

Informe Preliminar de Diseño



Equipo Team Ramiro

Mentor: Manuel Blazquez Perez Merino

Participantes: Esteban Lorite Arauzo, David Tarín Alcaide, Iker
González-Casallo Castro, Gael Álvarez Monés,
Clara Jimeno Redondo, Marco Padillo

IES Ramiro de Maeztu, Madrid, Madrid



Índice

1. Introducción	2
1.1 Equipo, organización y roles	2
1.2 Objetivos del proyecto	3
2. Descripción del proyecto	4
2.1 Fases de la misión	4
2.2 Misión Primaria	4
2.3 Misión Secundaria	5
2.3.1 GPS	5
2.3.2 Telemetría avanzada	5
2.3.3 Registro visual del descenso	5
2.4 Diseño Mecánico	6
2.5 Diseño eléctrico	8
2.6 Paracaídas	8
2.7 Estación de tierra	10
2.8 Software	11
3. Planificación	15
3.1 Planificación del proyecto	15
3.2 Estimación del producto	15
3.2.1 Presupuesto	15
3.2.2 Financiación y apoyo externo	17
4. Programa de difusión	18
5. Requisitos para lanzamiento	18
6. Observaciones	19
7. Bibliografía	19



Resumen: Cansat es un programa dirigido hacia institutos, universidades y grados superiores. Esta competición está iniciada por la Agencia Espacial Europea en la que cada equipo debe diseñar, programar y construir un satélite del tamaño de una lata capaz de cumplir una misión determinada, recoger datos y efectuar retornos controlados.




Este documento recoge toda la información del proyecto del Team Ramiro para el concurso CanSat España 2023-2024. En él se describe el trabajo realizado para el desarrollo del proyecto. El documento sigue las bases especificadas por la organización del concurso. Estos datos pueden consultarse en su página web. <https://esero.es/cansat-2/>

1. Introducción

1.1 Equipo, organización y roles

El equipo Team Ramiro está formado por 6 estudiantes de primer curso del bachillerato tecnológico y artes en IES Ramiro de Maeztu con una gran inquietud por la adquisición de conocimiento, en Madrid.

	David	Marco	Gael	Clara	Iker	Esteban
Diseño mecánico	Alta implicación		Baja implicación			
Diseño electrónico						Alta implicación
Sistema de frenado	Baja implicación	Baja implicación		Baja implicación		
Divulgación y redes sociales			Media implicación			Baja implicación
Contabilidad				Baja implicación		
Planificación	Baja implicación	Media implicación				
Desarrollo del informe		Baja implicación				
Comunicaciones						Media implicación
Telemetría avanzada	Baja implicación			Media implicación		
Soldado						

Alta implicación 
Media implicación 
Baja implicación 



También forma parte del equipo como educador y mentor Manuel Pedro Blázquez Merino, profesor de TIC en el IES Ramiro de Maeztu.

1.2 Objetivos del proyecto

El CanSat llevará a cabo dos misiones: una misión primaria, impuesta por las bases del concurso y común a todos los equipos, y una misión secundaria, la cual elige cada equipo.

La misión primaria consistirá en la toma de una serie de datos de la atmósfera durante el descenso y la transmisión de estos por telemetría a la estación de tierra a al menos 1 transmisión por segundo.

La misión secundaria de nuestro equipo es la telemetría avanzada y el registro visual del descenso. Tras la liberación y durante el descenso el cansat medirá y transmite telemetría adicional a la exigida para la misión primaria. Estas son: Aceleración, posicionamiento GPS y el nivel de la calidad del aire. También por otra parte durante el descenso se realizará una grabación del trayecto del satélite desde el propio satélite.

Para mantener una velocidad constante y un descenso con una velocidad controlada se empleará un paracaídas.



2. Descripción del proyecto

2.1 Fases de la misión

Hemos dividido la misión en 4 fases, la fase 0, de la cual carecemos control y en la que no podemos realizar nada, que sería la fase de lanzamiento y alcance de altitud máxima, la cual se aproxima a los 1000 metros. Después le siguen las siguientes 3 fases las cuales son:

Fase 1: En esta fase, después de ser liberado, el cansat desplegará su paracaídas gracias a la implementación del acelerómetro y el gps, los cuales determinarán si el cansat ha empezado a caer para la liberación del paracaídas.

Fase 2: En esta fase se hará toda la toma de datos de tanto la misión principal como la misión secundaria. Durante esta fase el CanSat se comunicará a la base de tierra mediante radio transmisión.

Fase 3: En esta fase el CanSat aterrizará y gracias a la integración de la unidad de GPS se localizará su ubicación para el rescate del mismo.

2.2 Misión Primaria

La misión principal consiste en la toma de datos de temperatura, presión y humedad y el envío de los mismos mediante un transmisor de radio LoRa (Long Range) a un mínimo de 1 comunicación por segundo con el control de tierra.

Utilizaremos como controlador un Arduino Nano V3. Hemos elegido este módulo por su reducido tamaño y consumo, aparte de la facilidad que proporciona a la hora de programar los sensores y la planificación de los circuitos.

El módulo BMP280 realizará las tres mediciones de los datos correspondientes, los cuales serán recopilados y enviados a través de la antena del modelo RYLR 896.

Esta misión transcurre durante toda la etapa de vuelo del satélite, desde su despliegue a los 1000 metros de altitud hasta el aterrizaje en tierra. Para la recogida de datos se emplea una base en tierra con otra antena que se conectara a un ordenador para recopilar los datos que recoja el Arduino.



2.3 Misión Secundaria

La misión secundaria se conforma de tres puntos principales, siendo estos los siguientes:

2.3.1 GPS

Para obtener los datos del posicionamiento del CanSat en tiempo real se empleara una unidad de GPS NEO-6M de U-Blox. Este módulo es capaz de conseguir las coordenadas con una precisión del 2,5 m en posición y 0,5° en orientación, por lo que se obtendrá una posición casi exacta del mismo.

Dicho módulo desempeña dos funciones, seguimiento del CanSat, lo cual resultará útil a la hora de encontrar la ubicación del aterrizaje y la recopilación de los datos de la altura y la velocidad durante el descenso.

2.3.2 Telemetría avanzada

Esta misión está designada a la recolección de datos adicionales a los que requiere la misión principal, en nuestro caso elegimos recolectar datos relacionados con la calidad del aire y la aceleración del CanSat durante el descenso y el ascenso.

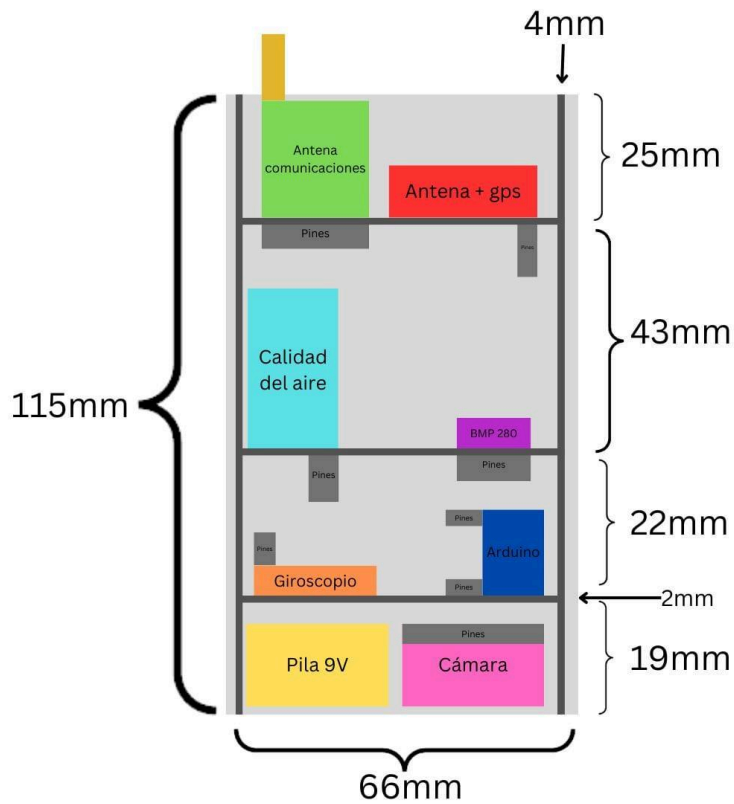
Para la primera misión se utilizará un módulo MQ-135, un sensor de calidad del aire de alta sensibilidad que se utiliza para determinar la presencia de ciertos gases en el aire que está en contacto con el módulo. Este módulo es capaz de medir NH₃, NO_x, alcohol, Benceno, Humo y CO₂ dando las medidas en partes por millón (PPM).

Para la segunda misión se empleara el módulo MPU 6050, un giroscopio y acelerómetro, el cual como su nombre bien indica, sirve para medir inclinaciones y aceleraciones respecto al propio sensor.

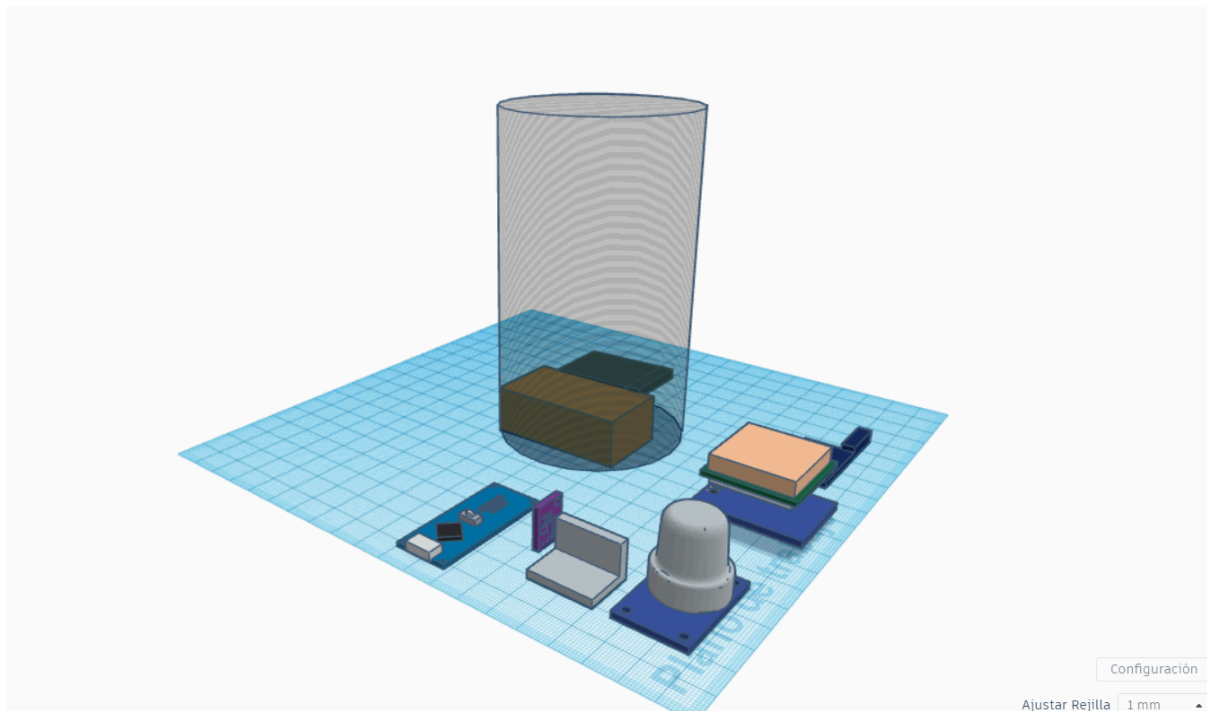
2.3.3 Registro visual del descenso

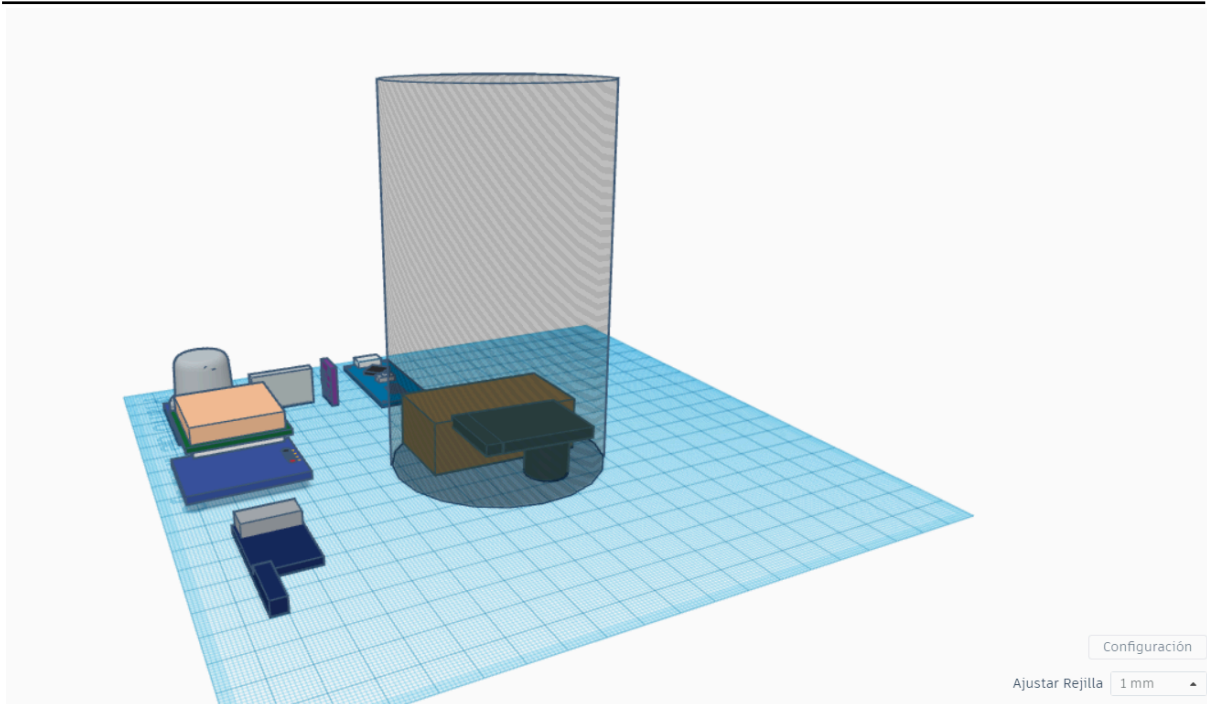
Esta misión está designada para la grabación del trayecto de descenso del CanSat. Para dicha tarea empleamos una cámara Arducam modelo OV2640 mini con una calidad de video de 2 megapixels y con la capacidad de grabar a calidad 1600x1200 a 15fps, o en caso de ser necesario una mayor tasa de frames, se podría reducir a 800x600 a 30fps.

2.4 Diseño Mecánico



Este es el diseño mecánico que hemos hecho de momento para empezar a diseñar la estructura interna del CanSat. Como se puede observar, hemos hecho los sensores y antena para poder estructurarlos dentro de la lata.





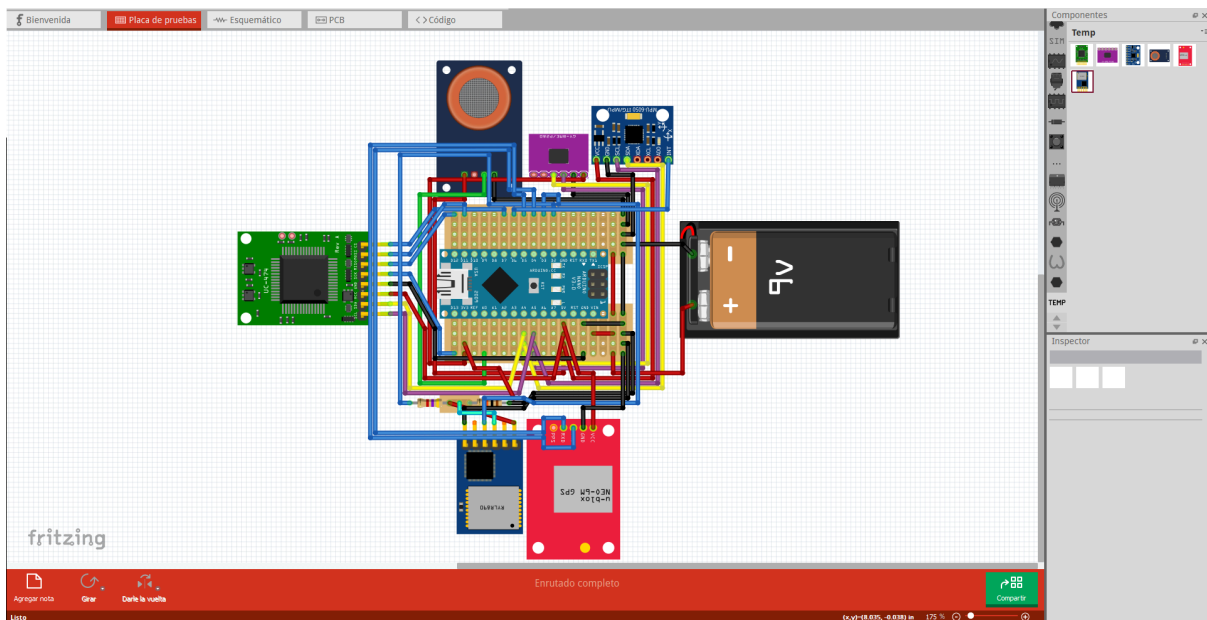
El diseño mecánico del cansat es una estructura cilíndrica de 115mm de alto y 66mm de diámetro. Dentro de este cilindro se encuentran los sensores y la unidad de procesamiento.

https://www.tinkercad.com/things/gNLMXktC0wG-lata-cansat/edit?returnTo=%2Fdashboard&sharecode=y1gsc81akA-Rrn-SdK9fpYsItwXEowZ_ouV2YZOVE

2.5 Diseño eléctrico

Ahora mismo se están barajando dos opciones para utilizar como ordenador central del CanSat, estos son un Arduino NANO y una Raspberry Pi Pico, siendo el primero más sencillo de programar pero teniendo menos pines y un menor control sobre todo el sistema, y la segunda opción siendo una con una mayor dificultad técnica y que requiere una mayor cantidad de tiempo requerido para aprender a utilizarse pero que otorga un control total sobre el CanSat.

De momento hemos estado haciendo pruebas con el Arduino debido a que no poseemos ninguna Raspberry Pi Pico todavía.



Diseño del circuito en Arduino con todos los componentes que se van a utilizar para la misión



2.6 Paracaídas

Con el fin de tener una caída controlada a una velocidad constante, el CanSat contará con un paracaídas que lo ayudará a frenar la caída durante el desarrollo de la misión.

Para la fabricación del paracaídas del cansat, debe tenerse en cuenta la masa, la velocidad a la que caerá el CanSat, la gravedad de la tierra ($9,8m/s^2$), la densidad del aire y la forma del paracaídas.

$$F_g = \frac{1}{2} \cdot C_d \cdot \rho \cdot A \cdot v^2$$
$$F_g = m \cdot g$$

Para que el CanSat caiga a velocidad constante, la fuerza de la caída y la de arrastre se deben anular. Tal que:

$$F_g = F_a$$
$$A = \frac{2 \cdot m \cdot g}{C_d \cdot \rho \cdot v^2}$$

Donde la densidad (ρ) del aire es aproximadamente, $1,225kg/m^3$, la velocidad límite a la que se espera que caiga el CanSat (v) es de $8m/s$, la masa del CanSat es de xg y C_d es un coeficiente que depende de la forma del paracaídas y se calcula de forma experimental.

Dicho paracaídas será fabricado a partir de una tela 3.8oz Polainas y unido al cansat con hilo monofilamento y tendrá una superficie de aproximadamente $0,14 m^2$ y seguirá uno de los patrones propuestos a continuación

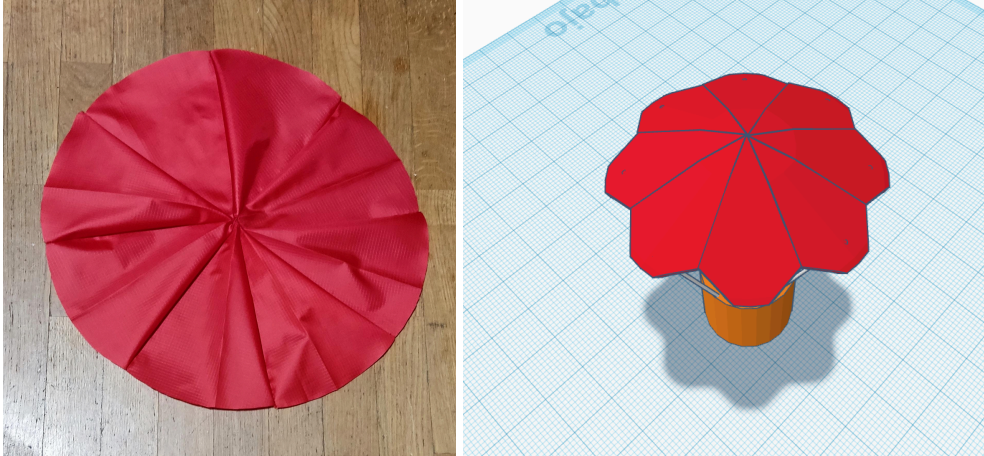
Paracaídas número 1:

Este paracaídas tendrá un radio de unos 22,5 cm de radio para que caiga a la velocidad requerida (entre 8 y 12 m/s). esto lo sabemos gracias a la siguiente fórmula:

$$r = \sqrt{\frac{2 \cdot m \cdot g}{\pi \cdot C_d \cdot \rho \cdot v^2}}$$

Los triángulos que se formarían en el modelo, estarían con lados redondeados, para obtener una mayor estabilidad.

Según las pruebas realizadas, hubo un cálculo erróneo a la hora de ajustar el radio, ya que era la sombra del paracaídas lo que debía medir 22,5. Este error resultó idóneo para poder ajustar con mayor exactitud esas longitudes.



A la izquierda: Imagen del modelo del paracaídas terminado.
A la derecha: Imagen del diseño del modelo del paracaídas.

Modelo número 2:

Este modelo está aún en proceso. Medirá 25 cm de radio, y tendrá un agujero de 3 cm de radio en el centro. Esto otorgará mayor estabilidad al CanSat.





Modelo número 3:

Este modelo es uno de los más estables de todos. Se trata de un paracaídas cuadrado. Gracias a los huecos que tiene en las esquinas lo hacen superestable.

Gracias a la siguiente fórmula hemos calculado las medidas del mismo.

$$A = \frac{2 \cdot m \cdot g}{C_d \cdot \rho \cdot v^2}$$

$$m = 350 \text{ kg}$$

$$C_d = 0,80$$

$$v = 6 \text{ m/s}$$

Después de calcular el área que este abarca, calculamos el lado del paracaídas (recuerden que este será cuadrado).

Como podrán ver en la foto del final del modelo, este tendrá unos "salientes" del 20% del largo del paracaídas.

Entonces el paracaídas tendrá las siguientes medidas:

Lado del paracaídas: 41,89 cm

Longitud del saliente: 8,37 cm

Hemos calculado el lado con una velocidad de 6 m/s ya que hemos tomado en cuenta la densidad del aire de Madrid. Mas el lanzamiento definitivo se ejecutará en Cádiz, con una mayor densidad en el aire que Madrid.

Sin embargo, este paracaídas está todavía en fase de pruebas, ya que el modelo que hicimos tenía algunos defectos en las medidas.





Modelo número 4:

El nuevo paracaídas triangular, utilizado incluso en las fuerzas militares de Colombia (también llamado X-TRIANGLE) es extremadamente estable al péndulo gracias a su CONSTRUCCIÓN EN 3 VÉRTICES, la forma triangular tiene un efecto auto estable.

Este paracaídas tiene una teoría similar al anterior, salvo por la forma triangular de la base. Como el anterior, tiene unos salientes que ayudan a estabilizar el CanSat.

Con el área obtenida anteriormente, calculamos el lado del triángulo, y como antes, hacemos unos salientes del 20%.

Entonces las medidas serían:

Lado del triángulo: 59,24 cm

Longitud del saliente: 11,85 cm

Este paracaídas está aún en fase de pruebas, pero está dando muy buenos resultados, y posiblemente sea el definitivo.

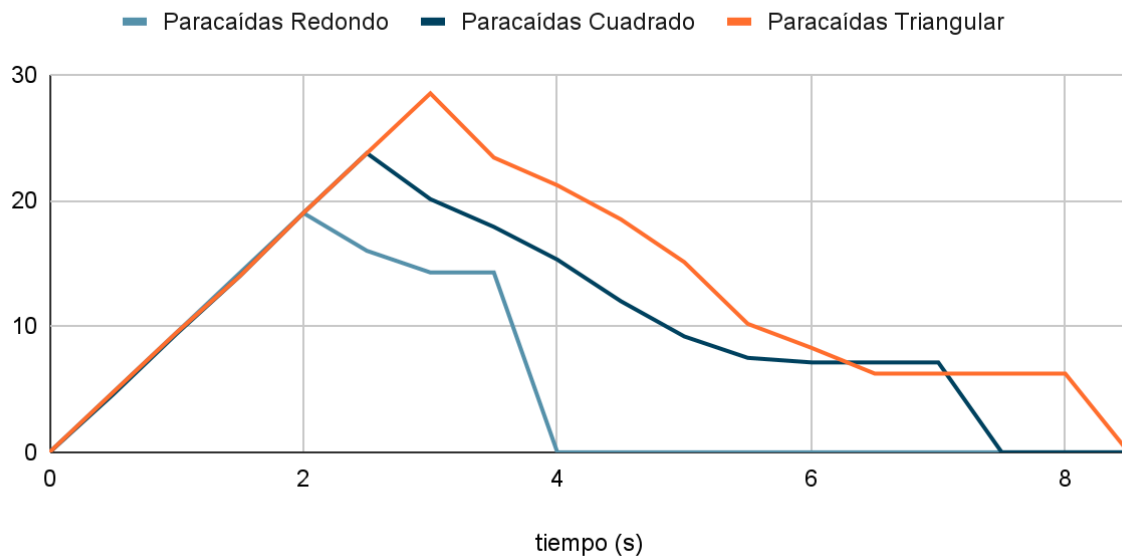


Paracaídas triangular



Gráficos de los paracaídas

Paracaídas Redondo, Paracaídas Cuadrado y Paracaídas Triangular





2.7 Estación de tierra

Hemos comenzado las pruebas de comunicaciones entre el Arduino del CanSat y un ordenador que reciba, intérprete y almacene los datos transmitidos, a través de la antena RLYR 896.

https://how2electronics.com/reyax-rylr890-lora-module-with-arduino/#google_vignette

Esta antena posee un rango operativo de hasta 15 km, el cual supera con creces lo que requiere para la misión.

La transmisión de los datos se realizará de manera que haga al menos 1 por cada segundo, la cual es el mínimo impuesto por la competición, aunque la antena tiene la capacidad de emitir las señales cada menor periodo, y se tendrá en cuenta a la hora de crear el código final para la transmisión de datos.

Los dos encargados de la electrónica y programación estamos barajando la opción de crear un programa que te permita ver los datos en tiempo real de manera gráfica, no solo a través del monitor en serie, aparte de almacenarlos para que sea más fácil interpretarlos durante la misión.



2.8 Software

BMP280

```
C/C++
#include <SoftwareSerial.h>
#include <Wire.h>
#include <SPI.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_BME280.h>

SoftwareSerial lora(2,3);

#define SEALEVELPRESSURE_HPA (1013.25)
Adafruit_BME280 bme;

void setup()
{
  // put your setup code here, to run once:
  Serial.begin(115200);
  lora.begin(115200);
  if (!bme.begin(0x76))
  {
    Serial.println("Could not find a valid BME280 sensor, check wiring!");
    while (1);
  }
}

void loop()
{
  float temperature = bme.readTemperature();
  float pressure = bme.readPressure() / 100.0F;
  float altitude = bme.readAltitude(SEALEVELPRESSURE_HPA);
  float humidity = bme.readHumidity();

  Serial.print(F("Temperature = "));
  Serial.print(temperature);
  Serial.println(F("°C"));

  Serial.print(F("Pressure = "));
  Serial.print(pressure);
  Serial.println(F("hPa"));
```




```
Serial.print(F("Approx. Altitude = "));
Serial.print(altitude);
Serial.println(F("m"));

Serial.print(F("Humidity = "));
Serial.print(humidity);
Serial.println(F("%"));

String temp = String(temperature);
String pres = String(pressure);
String alt = String(altitude);
String hum = String(humidity);

String values = String(temp)+","+ String(pres)+","+ String(alt)+","+
String(hum);
String cmd = "AT+SEND=0,"+String(values.length())+","+values;
lora.println(cmd);

while(lora.available())
{
  Serial.write(lora.read());
}
Serial.println();
Serial.println(cmd);
delay(5000);
}
```

MQ-135

```
C/C++
void setup(){
  Serial.begin(9600);
  pinMode( 0, INPUT);
}

void loop(){
  sensorValue = analogRead(0);
  digitalValue = digitalRead(0);
  Serial.println(sensorValue, DEC);
}
```



```
Serial.println(digitalValue, DEC);  
delay(2000);  
}
```

MPU 6050

```
C/C++  
#include "I2Cdev.h"  
#include "MPU6050.h"  
#include "Wire.h"  
MPU6050 sensor;  
int ax, ay, az;  
int gx, gy, gz;  
  
void setup() {  
  Serial.begin(57600);  
  Wire.begin();  
  sensor.initialize();  
  
  if (sensor.testConnection())  
    Serial.println("Sensor iniciado correctamente");  
  else Serial.println("Error al iniciar el sensor");  
}  
  
void loop() {  
  sensor.getAcceleration(&ax, &ay, &az);  
  sensor.getRotation(&gx, &gy, &gz);  
  
  Serial.print("a[x y z] g[x y z]:\t");  
  Serial.print(ax); Serial.print("\t");  
  Serial.print(ay); Serial.print("\t");  
  Serial.print(az); Serial.print("\t");  
  Serial.print(gx); Serial.print("\t");  
  Serial.print(gy); Serial.print("\t");  
  Serial.println(gz);  
  
  delay(100);};
```



Neo 6M

Enlaces a los recursos utilizados

<https://github.com/mikalhart/TinyGPSPlus>

<https://randomnerdtutorials.com/guide-to-neo-6m-gps-module-with-arduino/>

OV2640

Hemos empezado con el código de la cámara OV2640 usando las librerías Arducam y hemos conseguido grabar vídeos. [Enlace a la librería utilizada](#)



3. Planificación

3.1 Planificación del proyecto

Al principio del proyecto se planificó las fases del proyecto, la gestión de materiales y la organización de roles. Teniendo en cuenta la difusión del proyecto y apoyo externo.

Planificación	PRIMER TRIMESTRE		SEGUNDO TRIMESTRE			TERCER TRIMESTRE
	NOV	DIC	ENE	FEB	MAR	ABR
Planteamiento del proyecto	■					
Obtencion de componentes y programacion de ellos		■				
Pruebas de los componentes			■			
Diseño de estructura			■			
Contacto con empresas y acciones de difusion			■			
Construccion definitiva						■
Pruebas de estructura y compatibilidad					■	

3.2 Estimación del producto

3.2.1 Presupuesto

Componente	Precio unitario	Cantidad	Precio final
Arduino Nano	9,99€	2	19,98€
BMP 280	1,99€	1	1,99€
NEO 6MW2	9,99€	1	9,99€
Pilas 9V Recargables	11,42€	2	22,84€
Base carga pilas	11,99€	2	11,99€
MPU 6050	3,48€	1	3,48€



RLYR 896	18.00€	2	36.00€
OV2640	25.99€	1	25.99€
Pack de Cables (120)	8€	1	8€
Pack de Resistencias (400)	4,99€	1	4,99€
Tela Paracaídas Ripstop 2.8oz 2 metros	11,25€	1	11,25€
Hilo Monofilamento 0,5mmØ, 200m	5,99€	1	5,99€
Total: 154,49€			

3.2.2 Financiación y apoyo externo

Estamos financiados por el departamento de tecnología del instituto Ramiro de Maeztu.



4. Programa de difusión

A lo largo del proyecto se han llevado a cabo varias actividades con el propósito de difundir y divulgar sobre el proyecto, el equipo y la competición.

- Comunicación boca a boca: Los miembros del equipo comentaron con sus familias el proyecto para que estos lo contarán en puestos de trabajo etc.
- R.R.S.S: El equipo posee una cuenta de instagram y twitter, en las cuales se publican avances del proyecto. También estamos desarrollando una página web, en la que publicaremos avances del proyecto con menos frecuencia pero más detalladas
 - Enlace a la cuenta de Instagram: @Cansat Ramiro24
 - Enlace a la cuenta de Twitter: @CanSat_Ramiro24
- Carteles: Se emplearon carteles para la difusión de las R.R.S.S del equipo los cuales se colocaron dentro del recinto escolar para dar a conocer el proyecto a otros alumnos.

5. Requisitos para lanzamiento

Características	Unidades
Altura del CanSat	115 mm
Diámetro del CanSat	66 mm
Masa del CanSat	350 g
Velocidad de descenso	8 m/s
Tiempo de vuelo	12´5 s
Frecuencia de transmisión	868 MHz
Autonomía	12 h
Coste total del CanSat	154´49€



6. Observaciones

Programas utilizados para la elaboración de este proyecto:

- Diagramas:
- Diseño Electrónico: Fritzing.
- Informe: Google Docs.
- Diseño mecánico e impresión 3D: Tinkercad.
- Programación CanSat: Arduino IDE.
- Programación estación de tierra:

7. Bibliografía

<https://esero.es/wp-content/uploads/2021/09/plantilla-informecdr-cansat.pdf>

<https://esero.es/wp-content/uploads/2023/12/plantilla-pdr-cansat-2024-v3.pdf>