

# BLINK

---

## TÍTULO DE LA PROPUESTA

BLINK la luz que te acompaña

## AUTORES

Eva María Amate Gallardo (España)

Andrea López Duro (Colombia)

Oscar Salavarieta (Colombia)

## CONTEXTO DEL PRODUCTO

Maestría en Diseño

Universidad de Los Andes (Colombia)



# BLINK

---

## MEMORIA DESCRIPTIVA

BLINK es un dispositivo móvil, portátil y personalizable para colgar como complemento o apoyar sobre una superficie. Se trata de un objeto que no sólo te acompaña e ilumina, sino que además se adapta a tu estilo o a tu estado anímico cambiando su color.

BLINK se conecta con tus amigos y envía "blinks" (señales lumínicas de un color específico) a quién tú quieras con solo pulsar el botón de tu App.

A través de su interfaz, se controla tanto el color como la intensidad del brillo de la luz



## BRIEF - ENCARGO DE DISEÑO

### **Concrete City, un acercamiento ultraligero a las intervenciones urbanas**

Propuesto por Rafael Hernando Barragán Romero, *Chief Design Architect* de Arduino y diseñador especializado en dispositivos lumínicos interactivos incorporados a espacios urbanos.

Decano de la Facultad de Arquitectura y Diseño de la Universidad de los Andes (Colombia)

En las últimas décadas, la intensidad de las fuerzas de transformación que interrumpen la "condición urbana" ha puesto en tela de juicio el control y la estabilidad sobre las ambiciones en las que tradicionalmente se han basado. Este estado de cosas repercute en las maneras de imaginar, teorizar y poner en práctica el urbanismo y el proyecto territorial. Si las relaciones con los flujos pueden ser operadas por la acción de grandes infraestructuras arquitectónicas, el despliegue de una constelación de micro-elementos constituiría otra forma de pensar e implementar la idea de infraestructura como un equipo que es atomizado, en red, abierto y maleable. Un enfoque ultraligero del urbanismo refracta el papel y la ambición de micro-intervención en el proyecto urbano como estrategia de vector territorial de acción. Lo que habitualmente era menospreciado o simplemente no visto por la arquitectura con-

vencional y los enfoques urbanísticos -lo muy pequeño, el movimiento, el informal: el "plankton" - recupera así un nuevo valor informativo y operativo para imaginar la intervención. "Plankton" no como un conjunto de entidades indistintas, apáticas y organismos pasivos, sino más bien como una multitud virtualmente de pequeños dispositivos para facilitar variadas actuaciones y usos, una mirada en movimiento de colonización creativa de los potenciales espaciales y temporales del paisaje urbano. Se conceptualizará, construirá e implementará una intervención ultraligera, inalámbrica a gran escala de elementos insertados en un espacio, transformando una sección de la ciudad en una nueva experiencia de ambiente. El trabajo se llevará a cabo mediante un enfoque práctico que, desplegando una red de computación omnipresente inalámbrica efímera, con actuadores de injerto temporal,

con pequeños dispositivos "plankton", que habitan objetos e infraestructuras urbanas. Estos elementos urbanos, desviados de su uso primario, crearán una orquestación furtiva del ambiente. El espacio, las formas, incluyen elementos de las cosas materiales del sitio, actividades sociales, fenómenos y los procesos que se producen concomitantemente, específicos de un tiempo, lugar y cultura. La intervención colectiva se desarrollará específicamente para ser "incorporada" en el espacio público, una especie de plancton electrónico de infraestructuras. Estos elementos interactuarán a través de nodos y relaciones, controlados remotamente en tiempo real en un determinado espacio público, así como activados interactivamente mediante el uso de sensores incorporados al contexto.

## FUNCIONAMIENTO

Blink es una lámpara completamente portátil, compuesta de un circuito integrado que controla un led RGB a través de la app-interfaz en cualquier dispositivo inteligente. Cada lamparita Blink se comunica con sus "hermanas" a través de la red. Basta con abrir la app-interfaz y seleccionar el ícono del destinatario. El parpadeo durará unos segundos y se ejecutará en un color diferente al que el usuario tenga establecido. Por ejemplo, los "blinks" de Juan se realizan en color verde. María, tiene su lámpara de color amarillo y comienza a parpadear en color verde. María sabe que Juan ha enviado esa señal. Dispone de una batería y una base que permite cargarla o utilizar Blink como lámpara de mesa.

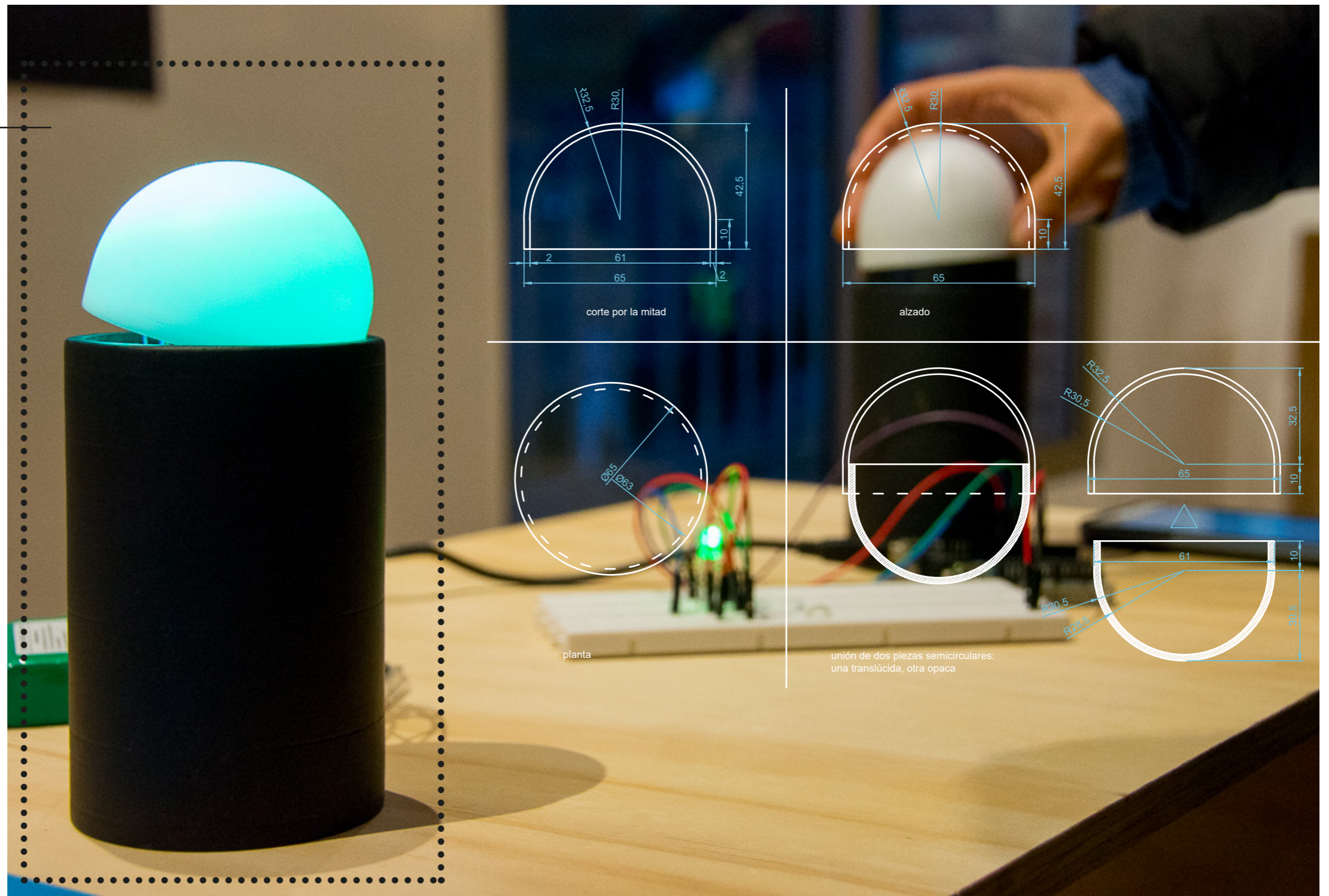
El objetivo de Blink es aunar en un mismo producto diseño, personalización, interacción y diversión capaz de enviar mensajes luminicos a cualquier parte del mundo.

La versión experimental de Blink cuenta con tres lámparas, una por cada miembro del equipo de diseño. Para este prototipo se utilizaron tres tarjetas Arduino One Wifi, una por cada lámpara. Tanto la tarjeta Arduino como su batería de litio son externas, es decir, funcionan provisionalmente como una prolongación de la lámpara. Para la versión de sobremesa, la base oculta en su interior la tarjeta y la batería. Para que las lámparas puedan comunicarse entre sí y enviar mensajes luminicos deben estar dentro de la misma red WiFi, así como los dispositivos inteligentes desde los que se controlan.

El resultado es un diseño sencillo, funcional y equilibrado estéticamente que si bien no permitía enlazar las lámparas a través de internet, sí que era capaz de conectar lámparas y smartphones dentro de una misma red Wifi para poner a prueba todas y cada una de sus funcionalidades.



Propuesta objeto para la lámpara en posición fija



## DESARROLLO DE LA PROPUESTA

El circuito de pruebas se realizó con una tarjeta "Arduino One Wifi". Arduino es una plataforma electrónica de código abierto basada en hardware y software fácil de usar. Esto permitió tangibilizar el primer prototipo de Blink con rapidez y, junto a la funcionalidad Wifi, permitió conectar varios dispositivos de manera inalámbrica.

La interfaz se diseñó y desarrolló en *Processing*, un software flexible para crear contextos visuales y artísticos interactivos a través de código de programación. Para poder controlar las lámparas desde los teléfonos, se instalaron apps capaces de ejecutar el código que se encuentran disponibles de forma gratuita en la Tienda de Google.

Por último, la parte física de prototipo se creó a partir de la combinación de piezas impresas en 3D, piezas de madera cortadas con láser y acrílico translúcido termoformado.

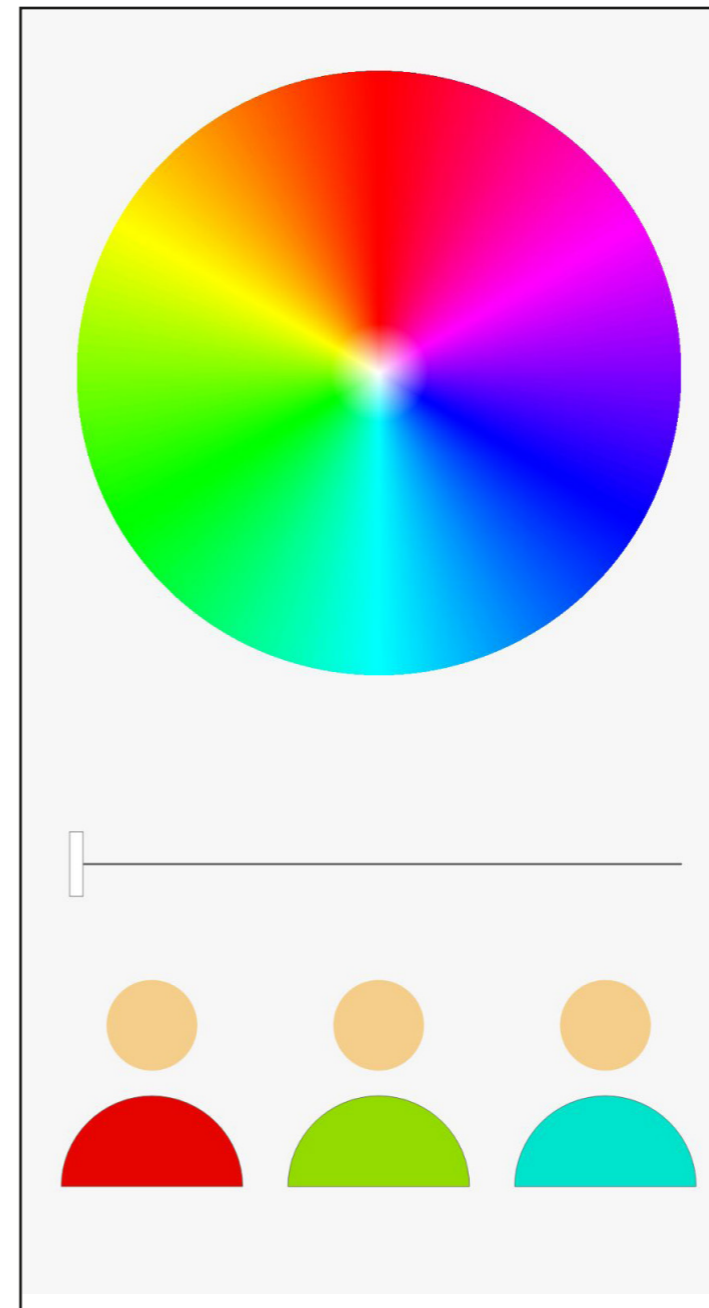
```

sketch_interfaz2_processing_client | Processing 3.5.4
Archivo Editar Sketch Depuración Herramientas Ayuda

sketch_interfaz2_processing_client
1 //librerías
2
3 import processing.net.*; //librería para llamar URL
4 Client c; //cliente
5 String data; //cadena de caracteres
6
7 //atributos LED1
8 int Red=0;
9 int Green=0;
10 int Blue=0;
11 color co=255;
12 PImage m;
13
14 //atributos LED2
15 int rojo = 0; //datos enviados al puerto serie
16 int verde = 0; //Corresponden a los niveles de brillo deseados
17 int azul = 0;
18
19 //slider
20 float slider = width/2;
21
22 void setup() {
23   size(600, 1066);
24   smooth();
25   ellipseMode(CENTER);
26   background(245);
27
28   //Dibujando la rueda de color
29
30   colorMode(HSB, 100);
31   for (int x=0; x<width; x++) {
32     for (int y=0; y<height; y++) {
33       if (dist(width/2, height/3.5, x, y)<=width/2.4) {
34         stroke(map(atan2(x-width/2, y-height/3.5), -PI, PI, 0, 100), map(dist(width/2, height/3.5, x, y), 0, width/2.4, 0, 100));
35       }
36     }
37   }
38   noFill();
39   // stroke(50);
40   // ellipse(width/2, height/3.5, (5*width)/6, height/2.132);
41   m=get();
42 }
43 void draw() {
44   //Si presiono el mouse, aparece una circunferencia negra para tomar el color
45
46   image(m, 0, 0);
47   if ((mouseX>=width/12)&&
48       (mouseX<=(11*width)/12)&&
49       (mouseY>=height/20)&&
50       (mouseY<=height/1.9)&&
51       (mousePressed==true)) {
52     fill(get(mouseX, mouseY));
53     stroke(50);
54     ellipse(mouseX, mouseY, 50, 50);
55     co = get(mouseX, mouseY); //obtener valor del color
56   }
57 }
58
59
60

```

Código de la interfaz gráfica en *Processing*



Prototipo interfaz gráfica para móviles

```

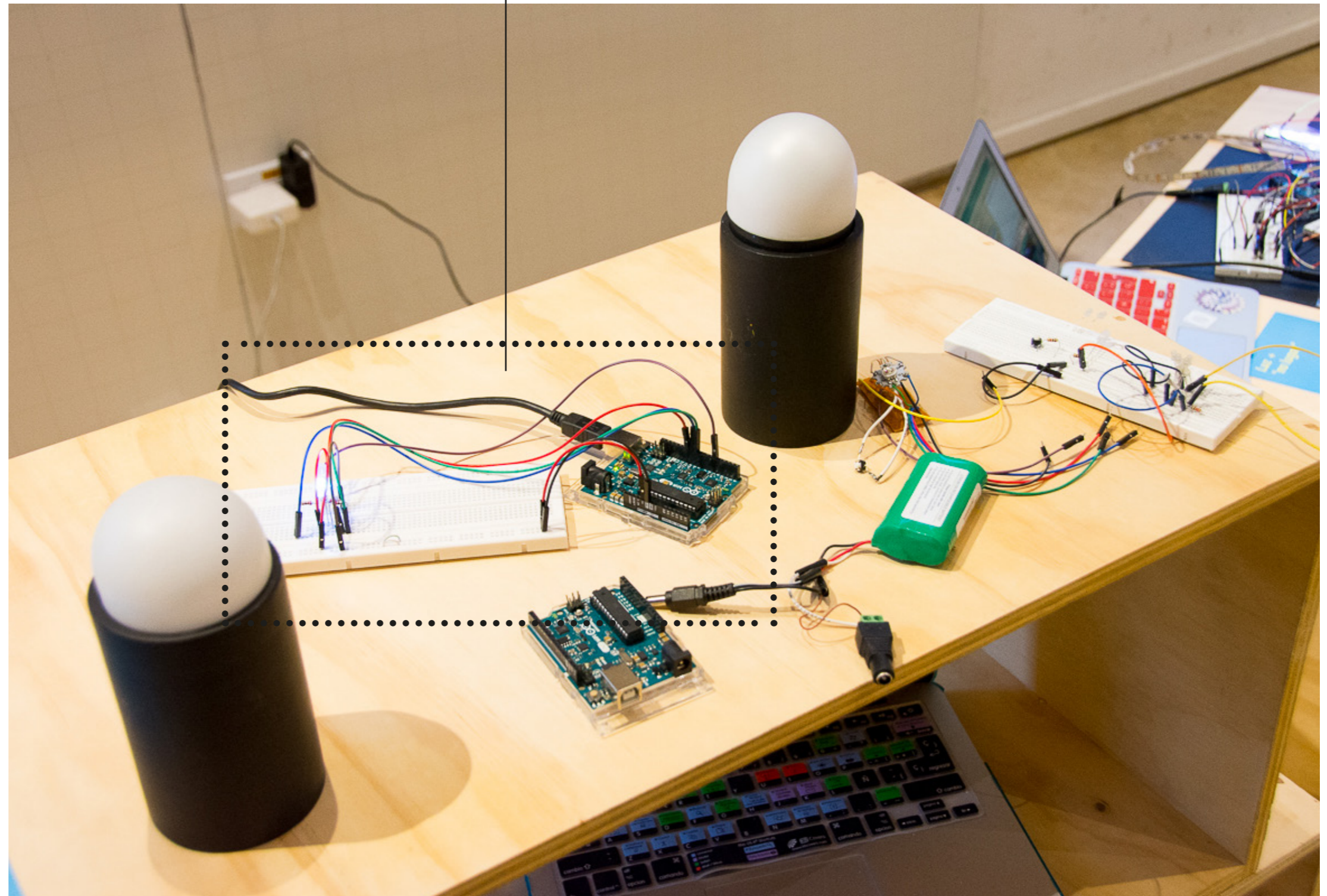
Prototipo_final-22_BRILLO-SI Arduino 1.8.13
Archivo Editar Programa Herramientas Ayuda

Prototipo_final-22_BRILLO-SI
1
2 Con este código se puede controlar la elección de 6 colores
3 de un LED RGB a través de presionar un botón.
4
5 Conceptos involucrados:
6 - PWM y escritura analógica
7 - Digital Input
8 - Creación de Funciones
9
10
11 //Fin usado como entrada para reconocer el botón:
12 int pinButton = 6;
13 //Variable para guardar el estado del botón:
14 int buttonValue;
15 bool controlSwitch = false;
16
17 //Variables RGB:
18 int R=1, G=9, B=10, W;
19 int valueR, valueG, valueB;
20 int pinR, pinG, pinB;
21
22 //Para conectar el servidor
23 #include <Wire.h>
24 #include <UnoWiFiDevEd.h>
25
26 //EEPROM
27 #include <EEPROM.h>
28 int addr1 = 0;
29 int addr2 = 1;
30 int addr3 = 2;
31 int addr4 = 3;
32
33 //Button Debounce
34 int buttonState; // the current reading from the input pin
35 int lastButtonState = LOW; // the previous reading from the input pin
36 unsigned long lastDebounceTime = 0; // the last time the output pin was toggled
37 unsigned long debounceDelay = 50; // the debounce time; increase if the output flickers
38
39 //Variables brillo
40 int azul, verde, rojo; //Variable que recibirá el nivel de azul
41
42 void setup() {
43   Serial.begin(9600);
44
45   pinMode(R, OUTPUT);
46   pinMode(G, OUTPUT);
47   pinMode(B, OUTPUT);
48   pinMode(5, OUTPUT);
49   pinMode(pinButton, INPUT);
50   WiFi.begin();
51   WiFi.println("REST Server is up");
52
53   valueR=EEPROM.read(addr1);
54   valueG= EEPROM.read(addr2);
55   valueB=EEPROM.read(addr3);
56   azul=EEPROM.read(addr4);
57
58   Serial.println("Valores iniciales");
59   Serial.println(valueR);
60   Serial.println(valueG);
61   Serial.println(valueB);
62 }
63
64 void loop() {
65
66   //Switch de on/off
67   buttonValue = digitalRead(pinButton);
68   //If (buttonValue == HIGH){
69     //controlSwitch = !controlSwitch;
70   // }
71   if (buttonValue != lastButtonState) {
72     // reset the debouncing timer
73     lastDebounceTime = millis();
74   }
75 }

```

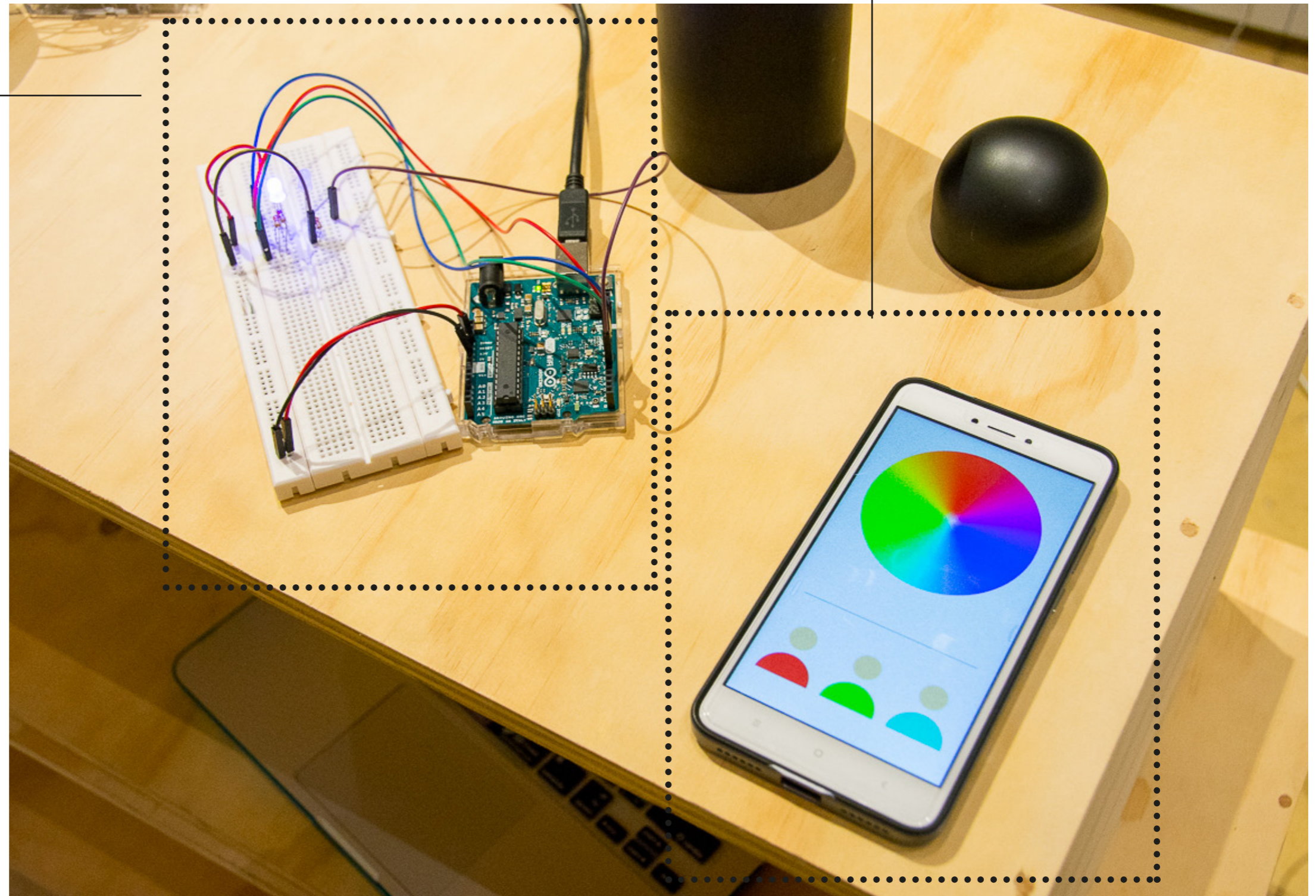
Captura del código de funcionamiento de la lámpara en Arduino

Prototipo de exploración: Arduino conectada a circuito sobre *board* con led RGB



Prototipo de exploración: Arduino conectada a circuito sobre *board* con led RGB

Interfaz gráfica desarrollada en *Processing*





# BLINK



# BLINK

