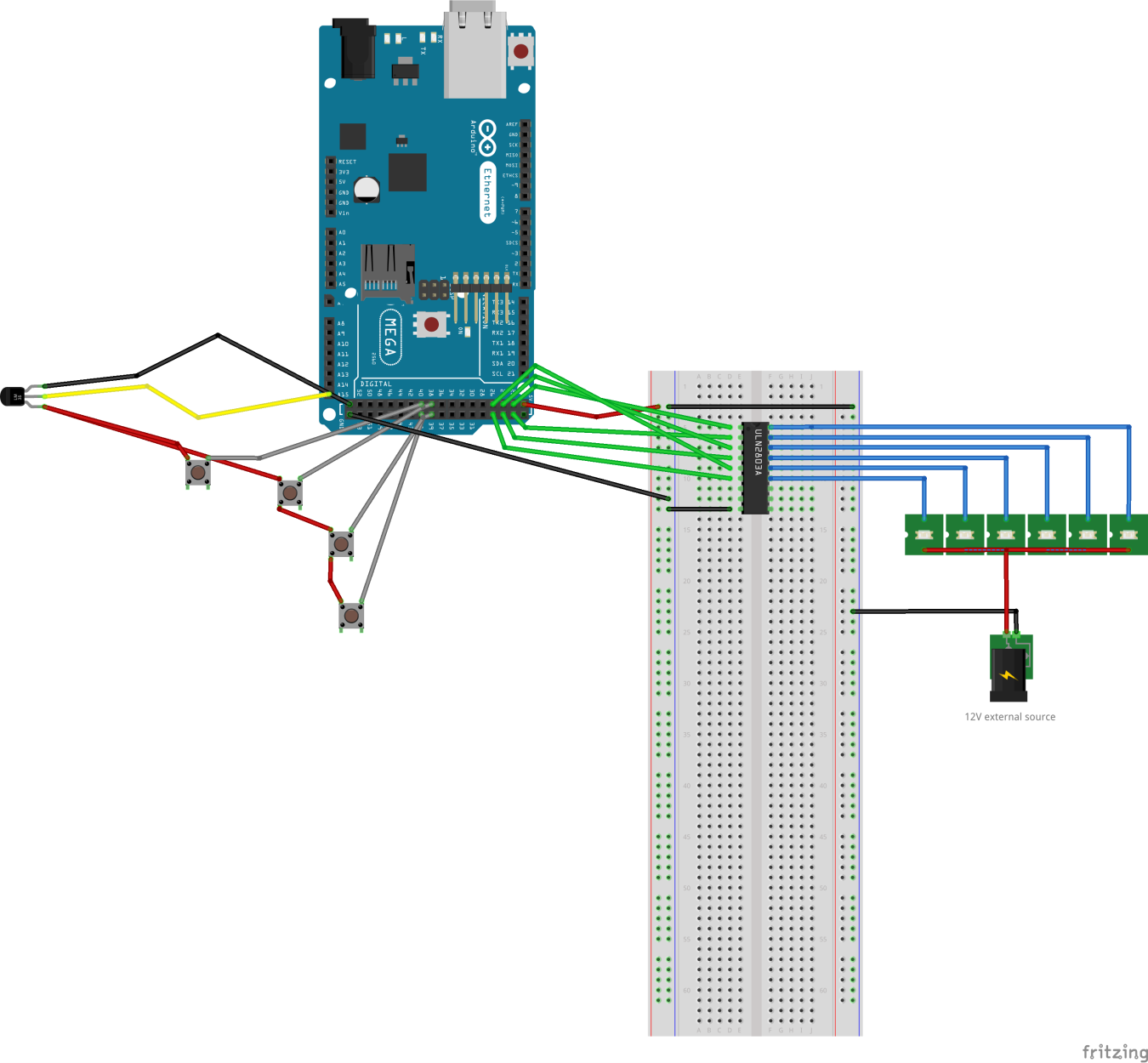
Todos estos componentes son de bajo costo y pueden ser adquiridos en cualquier local donde vendan componentes electrónicos.

### Diagramas de conexión

Las siguientes imágenes pertenece al esquema de conexión de los componentes específicos que conforman el hardware.

Estas imágenes se encuentran elaboradas utilizando el programa Fritzing. La primera de ellas pertenece al esquema de conexión de los componentes al microcontrolador. La segunda imagen pertenece al esquema de conexión de los componentes utilizando un protoboard, el cual es un tablero con orificios conectados eléctricamente entre sí, en el cual se pueden insertar componentes electrónicos y cables para el armado y prototipado de circuitos electrónicos y sistemas similares.





## Software

El software que compone a la aplicación se encuentre separado en 3 partes. Estas son : Back-End, Midle-End y Front-End

Tanto el Back-End como el Front-End se encuentran construidas a partir del generador MeanJs. El cual permite generar aplicaciones web utilizando MongoDB, Express, AngularJS y NodeJS.

### Back-End

Este se encuentra programado principalmente utilizando NodeJS, el cual está basado en Javascript. También se utilizan las siguientes librerías y Frameworks:

* MongoDB.
* Mongoose.
* Express.
* node-schedule.

### Midle-End

Este se encuentra programado en NodeJS y utiliza las siguientes librerías y Frameworks:

* Mqtt.
* Socket.io.
* Mosca.

### Front-End

Este se encuentra programado principalmente utilizando AngularJS, un Framework Javascript. También se utilizaron las siguientes librerías:

* Jquery.
* Bootstrap.
* socket.io-client.
* malhar-angular-widgets.
* malhar-angular-dashboard.
* font-awesome.
* ngDialog.
* Later.
* Schedule.
* es5-shim.
* d3.

## Firmware

El firmware que se utiliza en el microcontrolador está programado en el lenguaje C. En el mismo se incluyen las siguientes librerías:

* MQTT:
* Mosca.
* etc.

### MQTT

[Message Queue Telemetry Transport (MQTT)](http://mqtt.org/) es un protocolo de conectividad enfocado a M2M (machine-to-machine) y al IOT (Internet of Things) ya que se ha diseñado para ser un protocolo de mensajería extremadamente ligero basado en TCP.

Es útil para conexiones con sitios remotos donde se posee una alta latencia y el ancho de banda es muy importante.

Una característica muy importante es que al ser un protocolo tan ligero existen clientes y servidores MQTT en diversos lenguajes.

### Mosca

Es un bróker de mensajería, con lo cual es un mecanismo mediador de la comunicación entre aplicaciones y/o dispositivos, permitiendo minimizar el grado de conocimiento mutuo que estos necesitan tener, para poder intercambiar mensajes, implementando así efectivamente su [desacoplamiento](http://es.wikipedia.org/wiki/Dise%C3%B1o_estructurado#Acoplamiento).

El propósito del bróker es recibir los mensajes entrantes desde las aplicaciones y llevar a cabo determinadas acciones con ellas.

## Forma de comunicación

Las distintas partes de la aplicación se comunican de la siguiente forma:

Meter una imagen súper linda de cómo está unido todo.

# Conclusiones

### Resultados Obtenidos