



*Universidad Nacional de La Matanza*

**DEPARTAMENTO DE INGENIERÍA E INVESTIGACIONES  
TECNOLÓGICAS**

**SISTEMAS OPERATIVOS AVANZADOS**

**SISTEMAS EMBEBIDOS Y ANDROID**

**“CRIARD”**

**Segundo Cuatrimestre - AÑO 2018**

Día: Miércoles

Turno: Noche Aula: 266

Docentes:

Lic. Graciela de Luca

Ing. Gerardo García

Ing. Waldo Valiente

Ing. Esteban Carnuccio

Ing. Mariano Volker

Raúl Villca

Integrantes:

Adagio, Matías

DNI: 34.424.221

Alvarez, Erik

DNI: 37.989.053

Caballero, Beatriz

DNI: 34.249.917

Couto, Valeria

DNI: 37.620.687

Díaz, Santiago

DNI: 36.872.427

## OBJETIVO

- Desarrollar un sistema electrónico para que los padres puedan monitorear el descanso del bebé a través de cualquier dispositivo con Android en una “Cuna Smart” que le avisará, a través de notificaciones, si el niño está llorando o si está mojado
- Además, la cuna contará con mecanismos automáticos que se accionarán ante diferentes estímulos, como, por ejemplo, hamacar al niño si está llorando, ajustar las condiciones lumínicas para favorecer el sueño y reproducir música para calmar al bebé.

## MATERIALES UTILIZADOS

### Software Utilizado

- IDE Android Studio: Desarrollo aplicación Android
- IDE Arduino 1.8.5: Desarrollo en la placa Arduino Uno

### Hardware utilizado

- Placa Arduino UNO Mega
- Protoboard
- Fuente de energía (12v)
- Cables (x40)
- Módulo de conexión Bluetooth (HC-05)

### Sensores

- Micrófono (KY-038)
- Sensor de humedad en tierra (FC-28)
- Sensor de humedad y Temperatura (DHT11)
- Sensor de luz (LDR 5528)

### Actuadores

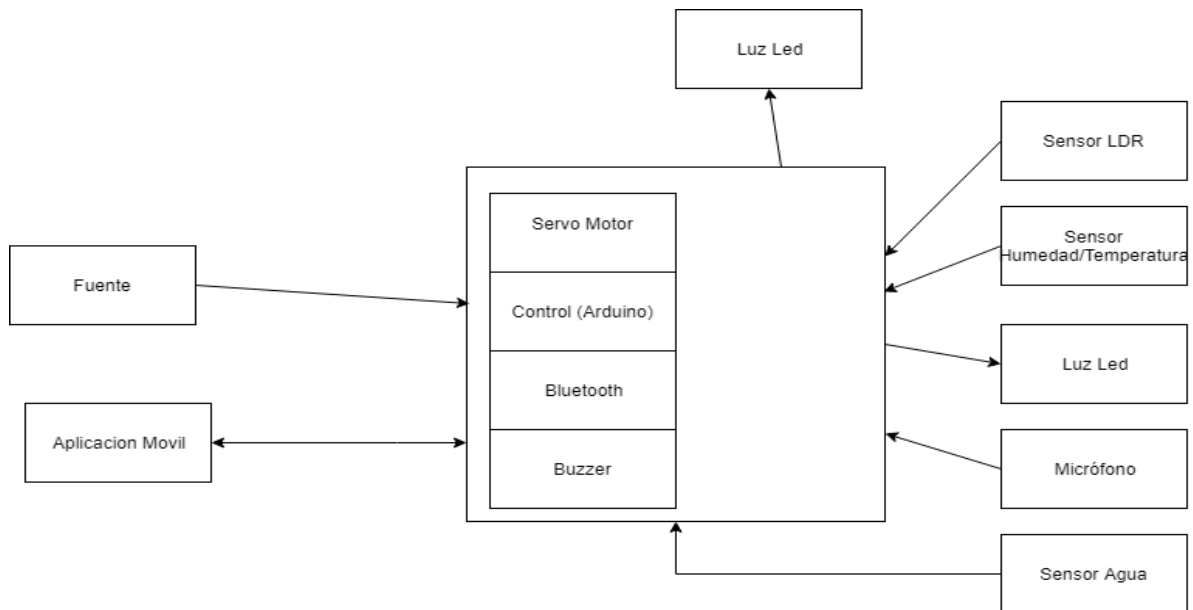
- Buzzer pasivo (KY-006)
- Luz LED (x2)
- Servomotor (SG90)

### Elementos varios

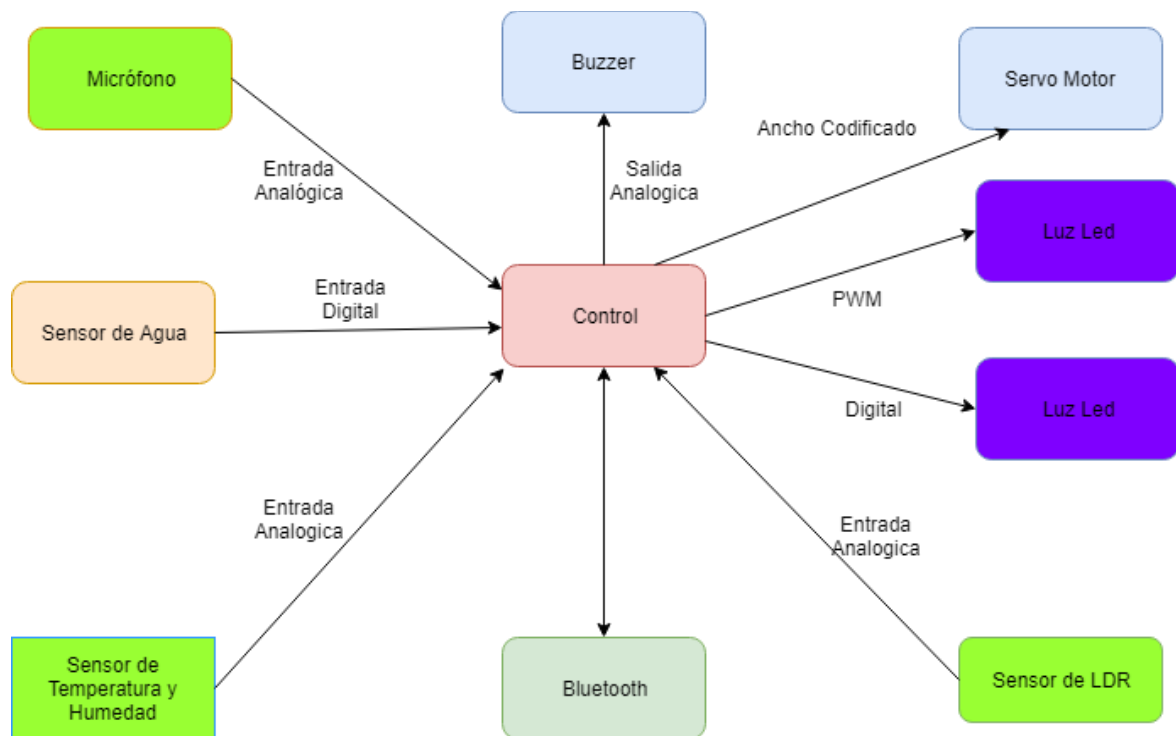
- Estructura cuna
- Colchón

## DIAGRAMAS

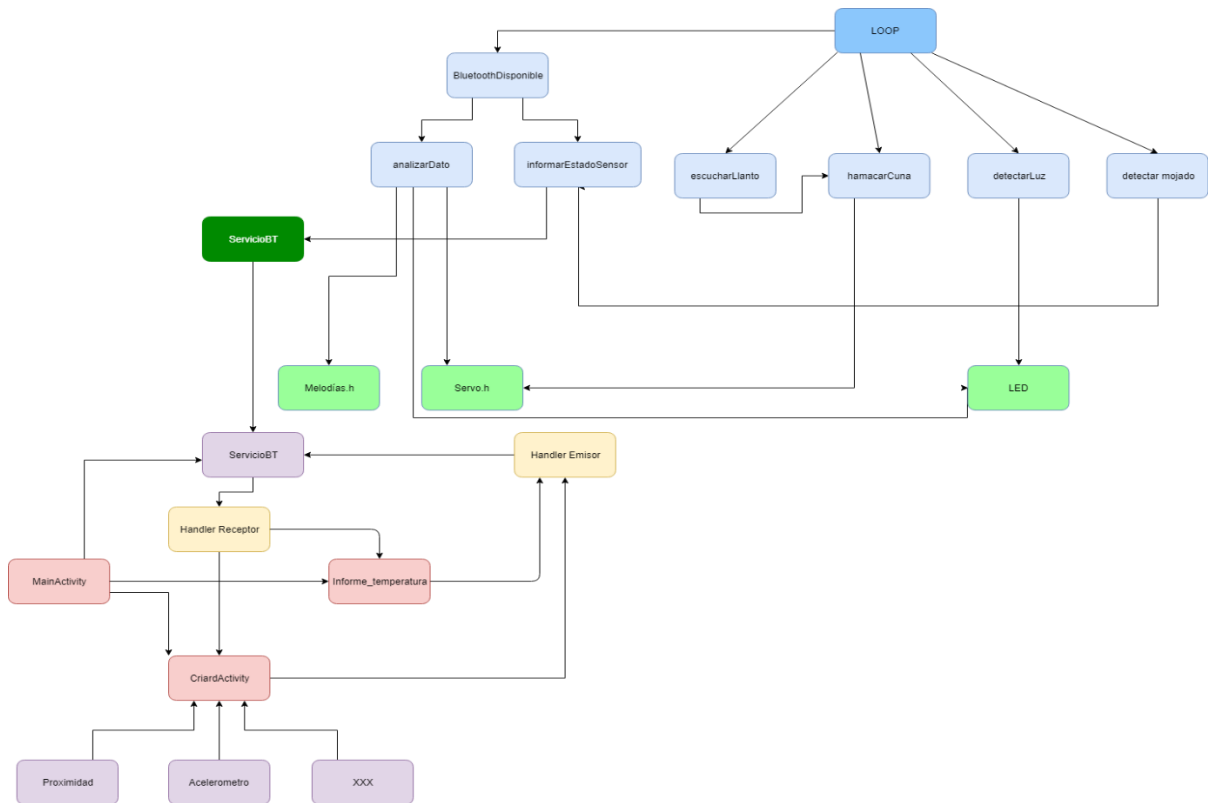
### Físico



### Funcional



## Software



## IMPLEMENTACIÓN

### Sistema Embebido

#### 1. Implementación módulo Bluetooth HC-05

Se utiliza la librería *SoftwareSerial.h* para implementar la comunicación del módulo Bluetooth conectado al Arduino con la aplicación Android. Esto permitirá controlar y visualizar información proveniente de la Cuna Inteligente en el sistema Android.

Se utilizan las siguientes funciones:

- *SoftwareSerial*: crea una nueva instancia del objeto *SoftwareSerial*.
- *Available*: Obtiene el número de bytes (caracteres) disponibles para la lectura de un puerto serie software.
- *Begin*: configura la velocidad de transferencia de datos entre el Bluetooth HC05 y el de Android.
- *Read*: Devuelve un carácter que se recibió el pin RX del puerto serie de software.
- *Print*: Imprime datos al pin de transmisión del puerto serie software.
- *Flush*: Espera que complete la transmisión de datos salientes del bluetooth.

- *Write*: Envía los comandos al dispositivo Android (distintos estados del sistema)

## **2. Implementación Micrófono KY-038**

Para implementar este sensor, usamos la función *analogRead*, donde solamente se le debe indicar el PIN al que está conectado en el Arduino y esta función nos devuelve la lectura que se hace del ruido que hay en el ambiente. Para poder detectar cambios bruscos de ruido, se tienen en cuenta el ruido promedio que hay en la habitación dónde se encuentra el Sistema Embebido. Para mantener este valor actualizado, cada un ciclo de mil muestras se recalcula lo que permite alta adaptabilidad a diversos ambientes

## **3. Implementación Sensor de Humedad en Tierra (FC-28)**

El mismo está constantemente midiendo. Para ello utilizamos la función *digitalRead*. En caso de detectar agua se arma el mensaje para comunicar que el bebé mojó la cama.

## **4. Implementación Sensor de Humedad y Temperatura DHT11**

Para este sensor, se utilizó la biblioteca “DHT.h”. El mismo realiza las lecturas cuando se notifica por bluetooth a la aplicación Android y le envía los datos de humedad relativa y temperatura ambiental en grados Celsius

## **5. Implementación Sensor de luz LDR 5528**

Este sensor se utiliza usando la función de Arduino *analogRead*, como es un valor que va del 0 al 1023, se hacen conversiones matemáticas para saber cuan iluminado está una habitación utilizando una taza porcentual de luz, dónde si la iluminación es menor al 50% se empieza a encender una luz de forma gradual e inversamente proporcional a la oscuridad medida

## **6. Implementación Buzzer pasivo KY-006**

El buzzer se activará a través de la aplicación Android haciendo que suene una melodía para calmar el llanto del bebé. Para esto fue necesario utilizar la librería “*melodía.h*” que contiene las melodías y nos permite inicializar y detener la música en el momento deseado.

## **7. Implementación LED**

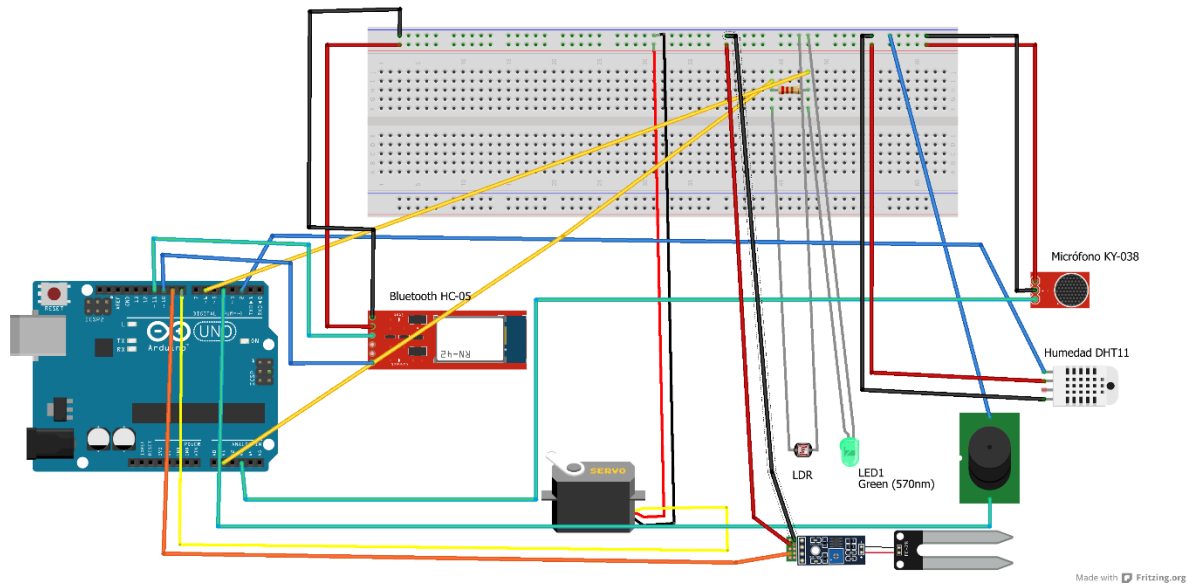
Para implementar este sensor, usamos la función *digitalWrite*, donde solamente se le debe indicar el PIN al que está conectado en el Arduino, y los valores HIGH (encendido) o LOW (apagado). Se le debe proveer una resistencia (para controlar la tensión que se envía del Arduino al LED, y este último no se queme).

## 8. Implementación ServoMotor SG90

El servomotor es el encargado de mecer la cuna al activar la opción correspondiente en el sistema Android. Para esto se implementa la librería *Servo.h* que permite establecer el pin al cual conectamos el servo (pin 8) e indicar entre que ángulos queremos girar el motor.

Se utilizan las siguientes funciones:

- *Attach*: permite indicar en que pin tenemos conectado nuestro servo.
- *Write*: donde se indica en qué ángulo queremos posicionar nuestro servomotor (en nuestro caso oscila entre 75° y 105°).

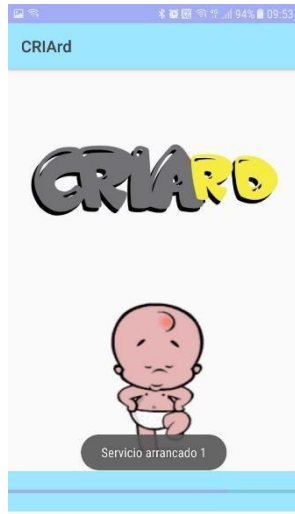


## Aplicación Android

### 1. Activities

El sistema consta de las siguientes Activities:

- *MainActivity*:



Esta activity es la que se acciona al entrar a la aplicación. En el método onCreate inicializa el servicio que hará la conexión Bluetooth con el Arduino y quedará activo en segundo plano para poder notificar distintos eventos que sucedan en el sistema embebido.

- *CRIArDMainActivity:*



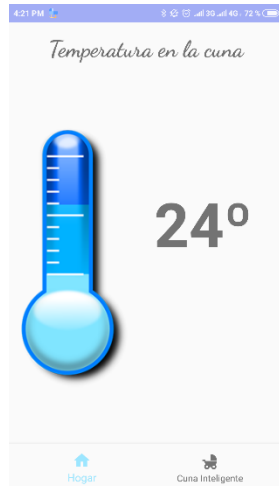
Es el hilo principal de la aplicación que administra el envío de mensajes con la placa Arduino.

Cuenta con dos view toggle: Uno llamado “Música” que permite encender o apagar la música, y otro “Cuna” que comienza a mecer la cuna o detiene su movimiento.

También hay dos MenuItem que disparan las siguientes activities:

- 1) Cuna Inteligente: En caso de estar en otra activity, vuelve a esta.
- 2) Hogar: Acciona la activity “Informe temperatura”

- *Informe Temperatura:*



En esta activity se muestra los valores de temperatura actualizados de la habitación donde se encuentra la cuna. En caso de que se cierre la conexión anteriormente establecida, la aplicación forzará el cierre.

## 2. Proceso en Background:

- *ServicioBT:*  
Se encarga de gestionar la conexión Bluetooth entre el dispositivo Android y el sistema embebido realizando las siguientes acciones:
  - 1) Recibe e interpreta las peticiones del Arduino
  - 2) Envía y arma los mensajes que el Arduino deberá interpretar
  - 3) Genera notificaciones para informar distintos eventos que ocurren en la cuna
- ActualizarCuna (AsyncTask): Permite modificar la interfaz de CRIARdMainActivity tomando los datos que nos provee la clase Handler.
- ActualizarTemperatura (AsyncTask): Permite modificar la interfaz de Informe\_temperatura tomando los datos que nos provee la clase Handler, actualizando sus valores.
- ConnectedThread: Hilo que corre sobre el servicio y establece la conexión con el módulo Bluetooth

## 3. Sensores

- Sensor de proximidad: Enciende luz LED en la cuna
- Sensor de luz: Comienza a reproducir melodías
- Acelerómetro: Al accionarlo, comienza a hamacarse la cuna.

## MEJORAS FUTURAS

Una vez completado el desarrollo inicial del proyecto se propone expandir el sistema con las siguientes funcionalidades:



- Detección de síntomas de Epilepsia y apnea
- Reconocimiento de distintos patrones de llanto
- Mejorar la conectividad implementando Wi-Fi para soportar múltiples conexiones
- Poder reproducir música y realizar otras acciones al mismo tiempo
- Poder detectar si ocurren cambios bruscos de temperatura o humedad ambiental dentro de la habitación.
- Mejorar la calidad del micrófono utilizado para detectar el llanto del bebe.