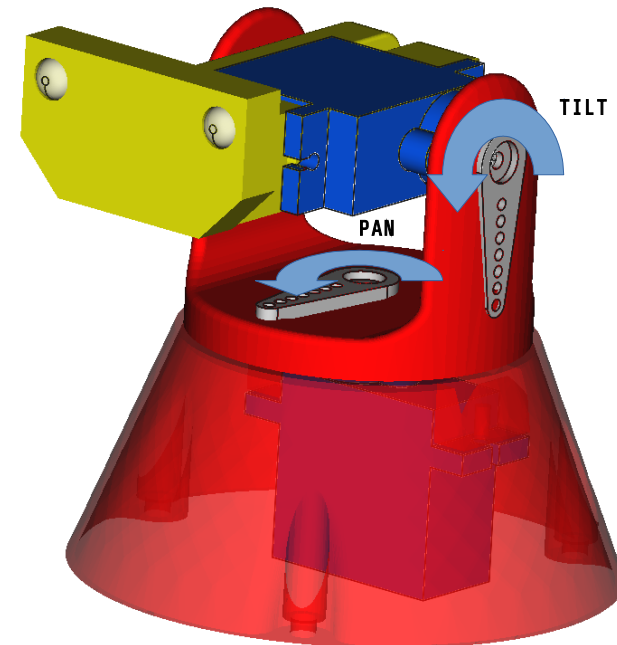


## #Reto 2. Brazoled

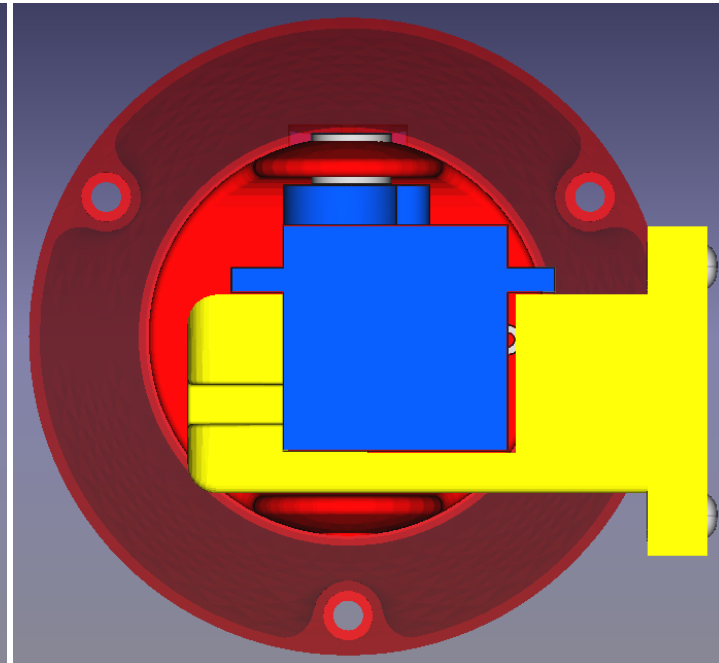
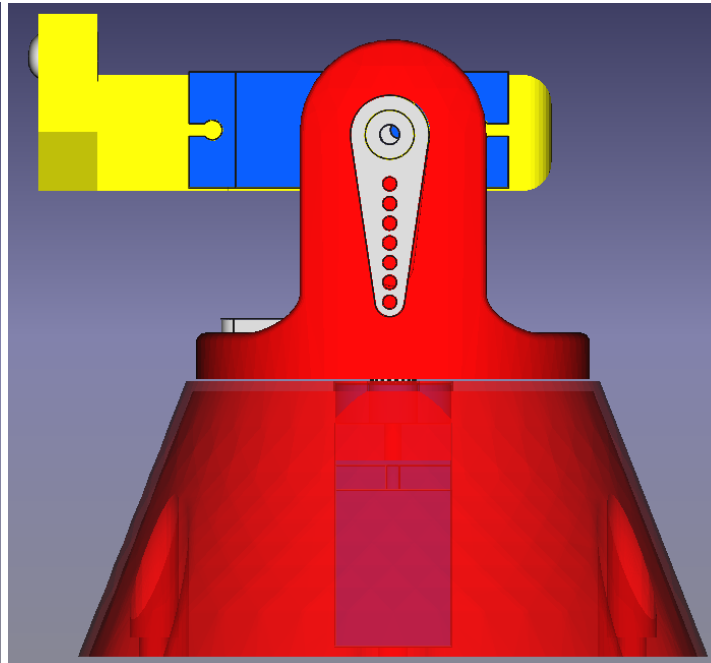
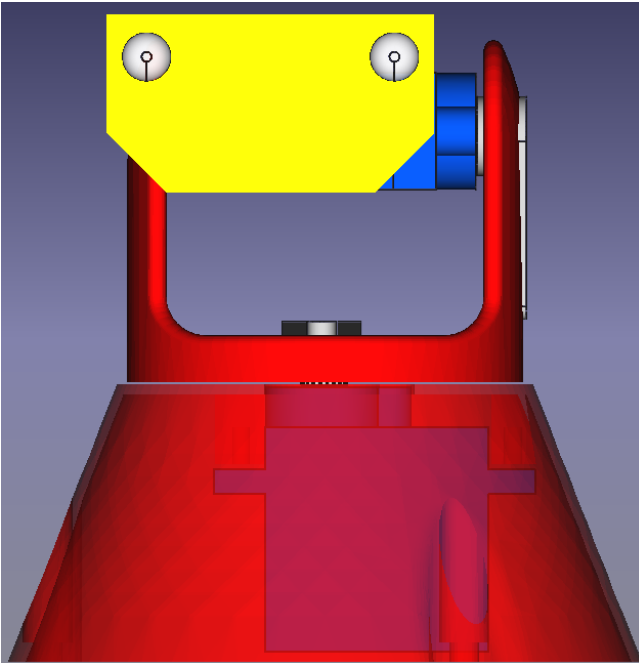
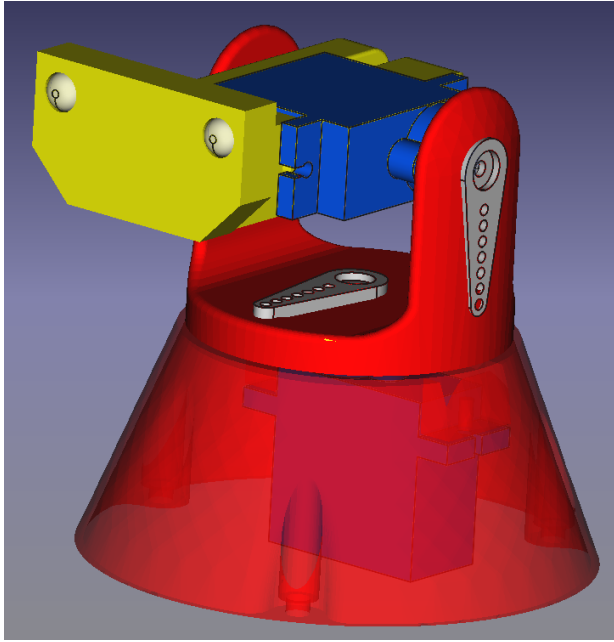
- Estamos desarrollando un sistema de vigilancia nocturno.
- Necesitamos luz para ver por la cámara de seguridad.
- Desarrollamos un brazo robótico que nos permitirá girar y ver con claridad

Hemos utilizado y modificado el diseño compartido por Manuel @eltitomanolo (<https://twitter.com/eltitomanolo>)



## #Reto 2. El diseño

- 2 servos SG90 + 2 adaptadores
- 2 LEDs
- 3 piezas impresas en 3D
- 1 Arduino
- Varios cables
- Tornillos



# #Reto 2. El código

```
/*
  Brazoled - RETO 2 - Taller Arduino Basics I
  Ripolab Hacklab Noviembre 2017

  by @akirasan
  ripolab.org
*/

#include <Servo.h>

// Definimos los dos Servos que vamos a utilizar:
// ...uno para el TILT --> Parte superior
// ...otro para el PAN --> Parte inferior

Servo servo_tilt;
Servo servo_pan;

// Definimos los pines que vamos a utilizar para controlar nuestros Servos
// Nos inventamos dos "definiciones" para hacer referencia los pines 2 y 3

#define PIN_SERVO_TILT 2
#define PIN_SERVO_PAN 3

#define PIN_LED1 9
#define PIN_LED2 10

void setup() {

  // Configuramos los pines para cada servo con el metodo "attach"
  servo_tilt.attach(PIN_SERVO_TILT); //servo superior
  servo_pan.attach(PIN_SERVO_PAN);   //servo inferior

  // Activamos la iluminación de los LEDs
  pinMode(PIN_LED1, OUTPUT);
  pinMode(PIN_LED2, OUTPUT);
  digitalWrite(PIN_LED1, 110);
  digitalWrite(PIN_LED2, 110);
}
```

```
// Función para controlar el angulo de giro del TILT
void mover_servo_tilt(int angulo) {
  servo_tilt.write(angulo);
}

// Función para controlar el angulo de giro del PAN
void mover_servo_pan(int angulo) {
  servo_pan.write(angulo);
}

void loop() {
  mover_servo_tilt(0);
  mover_servo_pan(0);
  delay(1000);
  mover_servo_tilt(45);
  mover_servo_pan(20);
  delay(1000);
}
```

## #Reto 2. BONUS EXTRA

- Modificar la función del TILT para controlar la velocidad de giro:
  - Utilizaremos un nuevo valor de entrada en la función de mover el servo: "`int` velocidad"
  - Moveremos en  $1^\circ$  grado cada vez
  - Debemos leer el ángulo actual de nuestro servo

```
int angulo_actual = servo_tilt.read();
```

