Development of a datalogger with open source hardware and software for the study of desert animals

Daniel Díaz[a,b], Sandra Oviedo[a,b], Ramiro Madriaga[b], Alejandra Otazu[b], Francisco Ibañez[b]

[a] LIAI- Instituto de Informática- [b]Dpto. de Informática FCEFyN - Universidad Nacional de San Juan CUIM - Av. Ignacio de la Roza 590 (O), Rivadavia - J5402DCS San Juan {ddiaz, soviedo, rmadriaga, otazuale, fibannez }@iinfo.unsj.edu.ar

Abstract. The territory of the province of San Juan is composed by 80 percent by mountains. Its climate is quasi desert. It is in this place, where a group of researchers from the San Juan National University, is dedicated to the study of amphibians. To accomplish this task a device that can record data is necessary. These data are some environmental variables and the sound that these animals produce at certain temporal points. Generically to this device type is denominated data logger. This article describes the process of developing a datalogger of technical characteristics that have been specified by biologist researchers. It is a device built with open source hardware and software. The capture and data processing is done with the Arduino platform. The human computer interface is developed using the Android platform. That is, the management of datalogger is carried out by a mobile device, tablet or Smartphone based on Android
Keywords: Innovate products, open source hardware, methodology, datalogger

1 Introduction

A datalogger is an electronic device that records data over time or in relation to location either with a built in instrument or sensor or via external instruments and sensors. The datalogger are often used in biology experiments and researches, as it is case of researchers that are dedicated to study of amphibians. Here, recording some environmental variables and the sound of amphibians (frogs and toads) is very useful for studying taxonomy and systematics of these species. This type of devices are present on the market, but almost all of them have generic functionalities. Also, the cost of datalogger is a problem. A research project needs an important amount of dataloggers. Itis impossible accomplish this requirement with the project budget of a typical research project of Argentine universities.

Commonly, the technologies to develop a device such features come from electronic engineering. New technologies have emerged in recent years that allow build such devices. These technologies are known as open source hardware and software platforms[1], [2], [3] y [4]. This paper describes the engineering process that allowed the development of a datalogger to prototype level using these technologies. This

device allows biologists researchers to record the sounds produced amphibians under different environmental conditions (temperature, humidity, sunlight, etc.).

2 Fundaments

The OSHWA(Open Source Hardware Association) statement of principles say “Open source hardware is hardware whose design is made publicly available so that anyone can study, modify, distribute, make, and sell the design or hardware based on that design. The hardware's source, the design from which it is made, is available in the preferred format for making modifications to it. Ideally, open source hardware uses readily-available components and materials, standard processes, open infrastructure, unrestricted content, and open-source design tools to maximize the ability of individuals to make and use hardware. Open source hardware gives people the freedom to control their technology while sharing knowledge and encouraging commerce through the open exchange of designs”[4].

Arduino is an example of open source hardware. According to creators “Arduino is an open-source prototyping platform based on easy-to-use hardware and software” [5]. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the Arduino programming language (based on Wiring), and the Arduino Software (IDE), based on Processing. From an innovation perspective Arduino is a medium to transforms ideas in innovative products. It is possible to transform Arduino in a robot, alarm system, irrigation system and others [6].

Android is a mobile operating system developed by Google, based on the Linux kernel and designed primarily for touchscreen mobile devices such as smartphones and tablets. From Android 3.1 (Honeycomb) is introduced the Android Open Accessory that allows external USB hardware (Android USB accessories) to interact with Android-powered devices in accessory mode. Some previously-released Android-powered devices are capable of acting only as a USB device and cannot initiate connections with external USB devices. Android Open Accessory support overcomes this limitation, enabling to build accessories that can initiate connections and interact with an assortment of Android-powered devices [7]. Arduino boards uses this mode for interacting with android devices. This attribute allows to develop a product composed of Arduino and Android device. The decision of designer determines that functionality can be assigned either to Arduino or to Android device.

3 Development methodology

The datalogger was developed following a methodology that was designed for this purpose called Andari. It is a methodology to agile and flexible development that aims to guide the development of software to integrate both platforms, Arduino and Android. Andari proposes a process of constant, iterative and incremental work and is based on

the agile method Scrum. The result of each phase is a functional prototype product. The figure 1 presents the various stages including the methodology.

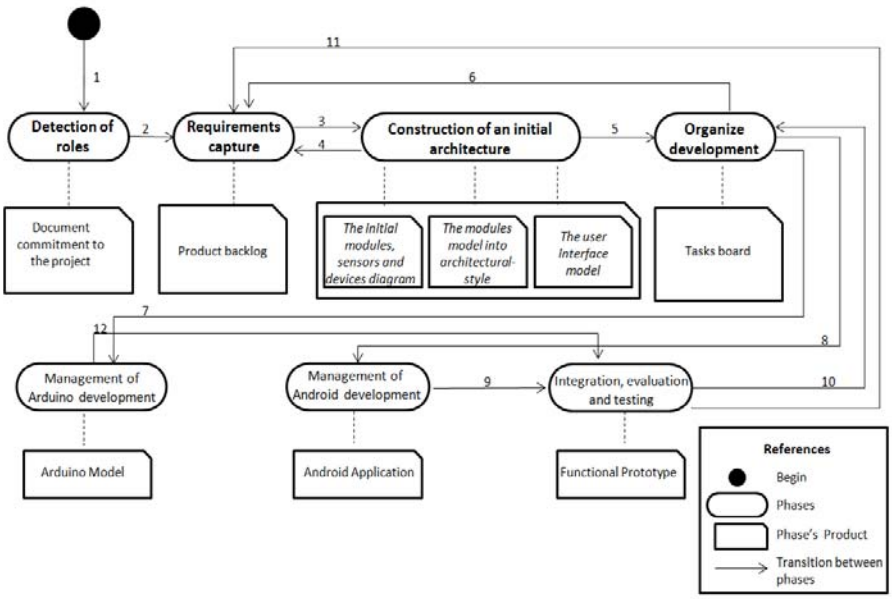


Figure 1. Andari methodology

3.1 Methodology phases.

1. **Detection of roles**: tasks and responsibilities of the participants are defined. Members of team must sign a document committing themselves to complete the project. Product: Document commitment to the project.
2. **Requirements capture:** it is defined as obtaining project requirements through different techniques and strategies. Product: List of requirements. Usually called product backlog.
3. **Construction of an initial architecture:** it is the description system through views using diagrams and models. It is a software architecture that serves only to functional requirements. The architecture is described by three components: the initial modules, sensors and devices diagram, the modules model into architectural-style and user interface model.

   ─*The initial modules, sensors and devices diagram* allow to describe the first overview of the major components of software and hardware and how they interact.

   ─*The modules model into architectural-style* allow to determine the more suitable architectural style as a base architecture for the product backlog and bring the initial modules, sensors and devices diagram to an arquitectural-style.

—*The user interface model* is a model that contains all user interfaces according to the initial view that the developer has about system. It is composed of hardware interfaces (Arduino interfaces) and Android interfaces.

Product: initial architecture.

4. **Organize development**: this process consists of organizing the development. Time and people are assigned to tasks. This organization is based on Sprint. The goal of each sprint is to develop certain requirements and consists of a list of tasks that compose the tasks board. The development in Arduino and Android is managed using the task board. Product: tasks board.
5. **Management of Arduino development**: in this phase Arduino development is managed. Product: Arduino Model
6. **Management of Android development**: in this phase in Android development is managed. Product: Android Application.
7. **Integration, evaluation and testing**: It is the process that integrates development activities in order to obtain a prototype of product. For each sprint the prototype evolves to the final product. Product: Functional Prototype.

3.2 Transitions between different phases of methodology

1. The process starts, and the first task is determining roles.
2. Upon completion determining roles of participants, we continue with the capture of requirements.
3. When finished with the capture of requirements should build an initial architecture which will do the base for the construction of the project.
4. New requirements can appear in the phase construction of the modules of the initial architecture that must be captured and analyzed.
5. With the initial architecture, the work-team passes to organize how modules will be developed, how long it will take us to do it and who are involved.
6. In Organize development, like phase of Initial architecture, new requirements may also appear, this capture and analysis is important.
7. Development of functionality in Arduino platform.
8. Development features in Android platform.
9. The work-team proceed to the integration of features developed in Arduino with the features of Android, the product prototype is tested and verified.

4 Development of prototype.

The datalogger prototype is developed following the Andari methodology. Below, tasks of each phase are presented.

4.1 Detection roles

At this stage the team is formed, it will be formed by customer, developer, technical experts and scrum master. The client is expert biologists. The developer is a senior student career Computer Science, who will make their final thesis. Technical experts

are teachers with technical skills required for product development technically assist the student. The scrum master is professor director of student final work.

4.2 Requirements Capture

Once completed the steps of defining roles the team started with requirements analysis where a detailed final product description desired with all functional specifications required by the customer was obtained. These requirements are summarized in the product backlog, shown in table 1.

Table 1. Product Backlog

ID	Functional requirements
R1	The system should record audio
R2	The system must record temperature
R3	The system should record humidity
R4	The system should display messages start and stop audio recording
R5	The system should display messages start and stop recording temperature
R6	The system should display messages start and stop recording humidity

4.3 Construction of the initial architecture

The main modules to form the system are determined in this phase. The most appropriate architectural-style is selected and the model of the user interface is defined.

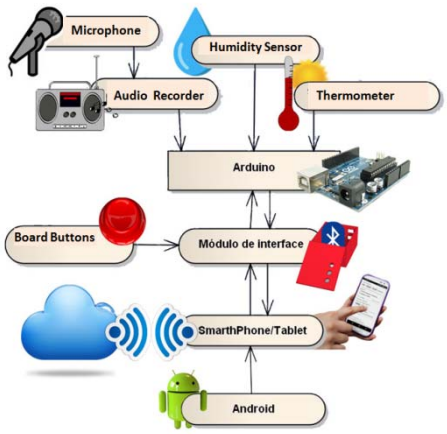


Figure 2. Initial modules, sensors and devices diagram of datalogger.

The initial modules, sensors and devices diagram.
The product backlog is the starting point to build the initial modules, sensors and devices diagram. Requirements indicate that the datalogger will have to ensure

functionalities to record environmental humidity and temperature variables, in addition to environmental audio recording. Another requirement is the communication using mobile devices. The figure 2 shows the diagram.

Description of the modules, sensors and devices

- Microphone: This device can transform sound waves into electrical energy and send all this information to the recorder module to be processed
- Module recorder: records sounds as ogg media files on a removable microSD card, attached to this module.
- Humidity sensor: performs the quantification of humidity in decimal numbers and sends this information to the Arduino.
- Temperature sensor: performs the quantification of temperature on decimal numbers and sends this information to the Arduino.
- Arduino is a device consisting of a plate with inputs and outputs that has a development environment that implements a language Processing / Wiring programming. Arduino receives data that were sent from the modules and sensors, determines the appropriate instruction and executes some action using the Arduino IDE tools.
- Pushbutton: a device that adds extra features to the standard Arduino interface.
- Interface Module: represents the functional connection between the Arduino and the smartphone and allows remote control of the smarphone with Android OS all the features of the prototype, uses SSP (Serial Port Profile).
- Android is comprised of a mobile operating system based on Linux, aimed for use on mobile devices with touch screen as smartphones, tablets, Google TV, and other device.
- Cloud is a device consisting of a platform for digital distribution of mobile applications for devices with Android Operating System. This platform allows users to browse and download applications. Through Cloud applications can be automatically updated if the user so requires, or can do it manually one by one.

The modules model into architectural-style

Once built the initial model of module system, it was determined that the most appropriate architectural style is the three-layer architecture. The figure 3 shows the datalogger architecture.

- *Physical Lager.* It is where the data resides and responsible for accessing them. Physical layer consists of a microphone, an audio converter, two sensors of humidity and temperature. These modules perform all storage data, receive the request for storage and retrieval of information from the processing layer.
- *Processing Layer.* It is where programs that run from the Arduino device reside. It is called processing layer because this is where all the rules are set to be met. This layer communicates with the user interface layer, to receive requests and deliver results. The processing layer communicates with the physical layer, to request the data manager to store or retrieve data from it.

– *User Interface Layer*. User Interface deals with present the system to the user, communicates information and capture user information in a minimum of process (performs pre-filtering to check for any formatting errors). The user interface resides on the android device. This layer communicates only with the layer processing.

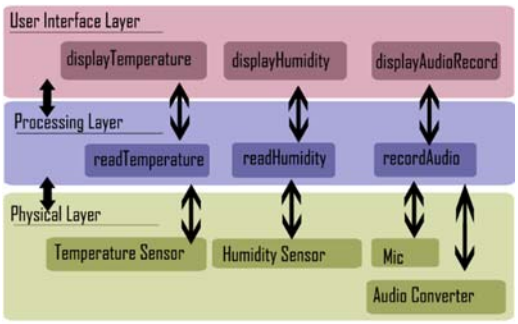


Figure 3. Datalogger architecture

User Interface Model.
From architecture it can be understood broadly general features. According to the proposal, the user interface will only be in the Android device, this was taken into account in designing an outline of the Android application.

4.4 Organize development

The project duration was estimated at 120 days with working day 5 hours daily. The first step in the project organization was grouped requirements of each sprint (see table 2). Subsequently the tasks board was generated (see table 3).

Table 2. Sprints

ID	From	To	Days	Name	Goal	Req.
S1	1-oct	1-nov	30	Record Sound	Reading and converting analog audio into digital audio in order to be stored into flash memory	R1
S2	1-nov	1-dic	30	Record temperature and humidity	Reading, displaying and storing of temperature and humidity	R2 y R3
S3	1-dic	1-en	30	Android interface design	Android interface design and message display	R4, R5 y R6
I4	1-en	1-feb	30	Communication	Integrate Android and Arduino using Bluetooth communication	R4, R5 y R6

4.5 Development Management in Arduino and Android

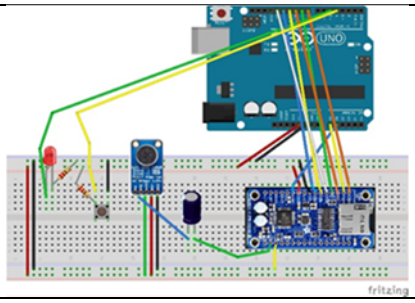
It is aimed to achieve in the each sprint an evolvable functional prototype as seen in tasks board (see table 3). The tasks for the development of Arduino are: record sound, record temperature, record humidity and communication. These tasks are present in 3 of the 4 sprint. The task of managing Android development was carried out in the sprint 3, and basically a layout object is used as a container for one or more views. The main development effort was in the programming and integration of Arduino with external devices. To this purpose, various experiments were conducted with identical functionality devices. For example, with the microphone were carried out three experiences with different types of microphones. This allowed to evaluate different devices with the same functionality to determine which device is best suited to the functionalities required by customers. Each experiment is described by the description, the diagram of connectivity, the program developed, results and conclusions. By way of summary, the development of experiment 3 included in the sprint 1 is presented in table 4.

Table 3. Tasks board

		Management of Arduino development				Management of Android development				Integration, evaluation and testing			
Sprint	Task	Subtasks	To do	In progress	Done	Subtasks	To do	In progress	Done	Subtasks	To do	In progress	Done
1	Record audio	Read audio		x									
		Analogic to digital	x										
		Save audio											
2	Record temperature	Read Temperature	x										
		Display Temperature	x										
		Save Temperature											
2	Record humidity	Read Humidity	x										
		Display Humidity	x										
		Save Humidity											
3	Android Interface Desing					Screen	x						
						Buttons	x						
4	Communication	Bluetooth connection	x			Bluetooth connection	x			Bluetooth connection	x		

Table 4. Sprint1 – Sound Record

Objetive	The device must capture sound waves. These waves must be transformed into data, that will then be stored in some type of mass storage.
Experiment 3	
Description	VS1053 Adafruit device was used. This contains the decoder chip VS1053 and audio encoder [8]. In addition to converting analog to digital signals, also it allows you to save data in a micro-sd.

Diagram of connectivity	
Program Developed	Sound record routine.
Results	The sounds were recorded and reproduced with acceptable fidelity
Conclusions	After performing various tests, and test several alternatives could be achieve the objective of sprint with the MAX9814 microphone [9], and the V1053 chip which satisfied the expectations of expert biologists.

4.6 Integration, evaluation and testing

Integration is the process that allowed in each sprint obtain a functional prototype. This was tested in order to assess the features that were added in each sprint. The final prototype is showed the figure 4.

Figure 4. Final prototype

5 Conclusions

The main objective of this work was to develop a datalogger to support the research activities carried out by biologist researchers of San Juan National University. The developed device has been specified by biologist researchers. It aims to meet current needs, capture sound, temperature and humidity. In addition, the device has been designed taking into account the reuse and / or adaptation in order to incorporate or adapt functionalities needed in future research. Another problem that motivated this work was the cost. In this regard, this work makes an important contribution because it allows the cost of acquiring a device this composed of only the costs of building the hardware. The cost of software is null or nonexistent, because the software is the result of this work and is available to be used in future devices.

References

[1] S. Oviedo, D. Diaz, A. Otazu, and F. Ibañez, "Software y Hardware de Código Abierto para Desarrollo de Habilidades de Innovación Tecnológica," *LACLO 2014- IX Conferencia Latinoamericana de Objetos y Tecnologías de Aprendizaje. Colombia. Manizales.,* 2014.

[2] J. M. Osier-Mixon. (2010) Open hardware: How and why it works. *IBM developerWorks*

[3] M. Hinkle, "Open Source: A Platform for Innovation," *http://insights.wired.com/profiles/blogs/open-source-a-platform-for-innovation#axzz2oOFSuMAt,* 2013.

[4] oshwa, "Open Source Hardware Association," *http://www.oshwa.org/,* 2016.

[5] Arduino, "Web Page," *http://www.arduino.cc/es/,* 2016.

[6] D. Díaz, S. Oviedo, and L. Muñoz, "Software e innovación: desarrollando productos con hardware y software flexible. ," *CACIC 2013 V Workshop Innovación en Sistemas de Software (WISS),* 2013.

[7] AOA, "Android Open Accessory (AOA)," *https://source.android.com/devices/accessories/protocol.html,* 2016.

[8] V. ADAFRUIT, "Recording with the VS1053," *https://learn.adafruit.com/adafruit-vs1053-mp3-aac-ogg-midi-wav-play-and-record-codec-tutorial/ogg-recorder,* 2016.

[9] A. MAX9814, "MAX9814," *https://learn.adafruit.com/adafruit-agc-electret-microphone-amplifier-max9814/overview?view=all,* 2016.