aWareHouse

Environment Control System for Warehouses

MSc in Informatics and Computer Engineering
Programming Paradigms
EIC0065-2S

Duarte Nuno Pereira Duarte - 201109179 (ei11101@fe.up.pt)
Hugo José Freixo Rodrigues - 201108059 (ei11086@fe.up.pt)
João Pedro Matos Teixeira Dias - 201106781 (ei11137@fe.up.pt)

Faculdade de Engenharia da Universidade do Porto
Rua Roberto Frias, sn, 4200-465 Porto, Portugal

May 13, 2015

Abstract

We are in the rise of IoT (Internet of things). A world were everything is connected and we can, with simple tools, monitory and control everything. In this context, there is a lot of space for a more recurrent use of different programming paradigms because of the need of interaction with different layers of system architecture for a single application, since hardware to web.

Our application, *aWareHouse*, was designed with the objective of, with a simple interface, we can monitory a house or a warehouse in terms of environment conditions (temperature, humidity, sound and luminosity).

For accomplishing this was used a combination of hardware/software and some different programming languages, in a way that gave us a stable application that can be used for setting a alarm system when occurs changes in our environment, for taking decisions analysing the past conditions and the relations with external (meteorological) conditions or simple look at the current conditions inside our warehouse.

Contents

1. **Introduction**

The *aWareHouse* project was developed for the Programming Paradigms course unit of the Master in Informatics and Computing Engineering at FEUP. The project was developed with the objective of combine, at least, three programming paradigms in the same application.

The base motivation for this project was the Internet of Things, as defined by *Gartner*:

“The Internet of Things (IoT) is the network of physical objects that contain embedded technology to communicate and sense or interact with their internal states or the external environment.”

So, we designed a low cost system capable of giving the user the possibility of, using a relative small hardware box, monitory the environment of a given place like a home or warehouse. This, associated with a system capable of maintain records of the past conditions of the environment (plus external weather conditions), results in one application that gives the user the capacity of taking decision, be aware of the environment status using alerts and see the current conditions.

As explained before the capabilities of this application make this a system useful in a lot of situations, for example, the monitoring of a warehouse and verify the relation between weather and internal conditions to make decision on what is the best settings for a refrigeration system and assure that the, for example, temperature is always below the maximum recommended. Other example is simple use it as a house control system and, for example, verify if someone forgot to turn any light off.

2. System Description

2.1 Conceptual Description

2.1.1 Functionalities

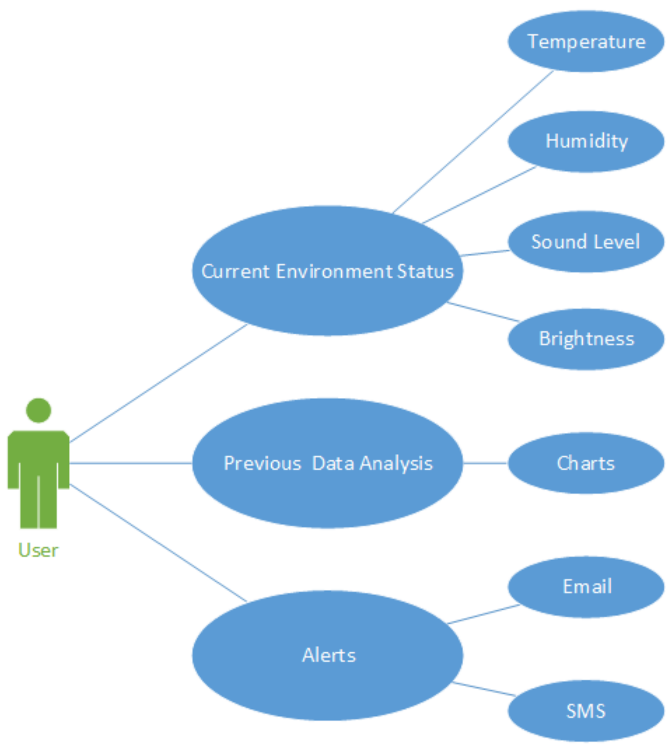

Figure 2.1: Use Case diagram.

2.1.2 Architecture

2.1.2.1 Physical Architecture

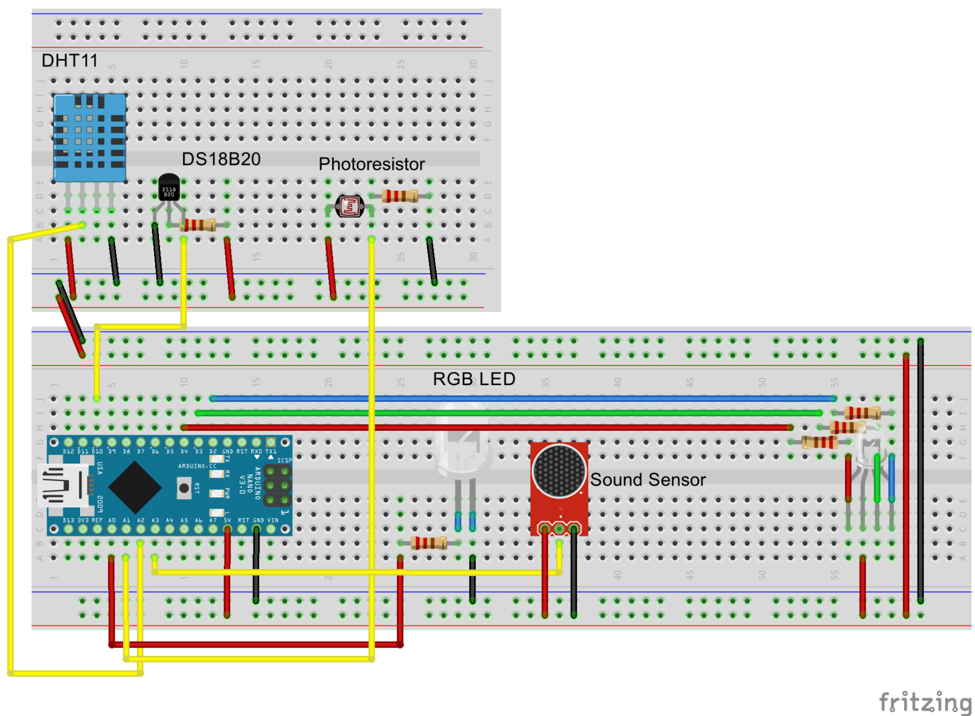

Figure 2.2: Circuit diagram.

Sensor	Description	Input/Output
DHT11	Basic and low-cost digital temperature and humidity sensor. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air, and spits out a digital signal on the data pin.	Temperature and Humidity values.
DS18B20	1-wire digital temperature sensor fairly precise (+- 0.5°C over much of the range) and can give up to 12 bits of precision from the onboard digital-to-analog converter.	Temperature values.
LM393	One single channel output sound level. Low level output signal and when there is sound output low, lights lit.	Sound level values.
Diffused LED	A diffuse LED with with separate red, green and blue LED chips inside, capable of emitting a color-mix resulted of the values passed to each chip.	RGB Color or Color-Mix.
Photo cell	Photo cell or CdS photoresistor is a little light sensor. As the squiggly face is exposed to more light, the resistance goes down and the voltage goes up.	Voltage value.

Table 2.1: Sensors description.

Arduino

Figure 2.3: Arduino micro-controller.

The Arduino Nano (fig.2.3) is a small, complete, and breadboard-friendly micro-controller based on the ATmega328 processor chip and works with a Mini-B USB cable for energy and data transfer. It has 32 KB of flash memory space of which 2 KB used by bootloader.

Each of the 14 digital pins on the Nano can be used as an input or output and 8 analog inputs, each of which provide 10 bits of resolution (i.e. 1024 different values).

The Arduino provides an UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An FTDI FT232RL on the board channels this serial communication over USB and the FTDI drivers (included with the Arduino software) provide a virtual COM port to software on the computer.

Raspberry Pi

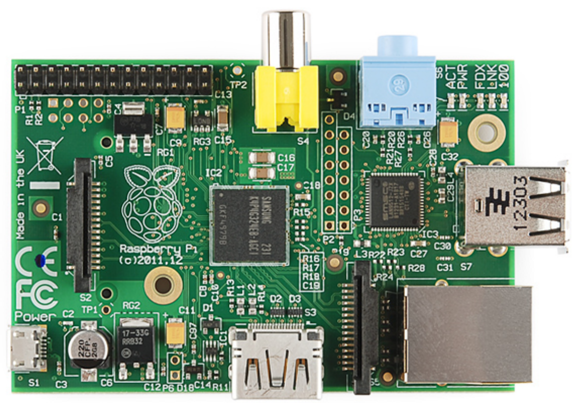

Figure 2.4: Raspberry Pi 1 Model B.

The *Raspberry Pi 1 Model B* (fig. 2.4) is a single-board computer which can be used for many of the things that a desktop is used to.

The design is based around a Broadcom BCM2835 SoC, which includes an ARM1176JZF-S 700 MHz processor, VideoCore IV GPU, and 512 Megabytes of RAM. The memory used is a SD card for booting and long-term storage. This board is intended to run Linux kernel based operating systems. Additionally this has two USB ports and a 10/100 Ethernet controller.

2.1.2.2 Logic Architecture

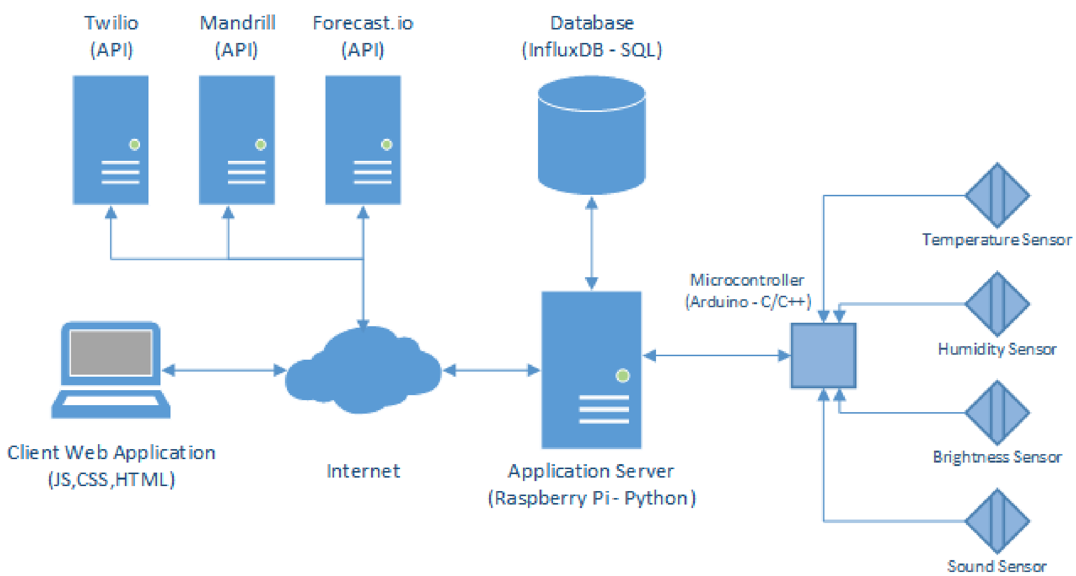

Figure 2.5: Architecture diagram.

2.1.3 Programming Languages and Technologies

In the context of realizing this project it was conciliated a diversity of technologies and programming languages that, working together in the architecture presented at fig. 2.5, made the application possible. Starting by enumerating the programming languages used and the reasons for choosing those languages is presented in the table 2.2.

Arduino	C/C++
Python	item 2
Web Languages	HTML,JS,CSS
SQL	InfluxDB

Table 2.2: Programming languages.

In addiction to the programming languages we used a set of technologies as a part of our application as presented in table 2.3.

InfluxDB	
Grafana	
Web API's	Twilio, Mandrill, Forecast.io

Table 2.3: Technologies.

-Python lib's (?)
-Arduino lib's (?)

2.2 Implementation Description

2.2.1 Implementation Details

Socket communication over usb (Python - Arduino (c/c++))

HTTP REST protocol (web application - python server - InfluxDB - external API's - JSON)

2.2.2 Development Environment

Raspberry Pi, Arduino, Arduino IDE + Compiler, Atom, SSH, Python Compiler, ARM architecture , Version Control - Git

3. Conclusion

almost ready to market (?), easy to expand or adapt, usefulness of programming paradigms, big data and internet of things

4. Improvements

Motion, better management of storage (slow), advice system (?), better safety (credentials)

5. **Resources**

5.1 *Bibliography*

Visual Studio 2013 Ultimate, Microsoft, `http://www.visualstudio.com/`.

5.2 *Software*

Visual Studio 2013 Ultimate, Microsoft, `http://www.visualstudio.com/`.

Appendices

A. Appendix

A.1 Hardware

A.1.1 Raspberry Pi

Chip Broadcom BCM2835 SoC full HD multimedia applications processor
CPU 700 MHz Low Power ARM1176JZ-F Applications Processor
GPU Dual Core VideoCore IV® Multimedia Co-Processor
Memory 512MB SDRAM
Ethernet onboard 10/100 Ethernet RJ45 jack
USB 2.0 Dual USB Connector
Video Output HDMI (rev 1.3 & 1.4) Composite RCA (PAL and NTSC)
Audio Output 3.5mm jack, HDMI
Onboard Storage SD, MMC, SDIO card slot
Operating System Linux
Dimensions 8.6cm x 5.4cm x 1.7cm

A.1.2 Arduino

Specifications:
Microcontroller Atmel ATmega328
Operating Voltage (logic level) 5 V
Input Voltage (recommended) 7-12 V
Input Voltage (limits) 6-20 V
Digital I/O Pins 14 (of which 6 provide PWM output)
Analog Input Pins 8
DC Current per I/O Pin 40 mA
Flash Memory 32 KB (ATmega328) of which 2 KB used by bootloader
SRAM 2 KB (ATmega328)
EEPROM 1 KB (ATmega328)
Clock Speed 16 MHz
Dimensions 0.73" x 1.70"
Length 45 mm
Width 18 mm

Weigth 5 g

A.1.3 Sensor DHT11

Technical specs:

- Low cost
- 3 to 5V power and I/O
- 2.5mA max current use during conversion (while requesting data)
- Good for 20-80% humidity readings with 5% accuracy
- Good for 0-50C temperature readings +- 2C accuracy
- No more than 1 Hz sampling rate (once every second)
- Body size 15.5mm x 12mm x 5.5mm
- 4 pins with 0.1” spacing

A.1.4 Sensor DS18B20

Technical specs:

- Usable temperature range: -55 to 125 C (-67F to +257F)
- 9 to 12 bit selectable resolution
- Uses 1-Wire interface- requires only one digital pin for communication
- Unique 64 bit ID burned into chip
- Multiple sensors can share one pin
- +-0.5C Accuracy from -10C to +85C
- Temperature-limit alarm system
- Query time is less than 750ms
- Usable with 3.0V to 5.5V power/data

A.1.5 Sound Sensor (LM393)

Technical specs:

- LM393 (?)
- Electret microphone
- Working voltage: DC 4 6 V
- With a signal output instruction
- One single channel output
- Low level output signal
- When there is sound output low, lights lit

A.1.6 Diffused Led

Technical specs:

- 10mm diameter
- Red: 623 nm wavelength, Green: 523 nm, Blue: 467 nm
- Red: 1.8-2.2V Forward Voltage, at 20mA current, Green: 3.0-3.4V, Blue: 3.0-3.4V
- 50 degree viewing angle.
- Red: 700 mcd typical brightness, Green: 2100 mcd, Blue: 900 mcd

A.1.7 Photo cell (CdS photoresistor)

Technical specs:

- When its light, the resistance is about 5-10KOhm, when dark it goes up to 200KOhm.
- The voltage on the pin will be 2.5V or higher when its light out and near ground when its dark.