

ICAM'E TEOR



Final Report



16/04/2024

Summary



• 1) The Team	5
• 2) The CanSat.....	5
◦ a. Context of our project	5
◦ b. Aim of the project.....	6
• 3) Plannification.....	8
◦ a. Agile Method.....	8
◦ b. User's stories.....	8
◦ c. Backlog of Product and sprint.....	9
◦ d. Gantt Diagram and Burndown chart.....	10
• 4) Functionnal analysis.....	11
◦ a. Horned-Beast Diagram.....	11
◦ b. SADT Diagram.....	11
◦ c. Octopus Diagram.....	12
• 5) State of Art.....	13
◦ a. State of Art	13
◦ b. Decision Matrix	14
• 6) Environmental planning	16
◦ a. Context	16
◦ b. Environmental Aspect.....	16
◦ c. Economical aspect	17
◦ d. Social Aspect	17
◦ e. Risk analysis	17
• 7) Technical Specifications.....	18
◦ a. Technical Specifications.....	18
◦ b. FAST Diagram.....	20
• 8) Resolution Process.....	21
◦ a. Overall Resolution Process.....	21
• 9) Components.....	22
◦ a. Bill of material.....	22
◦ b. List of components.....	23
• 10) Electronics.....	26
◦ a. Wiring Plan.....	26



Summary



• 11) Parachute	27
◦ a. Prototype.....	27
◦ b. Fabrication.....	27
• 12) Mechanical conception.....	29
◦ a. 3D conception.....	29
◦ b. 3D printing.....	31
◦ c. Testing.....	31
◦ d. Assembly.....	32
• 13) Design.....	33
• 14) Code.....	34
◦ a. Coding objectives	34
◦ b. Choice of language.....	34
◦ c. Current advancement.....	35
• 15)Results.....	36
◦ a. Gap between the results and the objectives.....	36
• 16) Conclusion.....	37
◦ a. Evolution of sprint.....	37
◦ b. Conclusion.....	38
• Appendix.....	40





Table of illustrations

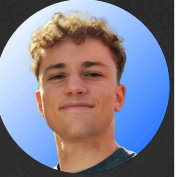
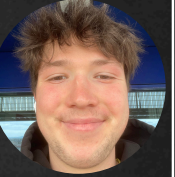


Figure 1 : Sprint 6 Backlog.....	9
Figure 2 : Annual Burndown Chart	10
Figure 3 : Horned beast diagram.....	11
Figure 4 : SADT Diagram.....	11
Figure 5 : Octopus Diagram.....	12
Figure 6 : Decision matrix for parachutes.....	14
Figure 7 : Decision matrix for the material.....	14
Figure 8 : Decision matrix for the landing system.....	14
Figure 9 : Decision matrix for painting system.....	14
Figure 10 : Decision matrix for Flight system.....	15
Figure 11 : Decision matrix for the Power supply.....	15
Figure 12 : Decision matrix for the CPU.....	15
Figure 13 : Risk Analysis.....	17
Figure 14 : Technical Specifications.....	19
Figure 15 : FAST Diagram.....	20
Figure 16 : Bill of Material (BoM).....	22
Figure 17 : Temperature sensor circuit.....	25
Figure 18 : Temperature Sensor Data.....	25
Figure 19 : Duracell 223 CR-P2 6V lithium.....	26
Figure 20 : Parachute Flat Sail	27
Figure 21 : Triangular shaped Parachute.....	28
Figure 22 : Folding Tutorial of the parachute.....	28
Figure 23: Body of CANSAT.....	29
Figure 24: Pod conception.....	29
Figure 25: Expected design.....	30
Figure 26: Cap of CANSAT.....	30
Figure 27: Inside of CANSAT.....	30
Figure 28: Interior Design face 1.....	30
Figure 29: Interior Design face 2.....	30
Figure 30: Interior Design face 3.....	30
Figure 31: Interior Desing + Wiring.....	31
Figure 32: Nastran Simulation.....	31
Figure 33: Prototype after first test.....	32
Figure 34: Interior Pin.....	32
Figure 35: Poka-yoke.....	32
Figure 36: Based design.....	33
Figure 37: Stencil.....	33
Figure 38: Final design.....	33
Figure 39: Test Temperature sensor.....	35
Figure 40: Image recognition.....	35
Figure 41: Flight Algorithm.....	35
Figure 42: Entire device + Parachute.....	38



1) The Team



 Mathis <i>Programmation (Scrum master)</i>	 Bruno <i>Electronique (Product owner)</i>	 Rémi <i>Electronique</i>	 Antoine <i>Programation</i>	 Edgar <i>Mecanique/ Conception</i>	 Célio <i>Mecanique/ Conception</i>
---	--	---	--	--	---

For this project, the team is composed of 6 people : Mathis BOUTIN, Bruno HENRY, Célio JOHNSON, Edgar KINDER, Rémi LAGARRIGUE and Antoine MERY. It was decided to apply the Agile Methodology, which enhances flexibility and efficiency by involving client participation in project management, throughout the year.

Pairs were formed within the group to focus each member's work on their area of interest—mechanics, electronics, and coding. Those working on mechanics were responsible for designing the CanSat that fits all devices perfectly, making all necessary calculations, and ensuring the CanSat meets technical specifications. The coding team determined and wrote the flight algorithm and performed various tests to refine the code and the algorithm. The electronics team worked closely with both the coding and mechanical teams, acting as a link between the groups to select the ideal components that met our requirements and to create a wiring plan that aligned with the coding specifications, those aspect will be detailed further in the report

To facilitate communication in the team, a messenger group chat was created.

2) The CanSat

a. Project Context

Planète Sciences, established in 1962, provides young people with opportunities to explore scientific and technological subjects such as astronomy, robotics, and environmental science in a dynamic, team-oriented setting. Participants can build rockets, CanSats, and robots, study biodiversity, discuss pertinent topics related to the environment and technology, and use resources like fablabs or astronomical observatories.

They also have the chance to participate in various competitions including the Trophies, Robot Challenge, and Rocketry Challenge. This program fosters practical engagement and teamwork, enabling individuals to realize their capabilities, value collaboration, and hone critical thinking skills.

2) The CanSat

b. Project Objectives

The CanSat projects, organized by CNES (Centre National d'Études Spatiales) and Planète-Sciences, are primarily designed for students in technology, physics, and programming. By offering this small-scale space project experience, both the student CanSat competition and the high school competition approach these subjects in an interdisciplinary manner, promoting collaboration and teamwork.

The capable of autonomously navigating to a point in space using GPS coordinates (implemented in the CanSat software) and marking a 1m² target. The device is released by a drone at an altitude of 80m, after which it reaches the target on its own, marks it, and stands upright upon landing.



Primary Mission: Marking the Position

- GPS localization of the area
- Autonomous navigation to the target using GPS coordinates
- Ability to modify stored GPS coordinates within the satellite
- Use of biodegradable/environmentally friendly paint (supported by a reference)

Secondary Missions:

1. The can must remain upright during flight for a minimum of three minutes (considering the nature of the terrain).
2. Design and manufacture a sensor from scratch. The choice of the sensor must be justified.
3. After landing, the CanSat must be able to raise a flag measuring at least 5x8cm. The flag should not touch the can once deployed.

The CanSat must fulfill specific missions. It must meet all primary missions, and for the secondary missions, at least one must be chosen. For the secondary mission, the decision was made to make a home-made sensor.

The project budget should not exceed €300. However, effective negotiation with the client for a valid reason may lead to a higher amount. For instance, if there is any need to purchase special equipment, its use can be justified with the client.

2) *The CanSat*

Who is the client ?

Name : Grégoire CHABROL

Objective : "He represents the client to whom the product is intended to be sold. For this, the primary and secondary objectives linked to our cansat must be achieved".



Who is the supervisor?

Philippe RAGOT Objective: "As project manager, presenting the progress of our research at the beginning of each week is the team's responsibility, with their support in case of misunderstanding on certain project topics."



Who are the Technical Experts ?

Marcelin BILASSE : *Mechanical Expert*

Guillaume COUDREUSE : *CAO Expert*

Ahmad KAZEMI : *Electric Expert*

The Technical Experts role in the project is to help or to give advices if there are trouble/issue of the completion of a task that fits their domain of expertise



3) Planification

a. Agile Methodology

To successfully implement this project, the decision was made to adopt the Agile methodology. This project management strategy breaks down the project into manageable phases, focusing on continuous collaboration and improvement, and utilizes a range of powerful tool.

- User Stories
- Product and Sprint Backlog
- Gantt Chart
- Burndown Chart

The Agile methodology, organized into sprints, guides the project management approach. At the start of each sprint, specific user stories are selected to accomplish. At the sprint's end, achievements are evaluated against initial expectations, and future projections are made for the next sprint. This process helps identify successful strategies and areas needing improvement. Notably, the freedom to choose sprints was not available as they were predetermined by clients for the entire year.

The choice of this method is due to its familiarity and past successes in similar projects. Additional training on its application has enhanced understanding of client requirements and clarified project direction.

b. User's Stories

3) Planification

c. Product backlog and Sprint Backlog

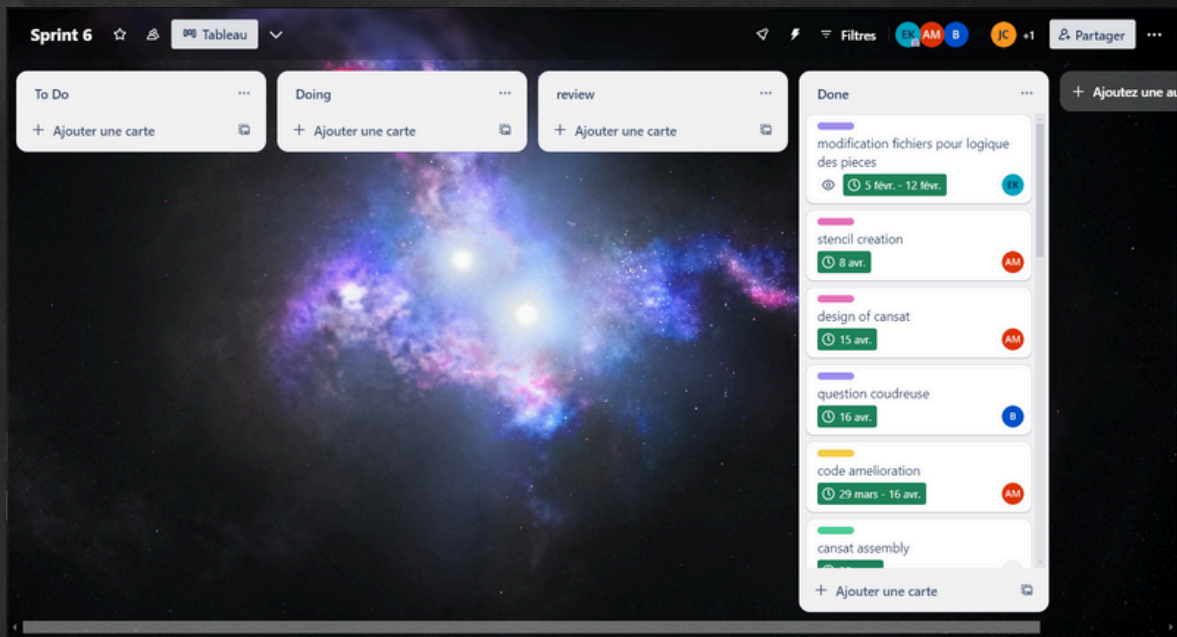
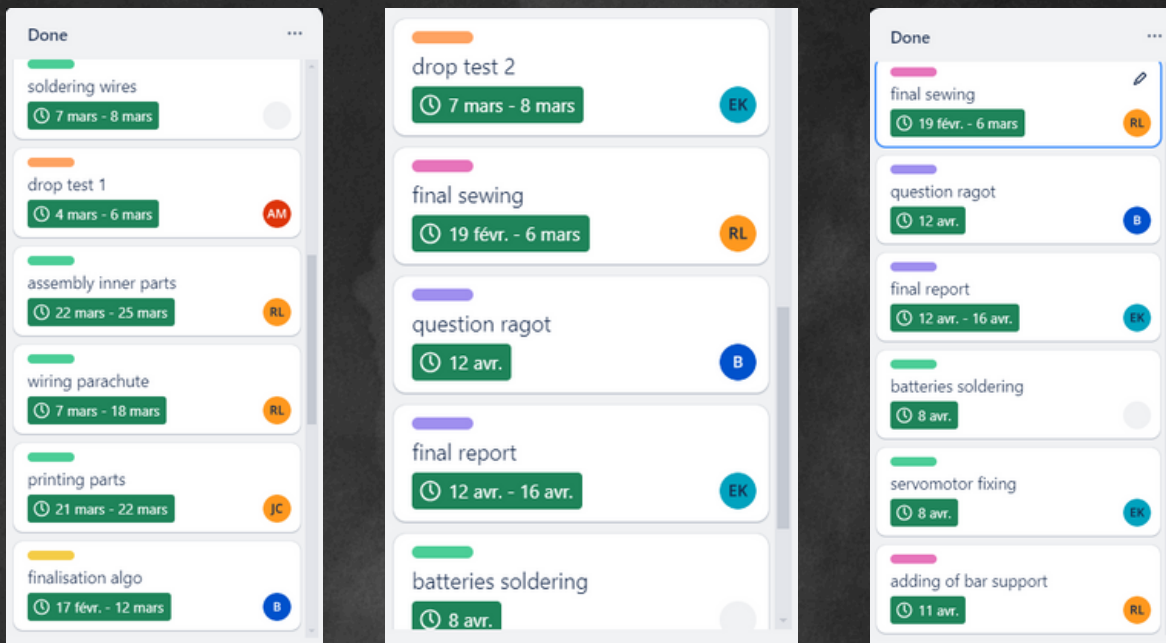


Figure 1



The product backlog was a resume of the big tasks that were to be accomplished through the project. Concerning the sprint Backlog, there were six of them, one for each sprint, that details every single action that was achieved (like sending an email).

During the entirety of the sprint, tasks were moved from columns to columns to specify the degree of achievement ("to do" => "Ongoing" => "Review" => "Done"). Concerning the product backlog, it was updated at the end of each sprint in order to observe the general advancement of the project. The following figure is the backlog of the sprint 6, if any more information is needed concerning the backlog, see the apendix.

3) Planification

d. Gantt Diagram and Burndown chart

After establishing the User Stories and backlogs, both a Gantt diagram (cf. apendix) and a Burndown chart were created to track the project's progress.

The Gantt diagram, initially crafted at the project's onset, served as a dynamic tool for monitoring overall advancement. It underwent adjustments as the project progressed; tasks are marked in green to indicate completion, while red signifies incomplete tasks. Modifications were made to reflect actual time spent on tasks, deviating from initial estimates. Upon completion of a task, as requested by the client, a cross is added on the completion date to signify 100% completion.

The entire Gantt diagram is in the appendix, since the screenshots from it were blurry.

The Burndown chart, though considered less critical for our internal team, proved valuable for the supervisor, Mr. Ragot. It provided the supervisor with a straightforward visual representation of how the project was advancing, helping to quickly gauge progress against planned timelines. Here is the burndown chart of the project over the year

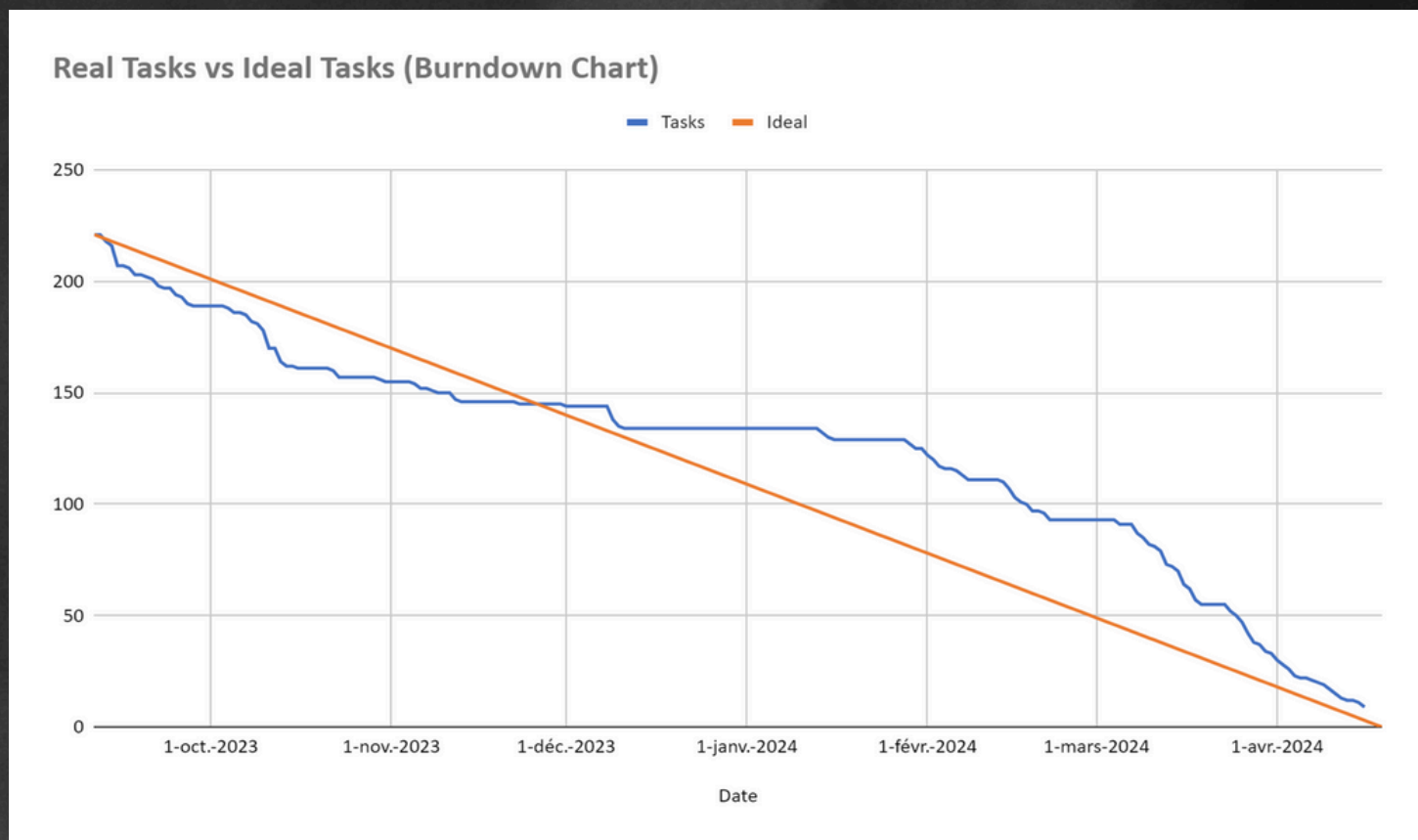


Figure 2

4) Functional analysis



a. Horned-Beast Diagram

The Horned-Beast Diagram initiates the technical analysis of a product, defining the beneficiaries, functionalities, and applications of the product, providing a comprehensive understanding of its utility and impact.

The product is tailored for the customer, Mr. Chabrol, designed to autonomously move in the air and mark a target using a raspberry lens. This clear objective helps in outlining the main functions of the product, setting the stage for defining the essential functions required to meet the specified objectives and requirements.

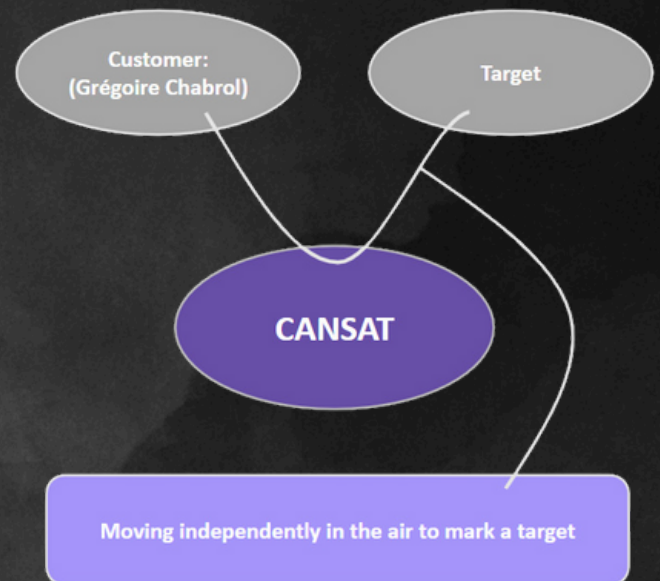


Figure 3

b. SADT Diagram

The SADT (Structured Analysis and Design Technique) diagram is a visual tool that helps in analyzing and representing system structures and processes, facilitating improved communication among stakeholders. This diagram provides a structured view of the necessary components listed on the left-hand side, which helps in planning purchases according to budget and deadlines. Notably, it emphasizes long lead times for items like electronic boards, critical for project scheduling. Control data essential for the operation of the CanSat satellite, and support resources such as access to FabLabs and 3D printers, are integral for the execution of the project. The heart of the diagram, the central part, outlines the project and specific missions of the CanSat satellite, detailing the functions and expected performance. This diagram shows output products, which include both the unintended waste and the required deliverables like the fully functional CanSat satellite. Overall, the SADT diagram serves as a comprehensive visual representation that supports clarity and coordination between project stakeholders.

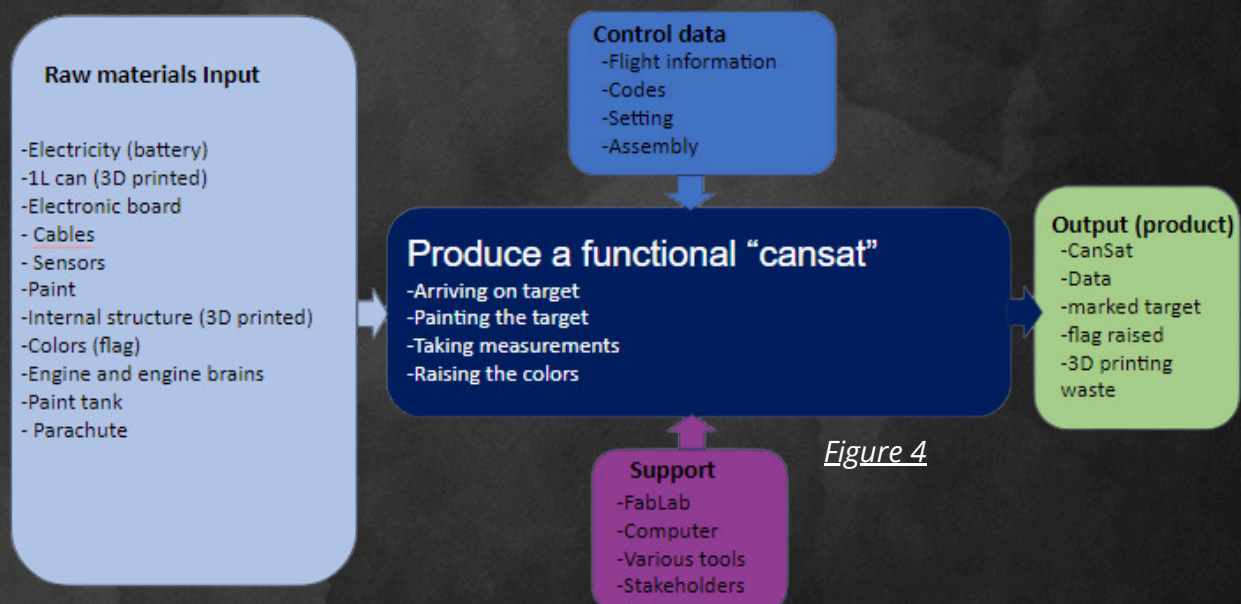


Figure 4

4) Fonctionnal analysis

c. Octopus diagram

The function analysis model diagram, often referred to as the octopus diagram, is a tool used to illustrate the various elements that will have an impact on the system, and their interactions. This diagram is therefore particularly useful for analyzing and visualizing the complexity and structure of the technical features of the project.

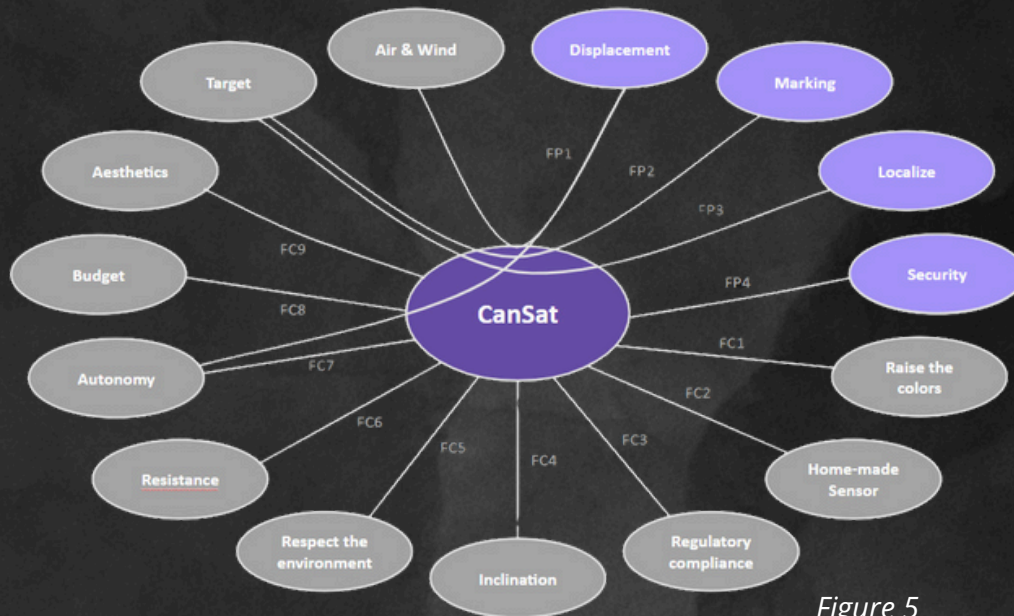


Figure 5

The elements affecting the system fall into two categories:

- Principal functions (PF): these are the essential tasks that the system must perform to meet user needs and objectives.
- Constraint functions (CF): these are tasks which, while not as critical as the main functions, nevertheless contribute to the proper functioning of the product and the satisfaction of user needs.

Based on the customer's requirements, three main functions were identified that encompasses the main objectives of our system: the ability to move autonomously in the air (FP1), the ability to mark the target (FP2) and the ability to acquire the target's location (FP3). In addition, The security function is also a principal function because safety is very important.

In addition to these main functions, nine constraint functions were identified that are necessary to meet the customer's expectations and ensure the proper functioning of the product. These constraints include compliance with safety measures, product durability and vertical landing capability.

Given the multitude of functions involved in the development of the system and the complex interactions between them, the complexity of the structure is showing up.

The documentation of all these functions will form the basis of the functional specification sheet.

5) *State of art*

a. State of the art

Here are the different choices of the principal parts of the CanSat:

- **Servo Motors:**

1. Miniature Analog Servo Motor FS0403: This servo motor is a miniature analog model.
2. Miniature Servo Motor MG91: The MG91 is another miniature servo motor.
3. Metal Servo Motor FT3325M/35063: The FT3325M/35063 is a metal servo motor.

The metallic servo motor is the one chosen for more impact resistance as well as a strong torque.

- **Electronic Boards:**

- 1.Arduino UNO: The Arduino UNO is an electronic board used for control and data processing.
- 2.Raspberry Pi 4: The Raspberry Pi 4 is another electronic board used for specific functions.
- 3.Raspberry Pi 0: The Raspberry Pi 0 is a compact version used for specific tasks.

The Pi 0 has Bluetooth and a Wi-Fi system integrated, which allows the acquisition of data from a distance, ideal for this project.

- **Camera**

- 1.Husky lens
- 2.Raspberry Pi Camera Module 3

For easier correspondence and link between the board and the camera, the Raspberry Camera is the best option.

- **3D Printing Material:**

The CanSat's outer casing is 3D printed using either PETG or PLA material.

- **Paint Deployment System:**

- 1.The paint deployment system includes a shower for liquid paint, paintballs, and sponges attached to the CanSat's legs.
- 2.The paint is released directly onto a surface, creating a paint splatter effect.

The sponges can combine marking the target and landing smoothly on it, a real asset, moreover, it is a very light material, perfect for load limitation.

5) State of art

- **Parachute Type:**

The CanSat is equipped with a parachute that can be round, paraglider-shaped, or triangular-shaped in design.

For more stability and innovation, the triangle shape was chosen.

- **Landing System:**

The CanSat's landing system involves a deployable tripod or balloons filled with pressurized gas.

Pressurized gas balloons would involve gas cartridge, which would add to much load to the system. A mechanical system as tripods deployment is easier to develop and less random in operation. Using this system allows to link with the paint dropping. That's why the tripods system is selected here.

b. Decision matrix

After conducting a state-of-the-art analysis to review various technical solutions, decision matrices were utilized to facilitate informed choices. These matrices categorize different aspects of each technical solution, ranking them with coefficients from one to five. Each solution is also scored from one to five based on its importance for each criterion, ensuring uniformity across all decision matrices for consistency. The decision matrices are used to evaluate choices concerning materials, the landing system, paint selection, the flying device, type of power supply, parachute design, and CPU selection

All the solutions in green are the technical solutions that will be used for the CanSat.

Parachute *Figure 6*

Criteria	Maneuverability		Stability		Paint		Impact		Total
Coeff	5		4		1		3		
Triangle	2	10	4	16	3	3	3	9	38
Crossed	1	5	5	20	4	4	2	6	35
Paraglider	4	20	2	8	2	2	1	3	33

Material Used *Figure 7*

Criteria	Production Ease		Mass		Production Time		Core Strength		Design		Price		Total
Coeff	2		4		2		5		3		2		
Impr 3D	5	10	4	16	4	8	3	15	2	6	4	8	57
Wood (Plywood)	4	8	5	20	3	6	2	10	3	9	3	6	50
Aluminium	2	4	2	8	1	2	5	25	5	15	1	2	41

Landing System *Figure 8*

Criteria	Core Strength		Simple Deployment		Structural simplicity		Mass		Price		Volume		Total Pondéré
Coeff	5		4		4		3		2		5		
Tripod	3	15	5	20	5	20	3	9	4	8	3	15	87
Balloons (inflation system)	4	20	1	4	2	8	2	6	1	2	2	10	50
Base deployment (mecanical)	4	20	3	12	3	12	4	12	4	8	4	20	84

5) State of art

Painting System *Figure 9*

Criteria	System Simplicity		Painted Zone		Chance to paint the target		Structural Simplicity		Mass		Price		Volume		Total
Coeff	5		2		4		4		3		1		5		
Shower system	2	10	3	6	4	16	2	8	3	9	4	4	2	10	63
Sponges	5	25	1	2	2	8	5	20	5	15	5	5	5	25	100
Direct Release	3	15	5	10	2	8	3	12	3	9	4	4	2	10	68

Flight system *Figure 10*

Critère	Easiness		Precision		Challenge		Total
Coeff	2		5		1		
GPS	5	10	1	5	3	3	18
PI0	1	2	4	20	5	5	27

Power Supply *Figure 11*

Critère	Availability		Environmental Impact		Total
Coeff	5		5		
Resusable	2	10	5	25	35
Non-reusable	5	25	3	15	40

CPU *Figure 12*

Criteria	Lien carte-caméra		Consomation		Puissance		Taille		Total
Coeff	4		5		1		3		
SEED	2	8	2	10	5	5	3	9	32
Arduino	2	8	4	20	2	2	2	6	36
Raspberry	5	20	4	20	3	3	3	9	52

6) *Environmental Plan*

a) Background

During the CanSat project, the necessity of having a "green" aspect, which encompasses environmental, social, and economic dimensions. Implementing this green plan is crucial as it aligns with the customer's requirements, along with legal and environmental regulations. Integrating an ecological perspective is essential to ensure the project is both sustainable and environmentally friendly.

The green plan involves considering the interior and exterior design of the CanSat and conducting a risk analysis to identify and evaluate potential hazards. By categorizing and managing these risks, the target's aim is to mitigate them effectively, ensuring both the project's success and its sustainability.

b) The environmental aspect

The ecological aspect is a central focus of this CanSat project, highlighting a commitment to sustainability—a crucial concern for the future of engineers. Here's how it was planned to manage the components of our CanSat for environmental responsibility:

Reused Components:

- CanSat Body + Interior Design: Made of recyclable PETG.
- Electronics: Includes servomotor, wirings, Pi0, Raspberry Lens, PT100, and switch, all of which will be reused in future technical projects.
- Parachuting System: Consisting of the parachute and hangers, reusable in upcoming CanSat projects.
- Battery Holders and Sponges: Both recyclable.

Components Not Suitable for Reuse:

- Batteries and Wires: Reusable batteries were not an option because they didn't exist as the size and capacity needs were met. The wires have been modified (welded) for specific use in this CanSat and cannot be reused.

This approach ensures that nearly every part of the CanSat contributes to a cycle of sustainability, minimizing environmental impact.

6) Environmental Plan

c) Economic aspect

The economic aspect is also important since the budget allocated by the client has to be respected. The client allowed a budget of 300 euros, where 273,84 euros were used. For more details about the specific spendings, (Cf BoM on figure 16, pg 22)

d) Social aspects

The Cansat project has a great social benefit, as it allows students to work on a multidisciplinary project that requires a lot of hard work. Communication, group work, and knowledge sharing are the 3 main points of this project. Thanks to it, it is now able to compete with students from all over France.

e) Risk analysis

Risk analysis is a way of grading the risks that the project will entail. In this case, taking into account all external constraints and the safety measures put in place by CNES, risk analysis is already at the lowest possible values. In fact, the values used are calculated on the basis of two factors. The first is the recurrence of the problem, which is the probability that it will occur, and the second is the physical danger that the project could cause.

risk analysis		severity					likelihood	severity
		insignificant	minor	moderate	major	severe		
likelihood	almost certain						1	1
	likely						2	1
	possible	3					2	1
	unlikely	2/3/5/6/8					2	1
	rare	1/4/ 9				10	3	1
							8	1
							9	1
							10	5

Figure 13

According to this analysis, there is no need to do any improvements in order to make the CanSat safer because it has no risks at all. Nonetheless, there are added rules that must be followed when the test are being done and that are being applied during the further tests as well as while working on the hardware/software. Those rules are as follows :

- Always wear protection equipments while working with tools.
- Always be aware of the surroundings while testing.
- Always communicate all informations to not be surprised by something.

7) *Technical Specifications*

a. Functional specification sheet

The main functions and constraints identified in the function analysis model diagram serve as the basis for the functional specification sheet.

The purpose of this document is to formally structure the project objectives and precisely define the expected features, based on the customer's needs in a hierarchical manner.

As a result, each main function identified above can be broken down into several sub-functions, creating a more detailed and comprehensive approach. This detailed breakdown of functions is useful to better understand the project's objectives and to organize the product's technical features.

By closely following these functions and their sub-functions, various criteria can be identified that serves as a guideline for product development. These criteria include technical specifications, quality standards, performance requirements and other relevant factors. They enabled the establishment of precise parameters to which the product must conform in order to meet customer expectations. They also provide a flexibility indicator for assessing the degree of flexibility required for each criterion.

7) Technical Specifications

Fonction description:	Sub-Function:	Function criteria:	Level of criteria:	Flexibility:	Exigence:
FP1: Moving independently in the air		Mode of travel	Controlled fall	F1	1
		Travel time after dropping	<5min	F0	2
		Descent speed	2m/s < X < 15m/s	F0	3
		Drop height	80m < X < 120m	F0	4
		Autonomous operation	Complete	F0	5
FP2: Mark the target		Target diameter	2m	F0	6
		Marked surface	The most centered	F1	7
			Maximum surface area	F1	8
		On-board paint volume	<100mL	F3	9
FP3: Acquire location of the target		Precision	±30cm	F1	10
		Detection capacity	>120m	F0	11
FC1: CanSat must be able to land straight		Inclination	Vertical	F4	12
FC2: Must integrate a home-made sensor		Studied temperature data range	±5°C	F4	13
FC3: Must comply with regulations	Physical product characteristics	Mass	<1kg		14
		Width (without parachute)	< 80mm	F0	15
		Height (without parachute)	<200mm	F0	16
	Use of frequency bands	Usable frequency bands, Max. frequency	433Mhz, <10mW	F0	17
			869.4MHz < X < 869.65MHz,	F0	18
			2.4GHz, <100mW	F0	19
			5.8MHz, <25mW	F0	20
			GSM Frequences	F0	21
FC4: Must comply with safety requirements		Protective device for cutting and rotating parts	Fairings	F0	22
		Dangerous substances and pyrotechnics	None	F0	23
		Application range for pneumatic systems	<10bar	F0	24
		Maximum voltage differential (Vmax Vmin)	<30V	F0	25
		Heating capacity for resistive wire	<232°C	F0	26
		Accessibility of power switch	Product exterior	F0	27
FC5: Must respect the environment		Paint	Biodegradable	F0	28
FC6: Resisting External Constraints		Impact force	>50N	F1	29
FC7: Power Supply		Energy	Electric	F1	30
		Energy storage	Battery	F1	31
		Technology	Lithium	F1	32
		System status	Not muffled, loaded, transported, stored in fireproof	F0	33
FC8: Must not exceed a certain cost price		Cost price	<300 euros	F2	34
FC9: The product must be aesthetically pleasing		Proportion of people attracted by the product's aesthetics	80%	F2	35

Figure 14

Below, you'll find the specifications, which present the different functions in a hierarchical manner, along with criteria that detail the customer's different expectations, as well as the criteria that contribute to the proper functioning of the system.

7) Technical Specifications

b. FAST diagram

The FAST diagram is based on the functional specification sheet that was developed earlier, which defines the product's needs, objectives and requirements in terms of functions and performance. The aim of the FAST diagram is to define and represent the technical solutions chosen as being the most appropriate in relation to the project's need and the technical characteristics of the product as previously defined in the functional specification sheet.

The FAST diagram is divided into 4 stages.

The first is the service function to be satisfied. The second is an inventory of the main technical functions derived from the service function. The third stage lists the sub-functions associated with the various main technical functions, and the final stage represents all the technical solutions required to satisfy all the above-mentioned technical functions and sub-functions.

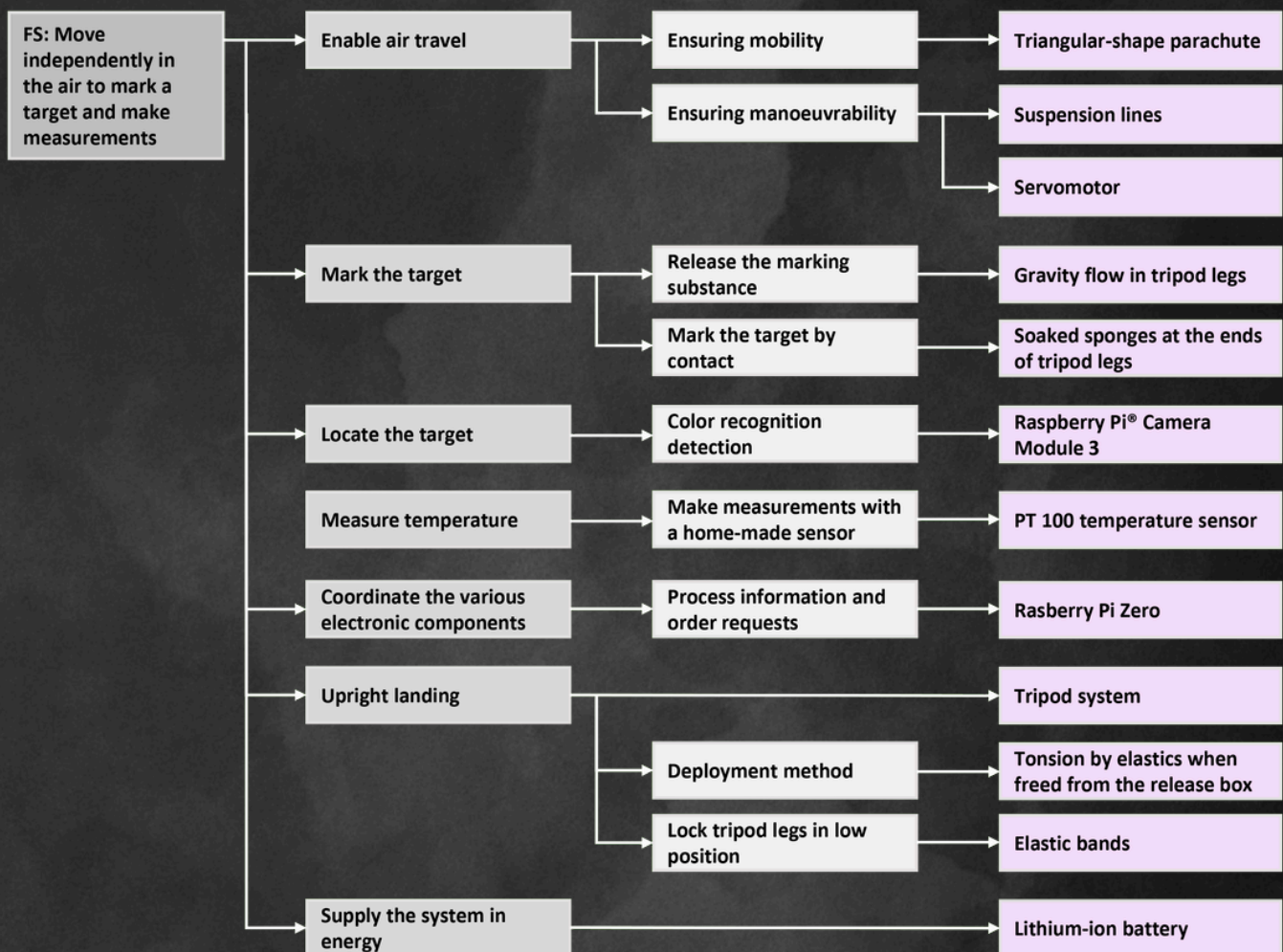


Figure 15

Above is the FAST diagram, specific to the service function that were entered above. This diagram shows the technical solutions chosen by the team.

8) Resolution Process

a. Overall Resolution Process

Here's a structured overview of the resolution process and the solutions that were implemented for various aspects of the CanSat, to meet the client's requirements, following the FAST diagram that was accomplished

Triangle Shaped Parachute:

- **Initial Idea:** Originally it was planned to use a hemispherical parachute, however during the year, it was reconsidered to a triangular shaped parachute.
- **Revised Approach:** Indeed the triangular shaped parachute is extremely stable during the flight.

Suspension lines :

- **Initial State :** The suspension lines are attached both to the CanSat and the Servomotor. The Servomotor pulls on those lines in order to control the CanSat during the flight.
- **Revised approach :** Despite the calculation made on the length of the suspension lines, they were too long and the parachute could not deploy effectively. 10cm were removed, so the current length is of 85cm. Moreover an anti-knot device has been added.

Servomotor :

- **Selection :** The servomotor is the most important of the CanSat. It is attached on the top of the inner body, as well pulling the hangers of the CanSat to be able to control the flight direction.

Gravity flow in tripod:

- **Selection:** The painting system consists of paint stored in the tripods, which fill up sponges, so when the CanSat lands on the target, the target is painted.

Soaked sponges at the end of tripod legs :

- **Selection:** Sponges are attached at the end of the tripods in order to paint the target. It is a cheap, easy, and eco-friendly way to paint the target without using servomotors.

Raspberry PI Camera Module 3:

- **Initial Idea:** The initial idea concerning the camera was to use a HuskyLens since it has all the image recognition parts integrated into it, nonetheless it didn't fit inside the CanSat. Moreover, it was decided to use a camera instead of a GPS recognition captor since no CanSat at ICAM ever touched the target, so a new solution was needed.
- **Revised Approach:** Since the HuskyLens didn't fit, the choice was to use a Raspberry Pi Camera Module 3, and then code everything from it (image recognition + commands to the servomotor)

Tripod System:

- **Choice:** Concerning the tripod system, it was decided to use a three-leg system since it is the most stable configuration.

Tension by elastics when freed from the release box:

- **Selection:** It was considered very easy to use elastics for the landing system since it is the most convenient solution as well as being very efficient. Nevertheless every five landings, the elastics need to be changed, for greater efficiency.

Li-ion Batteries :

- **Selection:** It was decided to use this type of battery since it is the most popular and cheap type of battery as well as delivers enough energy to be able to last for an hour (>50 minutes which is required in the technical specifications)

DIY Sensor :

- **Selection :** It was decided to make an home-made temperature sensor. It was decided to make this type instead of any other type since it was something easy to make and a member of the team had already made one in his previous projects. For the captor it was decided to use a PT 100 sensor

9) Components

a. Bill of Material

The next step in the advancement of the project was to find all the components that were needed. Nevertheless even if the majority of the used components were determined at an early stage of the project, small components such as the support pins were added during the year. The following Bill Of Materials represents all the components that are in the CanSat to this day. This BOM is helpful to make all the component orders needed.

BoM ICAMeteor CanSat						
name of part	part number	description	quantity	unit	unit cost	cost
Body	P1	body of cansat	1	1	8.3€	8.3€
Lid	P2	to close the up of cansat	1	1	0.83€	0.83€
Legs	P3-1	to land upright	3	1	0.12€	0.36€
Legs pin	P3-2	lock the legs	3	1	0.04€	0.12€
Rack	P4-1	skeleton for hardware	1	1	2€	2e
Rack Legs	P4-2	fix the rack inside	4	1	0.07€	0.28€
Rack pin	P4-3	lock in place the rack legs	4	1	0.02€	0.08€
Raspberry PIO	H1	hardware	1	1	14.19€	14.19€
Raspberry Lens	H2	to acquire the target	1	1	44.99€	44.99€
Wiring	HW	to acquire the target	1	1	0€	0€
Servomotors	H3	Move the parachute	2	1	25.50€	51 €
fixing for servomotors	H3-2	fix them in cansat	4	1	0€	0€
parachute	P5-1	Slow down the fall	1.6 m2	m2	24.49€	32€
fixing for parachute	P5-2	lock it with servo	2	1	0€	0€
rope for parachute	P5-3	keep chute in place	4	m	15€	15€
batteries	P5-4	power the raspberry	1	1	6.49€	19.48€
fixing for batteries	P5-5	put the batteries inside	1	1	5.41€	16.23€
recyclable sponge	P5-6	used to paint target	3x	1	3.50€	3.50€
stencil	P5-7	decote the Cansat	1	1	20€	20€
support pin	P5-8	prevent cansat from bending inside	1	1	13€/60cm	1.70€
PT100	P5-9	to analyse temperature outside	1	1	5.99€	11.99€
Elastic band	P6-1	opening the legs	3	1	0.68€	6.80€
switch on/off	H4	turn on or off the Cansat	1	1	0€	0€

Figure 16

The components that are marked as 0 euros in costing, were borrowed to ICAM, with the approval of the technical expert (in this case, Mr.KAZEMI).

Moreover, the decision to take the cost of the used PETG for the 3D impression into account was made, since it is a cost that is non-negligeable.

At the end the CanSat costs : **273,83 euros** which is under the 300 euros that were allocated for the project.

9) Components

Here is a list of the components used in the CanSat, and their role :

- CanSat Body:
 - The CanSat body is the main structure housing all project components. It is designed to be sturdy, lightweight, and shock-resistant to protect internal equipment during descent and landing.
- Lid:
 - The lid is the top part of the CanSat that closes the assembly. It is designed to open easily for access to internal components and securely closed to prevent air leaks during descent.
- Legs:
 - Legs are elements that allow the CanSat to land upright during descent. They should be foldable to minimize space during ascent. They also serve as the painting system.
- Legs Pin:
 - Leg pins are used to attach the legs to the CanSat, as well as making available the rotation for them to open easily.
- Rack:
 - Description: The rack is an internal structure that holds electronic components such as the Raspberry Pi, servomotors, and the camera in place. It ensures that the components are secure and protected.
- Rack Legs:
 - Description: Rack legs are used to attach the rack inside the CanSat, keeping it in place to prevent movement during descent.
- Rack Pin:
 - The rack pin is used to lock the rack in position inside the CanSat, preventing accidental movement.
- Raspberry Pi 0:
 - The Raspberry Pi 0 is an embedded microcomputer used to run programs and collect data aboard the CanSat. It is the brain of the CanSat that acquires data from the lens and the PT100 as well as giving orders to the servomotor. It is attached to the rack
- Raspberry Lens:
 - The lens is used with the Raspberry Pi to capture images during the flight. It should be correctly installed to ensure high-quality images. It is attached to the rack.

9) Components

- Wiring:
 - Wiring is essential to connect all electronic components of the CanSat, including sensors, servomotors, and the Raspberry Pi. It should be carefully planned and organized to avoid connection issues.
- Servomotor:
 - The servomotor is used to control the parachute's movements and maintain the CanSat in an upright position during landing. It is attached to the rack
- Fixing for Servomotors:
 - Fixings for servomotors are used to secure the servomotor in place inside the CanSat, ensuring it's proper operation.
- Parachute:
 - The parachute is used to slow down the CanSat's descent and ensure a smooth landing. It is attached to the CanSat and the servomotor with rope. The parachute totally is hand-sewed.
- Fixing for Parachute:
 - Fixings for the parachute are used to securely attach the parachute to the CanSat, preventing premature detachment.
- Rope for Parachute:
 - The rope for the parachute is used to keep the parachute attached to the CanSat. It should be strong enough to support the CanSat's weight during descent as well as the tension forces exerted by the servomotor.
- Batteries
 - It is the main source power of the CanSat. It has to be small enough to fit inside the CanSat, but powerful enough to last the duration of the flight. It is fixed on the battery fixing. The battery is a 223 CR-P2 Duracell Lithium 6V
- Fixing for batteries
 - It is attached to the rack, and permits the fiwing of the battery.
- Recyclable sponges
 - It is attacahed to the legs (paint storage) and is our paint system. It paints the target when the CanSat lands.
- Stencil
 - It is what is used to paint the exterior of the CanSat.
- Support pin
 - It is a security inside the CanSat so it can resists to the forces applied during the flight.

- PT100
 - It is our homemade temperature sensor. It is linked to the card.
- Elastic band
 - It is the leg deployment of the CanSat
- Switch on-off
 - It enables to switch on and off the CanSat, so the battery can be used only when needed.

To order all the components, a template that was given by the supervisor was used to make them. We used it whenever there was an order to be made.

This is an example of a circuit made with the components listed above, here the circuit for the temperature sensor, and the proofs of the functioning.

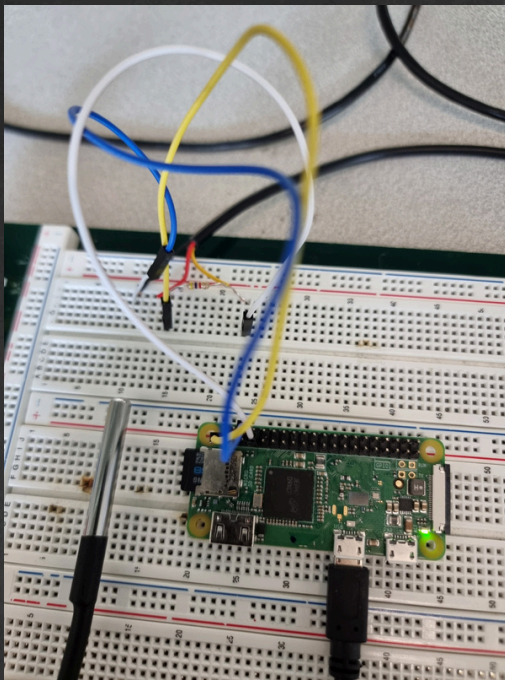


Figure 17

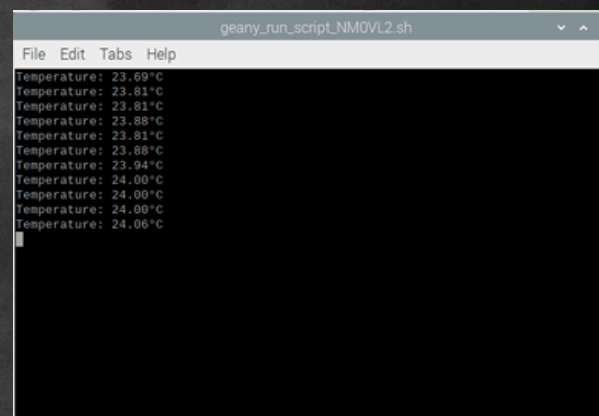


Figure 18

10) Electronics

a) Wiring Plan

The initial phase of creating the wiring plan involved using TinkerCAD, an online 3D modeling software. Initially, two designs were proposed: one with a single servomotor and another with two servomotors .

The secondary mission involved constructing a homemade temperature sensor.

After consultations with the client, the design featuring a single servomotor was chosen. This decision was driven by the benefits of reduced space and weight, which are critical given the limited capacity of the CanSat. Additionally, programming is simplified without the need for servo-control.

Phases:

1- Ground level

Limited performance (t= 20min)

2- Rise up to release altitude

Limited performance (t= 10min)

3- Controlled descent

Maximum demand (t= 5min)

4- Ground level

Limited performance (t= 20min)

Limited performance: (830 mAh)

- Homemade temperature sensor
- Raspberry Pi Zero

Maximum demand: (440 mAh)

- Homemade temperature sensor
- Raspberry Pi Zero (Imax)
- Raspberry Pi Cam
- Servo-motor

Overall consumption: < 1270 mAh



Figure 19

A Duracell 223 CR-P2 6V lithium battery with a capacity of 1400 mAh was chosen to power the system. This choice was motivated by its greater capacity than the estimated requirements, calculated by taking into account the maximum operating conditions of each component. In addition, this battery was chosen for its compact size and advantageous cost.

A set of resistors has been integrated to guarantee an adequate power supply to each of the components.

11) Parachute

The aim of the parachute is to slow the fall of the Cansat after the launch.

The device weights around 450g, from a 150m altitude of launching, it will reach a maximum speed of 3.5m/s. Then it needs a proper slowing. To insure a safe landing, a parachute of 1,2m² is sufficient to reduce the falling speed.

According to the previous decisional matrix page 14, the shape of the parachute chosen is a triangle shape.

a) Prototype

Before crafting the final parachute, two prototypes were made to confirm the decision of the shape , one hemispherical and one triangular.

Trashbag was use for these prototypes not to use real fabric in case of problems.

To simulate the body and weight of the cansat, a bottle of 500ml of water was attached to the hangers.

The tests were convincing because the triangular shape slowed down the fall as expected and had less pendulum momentum during the fall.

b) Fabrication

The choice of the fabric was quite simple, nylon is a material renowned for its lightness and strength and is already use in the industry for the manufacture of sails for parrachutes, parrapentes and other aerial applications, moreover it is cheap and easy to buy.

The making of the parachute includes 6 steps ; calculation of the size of the 3 sails to have a final surface of 1.2m², marking the dimension on the fabric, cutting following the lines, making hems, sewing the sails together, fix the hangers. Example of a sail :

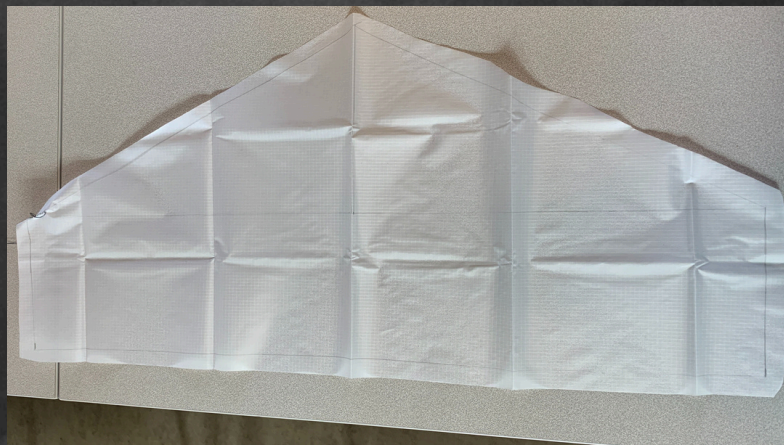


Figure 20

The hems are made to prevent the fabric from fraying. At the end, the parachute obtained looks like this :



Figure 21

For a good deployment after the launching, the parachute needs to be folded in a way which gives the quickest deployment. The three sails are laid flat (cf. picture 11) and superimposed. Each in turn they are folded on themselves following an accordion method. The 3 sails then folded are gathered into a single rectangular section as shown on the left picture below. The folding in an accordion way is the simplest and more efficient for a good air penetration in the sails. Indeed, the parachute flaps are oriented downwards, as shown in the picture below.

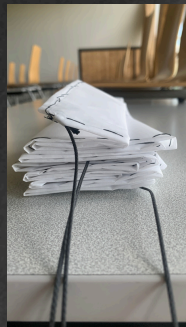


Figure 22

12) Mechanical Conception

a. 3D Conception

In order to make the conception started, the maximum dimensions allowed were considered, i.e. 20 cm in length and 8 cm in diameter. The CanSat is divided in three parts
An outer shell to house the tripod system that deploys on launch. This shell can be seen below.

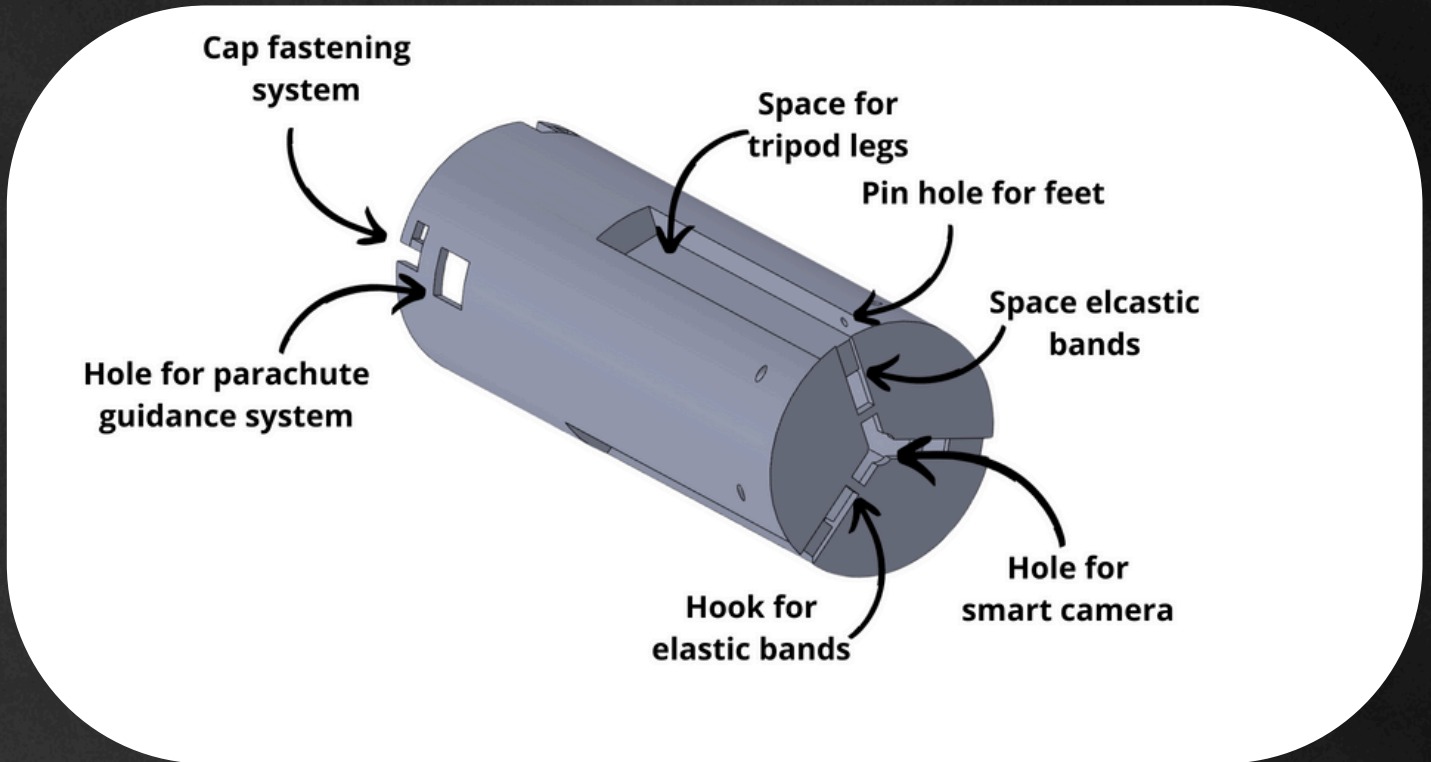


Figure 23

A system of elastic bands is used to deploy the feet when the launcher tube is extended. This system consists of hooks integrated into the tripods (visible below) and hooks integrated into the lower part of the CANSAT (visible above). The whole system will be connected by elastics measuring less than 2,5cm in diameter at rest. Inside the pods will be the storage for the paint with sponges attached to it.

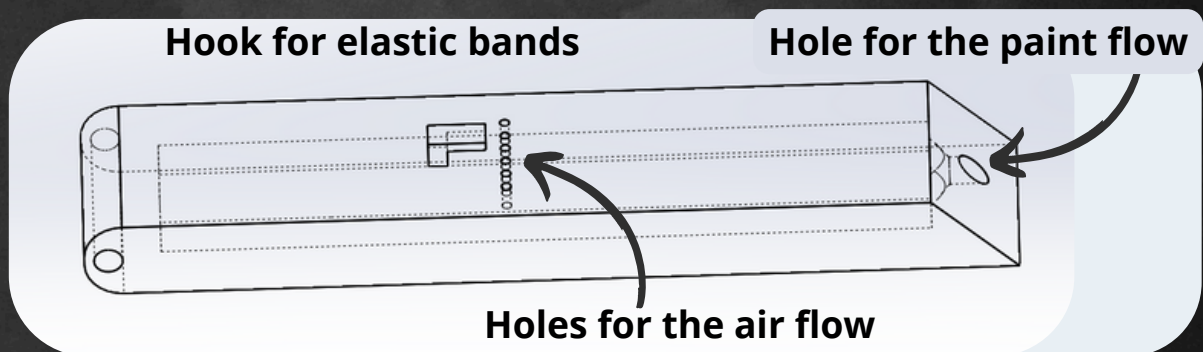


Figure 24

12) Mechanical Conception

a. Inner design

The complete system can be seen in the final design.

Next, the attention was turned to the cap, which is simply fixed by a system of slots.

The first prototype revealed that , featuring only 2 rounded slots, was ineffective as the cap rotated (by the horizontal axis).

Therefore, a 4-slot square system was chosen to mitigate this issue. This has been tested with the second prototype.

The hole on the top is destined to the temperature sensor.

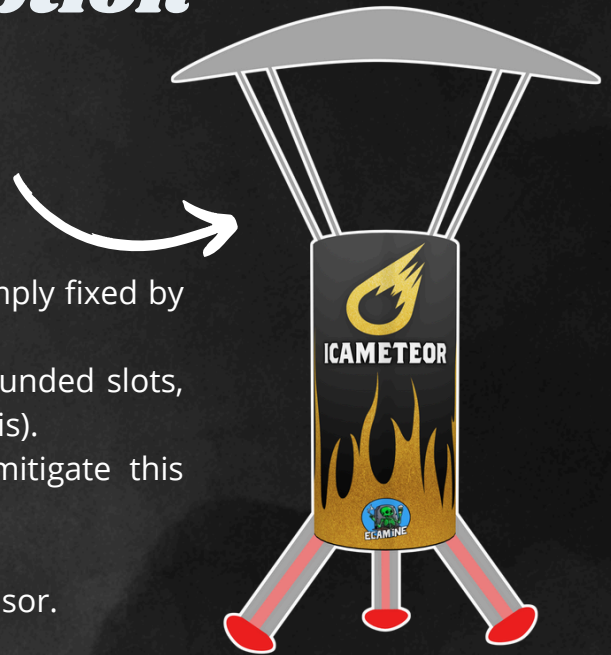


Figure 25

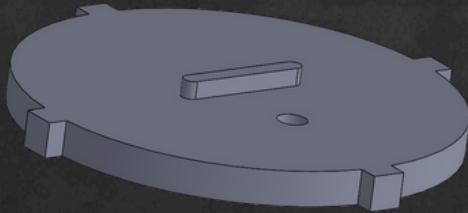


Figure 26

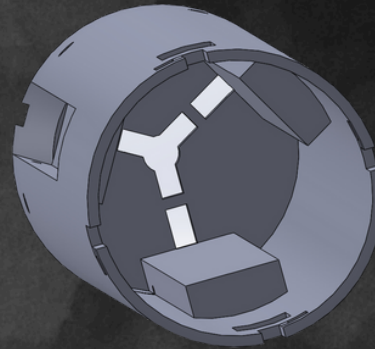


Figure 27

For the internal design (skeleton), a design with a central plate that can be fixed from either side was chosen, as this space is preferred for the electronics.

The hole in the skeleton is reserved for the battery and is at the bottom of the CanSat, in order to stabilize the CanSat during the flight.

The lower disk will be utilized for attaching heavy components such as batteries as well as the camera. The upper disk will be designated for mounting the PT100 temperature sensor and the servomotor responsible for steering the CANSAT.

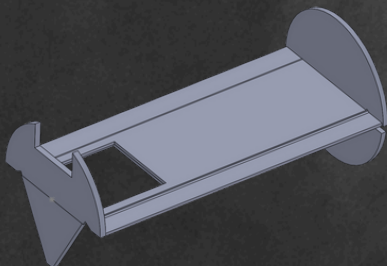


Figure 28

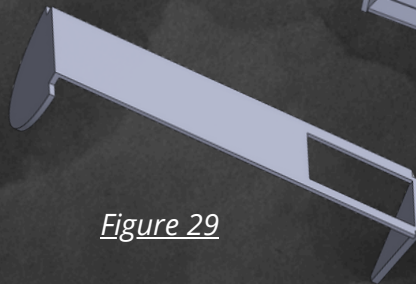


Figure 29

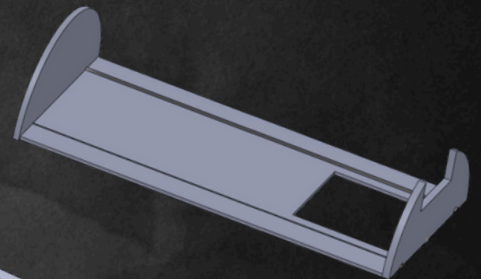


Figure 30

12) Mechanical Conception



b. 3D Impression

An ecological approach was taken in our design, aiming to minimize the utilization of PETG for the printing. To achieve this, the "skeleton" part was divided into two sections in order to minimize the use of supports during the printing. Those two parts are then attached together

We can see below a figure of the interior wiring

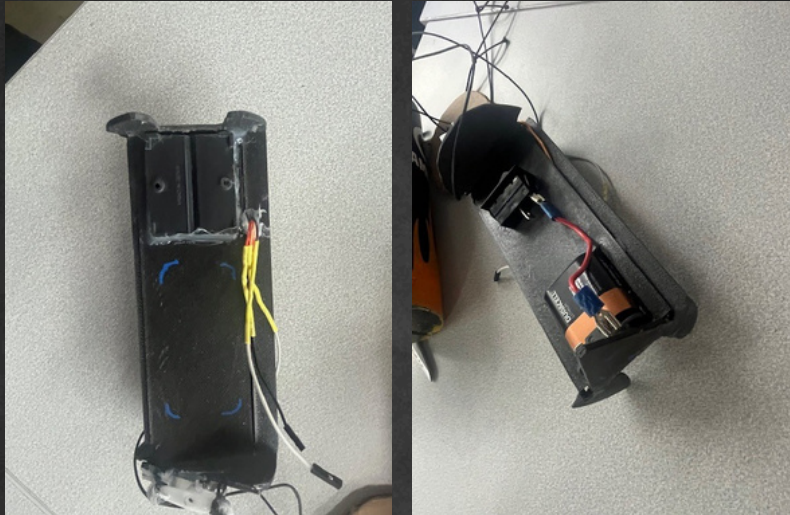


Figure 31

Note that the card as well as the camera are not yet installed, since some details had to be adjusted on the code, but when it is achieved, they will be attached to the inner design at the location determined by the blue marks

c. Testing

Testing of the system and solutions is mandatory to ensure quality for the client. Given that the CanSat will be dropped and will hit the ground at a speed of around 3m/s (thanks to the parachute), drop tests were conducted to assure the structural strength of the project. These tests were conducted in two parts, with the first part performed on Inventor Nastran (software enabling drop test simulations) where all conditions were taken into account (real weight of the CanSat = 450g with paint and parachute, as well as having a single contact point on the floor). This test was applied under the condition where there is a failure of the tripod system.

The simulation on Nastran yielded the following result:

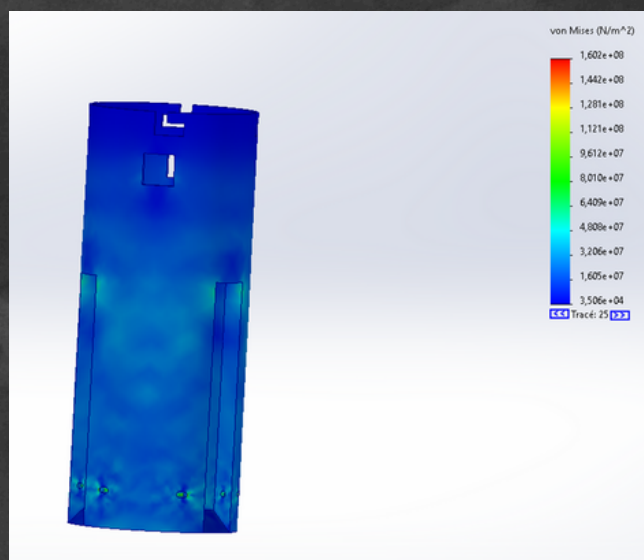


Figure 32

12) Mechanical Conception

According to our tests, the cansat can withstand falls at a speed of 4m/s. This has been verified in our tests, indeed the fall time is of 3,3m/s (outdoor test).

In our simulations, it broke with forces in excess of $8.5 \times 10^7 \text{ N/m}^2$; this threshold was exceeded in certain areas of the CanSat, identified as the 'greener zones'.

With our first prototype, we carried out drop tests at 1m, 2m, 5m and 10m.

During these tests, the CanSat suffered no damage at the first 2 heights tested, but began to crack at 5m and at 10m, as can be seen in the following photo.

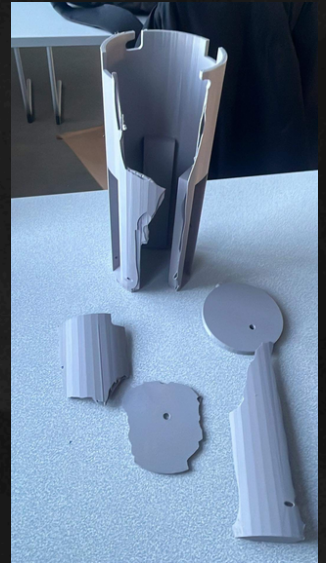


Figure 33

The solution for strengthening our hull was to switch to Honey comb filling.

Our finished product was tested with this and withstood a 10m drop on grass without a parachute, as well as a 60 meters droptest, with the parachute.

D. Assembly

First of all the outer shell is completely independent of the inner shell which gives easy access to the internal components for easy access to the system.

During assembly, few alterations were made in the fabrication laboratory to ensure that the parts fit together perfectly indeed, one hole was drilled in the hull in order to fix the first parachute hanger while the two others are attached to the servomotor.

A pin is here to lock the interior design and solidify the structure

After a first try to fit the servomotor into the interior design, small modification were made to the existing interior design by hand in order to not have to reprint the inner design.

The main modification was to cut off a part of the top disk to make room for the servomotor to operate.

Finally a poka yoke was needed for the interior and exterior design, using a paint mark, to ensure a perfect assembly and maximum operability.

The poka yoke for the cap can be seen in the photo circled in orange.

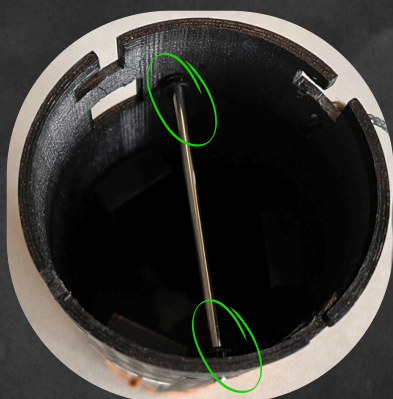


Figure 34



Figure 35

13) Design

The initial exterior design, visible in figure 33, guided the creation of the stencil shown in figure 34. This stencil was crafted from vinyl material using the Cricut Maker III machine, available in the ICAM fabrication laboratory.

Before stencil application, the CanSat's surface was prepared by sanding and painting it black.

The paint, borrowed at no cost from the school, was then applied to complete the design. The final appearance of the CanSat is depicted in figure 35.



Figure 36

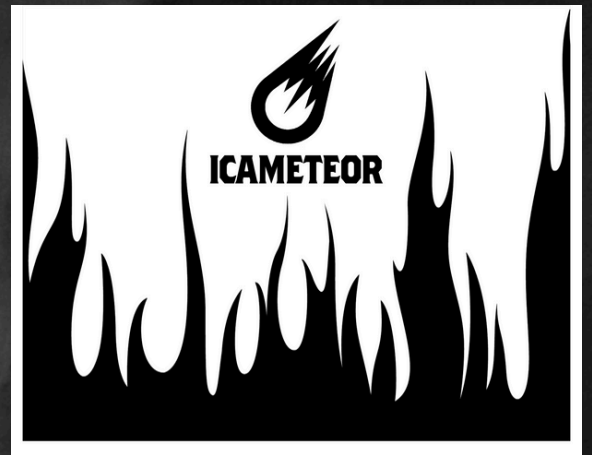


Figure 37



Figure 38

13) Code

a. Coding objectives

The project's programming is divided into 5 main steps, providing a general overview of the project's objectives.

Detecting the target

Intelligent color and shape detection

Navigation and control algorithms

Defining the optimal trajectory and all the different actions required in flight

Control logic

Real-time adjustment of servomotor position based on sensor data to maintain trajectory

Temperature measurements

Real-time temperature measurement thanks to the PT100 temperature sensor

Optimizing energy consumption

Defining the different flight phases and put unnecessary actions on "sleep mode"

All these parts are grouped together in a python file which is stored on our raspberry pi zero's sd card, allowing the cansat to operate autonomously while performing all these tasks simultaneously.

b. Choice of the language

Python programming language is used for this project, coupled with the OpenCV programming library. This combination represents several key advantages in the part dedicated to real-time target detection.

- Python is a language easy to learn and read, with extensive community support, as we could see with many resources available on the internet of similar applications that we plan to apply in our project.
- OpenCV is a powerful library vision computer, that offers advanced features for the treatment of images and videos in real-time, essential for the follow-up of forms, which will simplify considerably the writing and global understanding of our code.



13) Code

c. What does the program do :

1. Target detection

- Frame-by-frame image acquisition from the camera
- Isolation of white color in the image
- Detection of circular shapes if they are white
- Track white circular objects movement in images
- Determine the direction of object movement
- Storage of output on sd card

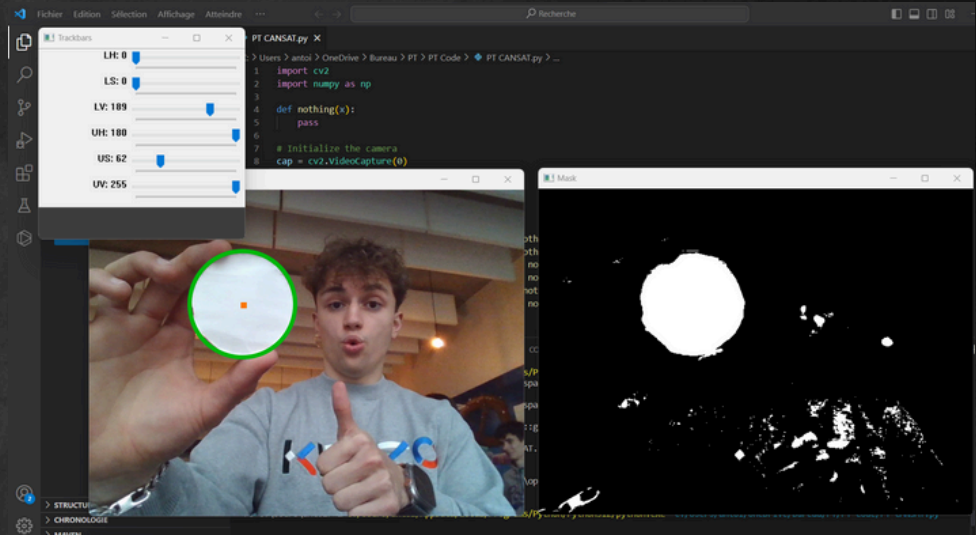


Figure 40

2. Navigation algorithm

- Split the frame into 4 quadrants
- Until the target is spotted, the cansat circles in the air to locate it.
- If the target deviates from the center, the servomotor makes a correction to bring it back to the center according to its position.

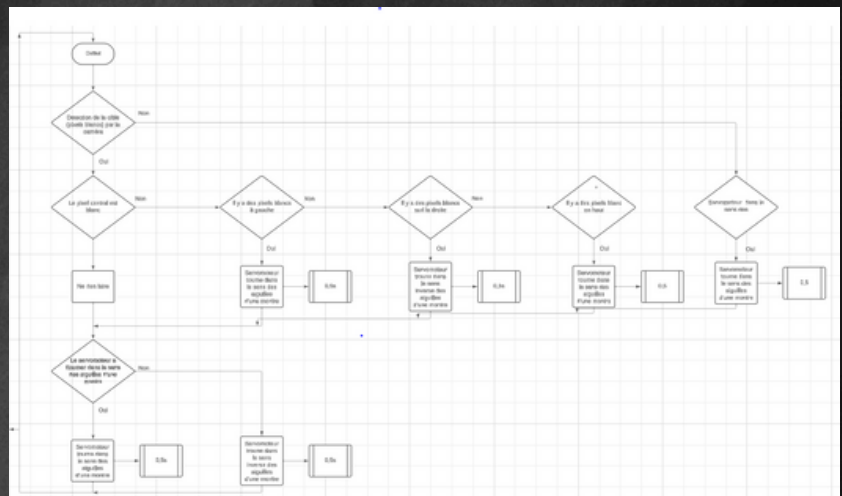


Figure 41

3. Temperature measurements

- The temperature sensor takes a temperature reading every 10 seconds
- Storage of output on sd card

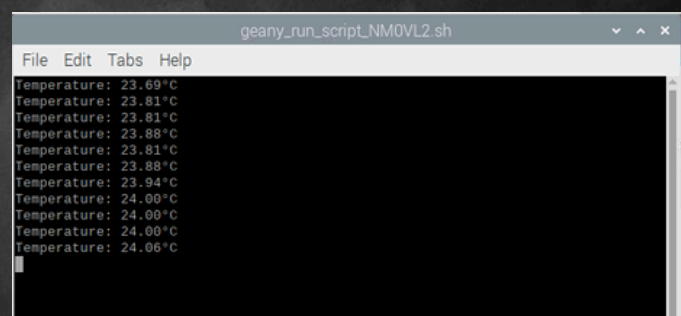


Figure 39

*The complete system code can be found in the appendix on page 49.

14) Results

a. Gap between results and objectives

While working on the project, there were high expectations that the Computer-Aided Design (CAD) would function efficiently from the outset because everything was fitting perfect and was printed the way it was designed. However, it became apparent that the scope of challenges arising during the process had been underestimated. The CanSat printing had to be redone several times as components were incrementally added.

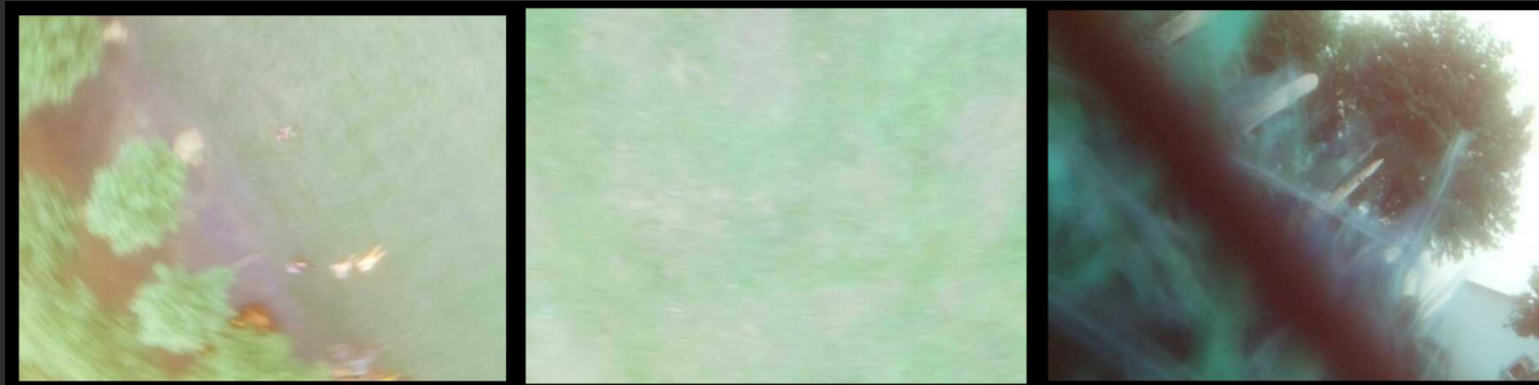
The landing mechanism of the CanSat posed significant challenges. Testing revealed unexpected weaknesses in its durability especially in the pin for the legs that were too weak. Consequently, modifications were necessary, including adding bands to reinforce the legs and changing the printing settings. This led to a great improvement for the overall stability and durability.

Furthermore, adjustments were made to the hardware configuration. Notably, the lens system was upgraded from a Huskylens to a RaspberryLens to better meet project requirements and enhance CanSat performance. The code in itself is meeting the requirements but is not as efficient as what was plan. In fact, several factors like wind and balancement made it more difficult than expected.

15) Result of the flight

The flight was almost a complete success, we didn't have any problem with the parachute but we didn't reach the target. Even if the motor was trying to control the flight, there was too much wind to reach the target but we landed straight. The target recognition system was probably not reliable enough, although it was operational despite the fact that it alternated with the GPS in case of loss of image from the camera. If our team were to make another flight, we'd modify the parachute slightly to give it a chimney to stabilize the descent. We'd also be looking for other, more reliable components for better target recognition.

Here some picture of our flight and the result return by our GPS



Pictures of the flight



Target recognition



Data from GPS

2024-07-09	10:45:50,	0.0,	0.0,	359.9447441053467
2024-07-09	10:46:28,	0.0,	0.0,	359.9447441053467
2024-07-09	10:46:31,	0.0,	0.0,	359.9447441053467
2024-07-09	10:46:34,	0.0,	0.0,	359.9447441053467
2024-07-09	10:46:39,	0.0,	0.0,	359.9447441053467
2024-07-09	10:48:32,	0.0,	0.0,	359.9447441053467
2024-07-09	10:48:35,	0.0,	0.0,	359.9447441053467
2024-07-09	10:48:39,	0.0,	0.0,	359.9447441053467
2024-07-09	10:48:42,	0.0,	0.0,	359.9447441053467
2024-07-09	10:48:45,	0.0,	0.0,	359.9447441053467
2024-07-09	10:48:48,	0.0,	0.0,	359.9447441053467
2024-07-09	10:48:51,	0.0,	0.0,	359.9447441053467
2024-07-09	10:48:54,	0.0,	0.0,	359.9447441053467
2024-07-09	10:48:58,	0.0,	0.0,	359.9447441053467
2024-07-09	10:49:02,	0.0,	0.0,	359.9447441053467
2024-07-09	10:49:27,	0.0,	0.0,	359.9447441053467
2024-07-09	10:50:17,	0.0,	0.0,	359.9447441053467
2024-07-09	12:08:38,	0.0,	0.0,	359.9447441053467

Timestamp,	Latitude,	Longitude,	Target bearing
2024-07-08 11:19:15,	43.242195333,	-0.039328333,	60.81295059981858
2024-07-08 11:19:17,	43.242192333,	-0.039332833,	60.812946024114694
2024-07-08 11:19:20,	43.242195333,	-0.039328833,	60.812950517110494
2024-07-08 11:19:23,	43.2422015,	-0.039327167,	60.81295866863593
2024-07-08 11:19:26,	43.242219667,	-0.039330167,	60.81298137366525
2024-07-08 11:19:29,	43.242224167,	-0.039337167,	60.81298596274769
2024-07-08 11:19:33,	43.242222167,	-0.03933,	60.81298459406884



16) Conclusion

a. Evolution of the sprints

The CanSat project evolved through multiple development sprints, each addressing various components and design improvements. The initial setup began with the establishment of a backlog, the first hardware design, and the selection of materials, systems for paint dropping, landing mechanisms and parachute designs.

The first sprint focused on the establishment of the technical specifications and all the tools linked to it, as well as defining the firsts prototypes of each part of the CanSat.

During the second sprint, new pod designs and exterior aesthetics were introduced, along with state-of-the-art camera tracking, selection of electrical components, and calculations for parachute sizing based on desired fall speed. Initial coding concepts were also developed.

The third sprint included theoretical and experimental dropping tests, further refinements to the CanSat design, creation of the first Bill of Materials, enhancements in target detection, and development of wiring plans.

The fourth sprint saw the completion of a tripod, the design and testing of the first parachute prototype in two versions, and advancements in the coding algorithm.

In the fifth sprint, the team crafted a temperature sensor, cut and sewed the first parachute, and finalized the target detection systems.

The sixth sprint focused on folding the parachute and further adjustments to the internal design to enhance ergonomics and space management. Throughout these stages, continual updates and redesigns were applied, including new filling materials for 3D printing to increase resistance, and adjustments to internal layouts to accommodate various components like the camera.

16) Conclusion

b. Conclusion

The project achieved its initial objectives, yet there are opportunities for further improvements. For instance, the internal design was modified rather than reprinted, which could have resulted in a cleaner final product if it had been reprinted anew. Additionally, components inside the CanSat were secured with glue; using fasteners like screws might have provided a tidier appearance and more durable assembly.

A significant area for enhancement is the flight algorithm. Although it has been updated following one flight test, achieving a fully accurate flight algorithm requires more extensive testing. Furthermore, incorporating a self-learning AI into the CanSat could significantly enhance its performance and adaptability in future deployments.



Figure 42



Appendix 1 - User's Stories

Initiative	Epic	As a	I/We want to	So that	Sprint
Make a Cansat	GPS Localisation of the landing zone	User	To have a code that can process GPS coordinates	The system can then navigate using those coordinates	3
		Team	To have the most suitable system for the GPS	It can process the navigation parameters efficiently	2
		Team	Make a state of art of possible solutions	I can choose the best possible solution	1
		User	Choose our final solution	Our project can advance	1
		Team	The satellite to navigate the most efficiently	My landing on the target expectations can be fulfilled in the best way possible	3
	Navigate autonomously to the area using GPS coordinates	Team	To understand clearly what is asked by our client	We find a clear way to land on target	1
		User	To dimensionate properly the flying system	The satellite does not crash during flight	1
		Team	To get the easiest design for our electronics	We mitigate the risks of errors in the electronics	2
			To get the easiest code design	We mitigate the risks of errors in the code	3
		User	To have the most efficient materials in my price range	The flying system is as efficient as possible	2
	Mark a target on the ground as much and as centered as possible		The flying system to take the least place possible	There is enough space disponible for other components in the 1L volume	2
		User	The satellite to be autonomous	It can navigate to a defined landing site	5
			To receive a prototype	The chosen solutions can be assessed	4
			To have the final solution produced	The satellite can be finalized	5
			To have an easy way to mark the target	I do not struggle with possible errors	1
	Be able to change anytime recorded GPS coordinates in the satellite		To have the best material for the painting system	I can have the material that is best suited to marking the target in accordance with all my expectations	1
		User	Have an efficient way to store the paint	The satellite won't be too heavy because of the paint	2
			Make a first prototype	The chosen solutions can be assessed	2
			To have the best working painting system	I can mark as much as possible the target	4
		User	Make a state of art of possible solutions	I can choose the best possible solution	2
	Have a flag brandished after landing	Team	Choose our final solution	Our project can advance	1
		User	An easy and manual design	I can change easily the GPS location	2
			To be able code the satellite to change easily the GPS locations	During the launch I do not lose time programming the satellite for the flight	2
		Team	Make a state of art of possible solutions	I can choose the best possible solution	3
		Team	Choose our final solution	Our project can advance	1
Green plan	Create a captor from scratch	User	Have my colors brandished	My satellite can be recognized among others when flying	2
		User	Have my colors brandished when the satellite is landed	My satellite can be recognized among others when landed	3
		Team	Have a state of art of the captor	I can choose the best possible solution	5
			Choose the adapted solution	Our project can advance	1
		User	A captor made from scratch	I can insert it in my satellite	2
	The satellite shall land upright		Have a first prototype	The chosen solution can be assessed	2
			To insert my captor in the PCB	My captor can be fully integrated in the satellite	3
		User	A design that can make the satellite land	I can visualize and approve the design	4
			To have the best material for the landing system	I can have the material that is best suited to landing on the ground in accordance with all my expectations	2
		User	The landing system to not be destroyed at each landing	I do not have to replace each time the landing system	1
	Environmental Aspect		Have a first prototype	The chosen solutions can be assessed	2
			Finalize the product	The satellite can land on the ground without breaking	5
		User	Not have a lithium battery	It is respectful of the environment	1
			Have a reusable paint	When marking the target we do not damage the surrounding eco-system	1
		User	Have reusable materials	I can reuse the materials of the satellite if it is not needed anymore	2
Green plan	Risk Analysis	User	Get a risk analysis	So I can assess and be precautionous about the different features of the satellite	2
	Economic Aspect	User	The price of the satellite shall not go over 300 euros	I have an interesting return on investment	3
	Social Aspect	User	Have assurance that the satellite is ethically managed	There is transparency and accountability in the project's social implications	1

Appendix 2 - Backlog

Sprint 1 Backlog ☆ 📍 000 Tableau ▾ 🔍 ⚡ ⚙️ Filtres RL AM B EK +1

À faire

+ Ajouter une carte

En cours

+ Ajouter une carte

Review

+ Ajouter une carte

Terminé

Calculation sheet

🕒 7 oct. 2023 - 13 oct. 2023 RL

Meetings

Sprint 1 Meeting

🕒 11 oct. 2023 - 16 oct. 2023 1 B

Design

Components Research

👍 1 🕒 13 oct. 2023 1 JC

Présentation

Diaporama

👍 1 🕒 9 oct. 2023 - 14 oct. 2023

Define team roles

👍 1 🕒 11 sept. 2023 B

+ Ajouter une carte

Terminé



Technical Specification

Functional Technical Specifications

🕒 26 sept. 2023 - 13 oct. 2023 2 AM

Technical Specification

Project Comprehension/Study

🕒 13 sept. 2023 1

+ Ajouter une carte

Terminé

Green Plan

Risk Analysis

🕒 10 oct. 2023 - 13 oct. 2023 EK

Technical Specification

Resume of the project specification

🕒 13 sept. 2023 1 B

Technical Specification

User Stories

🕒 8 oct. 2023 - 10 oct. 2023 1 B

Technical Specification

State of art

👍 1 🕒 17 sept. 2023 - 26 sept. 2023

+ Ajouter une carte

Terminé

Management

Backlog/Kanban

🕒 11 sept. 2023 B

Management

Gantt Diagram

🕒 11 sept. 2023 - 15 sept. 2023 B

Management

Burndown chart

🕒 18 sept. 2023 B

Technical Specification

SADT Diagram

🕒 14 sept. 2023 - 18 sept. 2023 1 AM

+ Ajouter une carte

Terminé

Technical Specification

FAST Diagram

🕒 14 sept. 2023 - 18 sept. 2023 1 RL

Design

1st Interior Design

👍 1 🕒 27 sept. 2023 - 13 oct. 2023 EK

Technical Specification

Diagramme Bête à corne

👍 1 🕒 14 sept. 2023 - 18 sept. 2023 1 JC

+ Ajouter une carte

Terminé

Technical Specification

Diagramme pieuvre

🕒 14 sept. 2023 - 18 sept. 2023 1 AM

Technical Specification

SWOT Analysis

🕒 14 sept. 2023 - 18 sept. 2023 1 EK

Technical Specification

Vérification des rendus

👍 1 🕒 25 sept. 2023 B

Design

1st Exterior Design

🕒 26 sept. 2023 - 9 oct. 2023

Technical Specification

Decisional Matrix

👍 1 🕒 2 oct. 2023 - 9 oct. 2023 1 EK

Meetings

Client Meeting 1st Meeting

🕒 18 sept. 2023 1 B

Meetings

+ Ajouter une carte

Terminé

Meetings

Client 2nd Meeting

🕒 25 sept. 2023 1 B

Meetings

Client Meeting 3rd Meeting

🕒 9 oct. 2023 1 B

Meetings

Supervisor Meeting 1st Meeting

🕒 18 sept. 2023 1 B

Meetings

Supervisor Meeting 2nd Meeting

🕒 2 oct. 2023 1 B

Meetings

Meeting with Thibault Abalain

🕒 22 sept. 2023 1 B

Appendix 2 - Backlog

Sprint 2 Backlog ☆ 👤 Tableau ▾ 🔍 ⚡ ⚙️ Filtres RL AM B EK +1

To Do ...
+ Ajouter une carte 📄

Doing ...
+ Ajouter une carte 📄

Review ...
+ Ajouter une carte 📄

Done ...

- Code**
Start coding
🕒 21 oct. 2023 - 13 nov. 2023 AM
- Conception**
Test résistance canette
🔥 1 🕒 10 nov. 2023
- Sprint 2**
Sprint 1 Preparation
🕒 10 nov. 2023 - 13 nov. 2023 B
- Wiring**
Wiring Plan Captors
🕒 28 oct. 2023 - 5 nov. 2023
- Wiring**
Wiring plans GPS
🕒 28 oct. 2023 - 5 nov. 2023 JC

+ Ajouter une carte 📄

Done ...

- Conception**
Final Design
🕒 3 nov. 2023 - 9 nov. 2023
- Wiring**
Gathering WP
🔥 1 🕒 2 nov. 2023 - 8 nov. 2023 EK
- Conception**
3D Impression Prototype
🕒 8 nov. 2023 - 10 nov. 2023 JC
- Wiring**
Wiring plan Parachute
🕒 28 oct. 2023 - 5 nov. 2023 EK
- Technical Specification**
Report

+ Ajouter une carte 📄

Done ...

- Technical Specification**
Report
🔥 1 🕒 23 oct. 2023 - 26 oct. 2023 B
- Technical Specification**
Components list
🕒 7 nov. 2023 - 10 nov. 2023 EK
- Technical Specification**
Upgrade Client TS
🕒 16 oct. 2023 - 23 oct. 2023 AM
- Technical Specification**
Decisional Matrix on GPS
🕒 16 oct. 2023 - 23 oct. 2023
- Technical Specification**
Green Plan

+ Ajouter une carte 📄

Done ...

- Technical Specification**
Green Plan
🕒 16 oct. 2023 - 23 oct. 2023 EK
- Technical Specification**
State of Art Design
🕒 16 oct. 2023 - 22 oct. 2023 RL
- Technical Specification**
Upgrade octopus diagram
🕒 16 oct. 2023 - 23 oct. 2023 AM
- Management**
Gantt Chart Amelioration
🕒 16 oct. 2023 - 22 oct. 2023 B
- Technical Specification**
Calculation sheet
🕒 16 oct. 2023 - 22 oct. 2023 B

+ Ajouter une carte 📄

Appendix 2 - Backlog

Sprint 3 ☆ 👤 000 Tableau ▼

To Do ...
+ Ajouter une carte 📄

Doing ...
+ Ajouter une carte 📄

Done ...

- Sprint3**
Diapo Sprint 3
🕒 24 nov. 2023 - 6 déc. 2023
- Code**
Coder l'acquisition d'image
🕒 21 oct. 2023 - 11 déc. 2023 **AM**
- Material**
Do a BOM (Bill of Materials)
🕒 16 nov. 2023 - 1 déc. 2023 **EK**
- Conception**
Système de déploiement des pods
🕒 21 nov. 2023 - 4 déc. 2023
- Material**
Order of components
🕒 4 déc. 2023 - 10 déc. 2023 **B**
- + Ajouter une carte 📄

Done ...

- Technique**
Review expert technique
👍 2 🕒 15 déc. 2023 - 18 déc. 2023 **RL**
- Conception**
Faire un patron de la huskylens
🕒 22 nov. 2023 - 25 nov. 2023
- Sprint3**
Envoyer RDV au client
🕒 6 déc. 2023 - 6 déc. 2023 **B**
- Test**
test resistance nastran
👍 1 🕒 9 nov. 2023 - 9 nov. 2023 **RL**
- + Ajouter une carte 📄

Done ...

- impression remplissage nid d'abeilles
🕒 15 nov. 2023 - 27 nov. 2023 **JC**
- Technique**
Voir kazemi pour les WP
👍 1 🕒 23 nov. 2023 - 24 nov. 2023 **RL**
- Technique**
compilation WP en 1
👍 1 🕒 24 nov. 2023 - 27 nov. 2023 **RL**
- Test**
Simulate impact on the ground via inventor
🕒 15 nov. 2023 - 15 nov. 2023 **B**
- + Ajouter une carte 📄

Done ...

- Test**
Faire le test de resistance : Spoil : ca s'est mal passé mdr
🕒 30 nov. 2023 - 30 nov. 2023
- calcul poids total
🕒 15 nov. 2023 - 27 nov. 2023 **EK**
- Conception**
revoir le proto pour accueillir la huskylens
🕒 1 déc. 2023 - 5 déc. 2023
- Sprint3**
Envoyer un mail a chabrol
🕒 4 déc. 2023 - 4 déc. 2023 **B**
- Technique**
Trello
🕒 15 nov. 2023 - 6 janv. **B**
- + Ajouter une carte 📄

Done ...

- Technique**
envoi mail a Mr. Chabrol pour précisions
🕒 6 déc. 2023 - 6 déc. 2023 **B**
- Technique**
Modifier gantt
🕒 4 déc. 2023 - 5 déc. 2023 **EK**
- Technique**
Aller voir bilasse
🕒 14 nov. 2023 - 15 nov. 2023 **B**
- Material**
Recherche sur le type de flex recyclable
🕒 14 nov. 2023 - 15 nov. 2023 **JC**
- Test**
test résistance CanSat #1
- + Ajouter une carte 📄

Done ...

- Test**
test résistance CanSat #1
🕒 15 nov. 2023 - 15 nov. 2023 **JC**
- Technique**
Voir Mr Ragot pour savoir si on peut imprimer en flex
🕒 23 nov. 2023 - 24 nov. 2023 **JC**
- Test**
brainstorming impression/test resi
🕒 15 nov. 2023 - 27 nov. 2023 **JC**
- Material**
Recherche sur type de toile pour parachute
🕒 12 nov. 2023 - 14 nov. 2023
- Technique**
Aller voir coudreuse pour info sur
- + Ajouter une carte 📄

Done ...

- Technique**
Aller voir coudreuse pour info sur le bon de commande
🕒 21 nov. 2023 - 21 nov. 2023 **B**
- Technique**
Review Sprint 2
🕒 15 nov. 2023 - 17 nov. 2023 **B**
- Material**
Trier les composants
🕒 15 nov. 2023 - 17 nov. 2023 **EK**
- Conception**
Nouveau CAD Pod
🕒 21 nov. 2023 - 4 déc. 2023
- Conception**
Nouveau CAD cap
🕒 21 nov. 2023 - 4 déc. 2023
- Conception**
Nouveau CAD coque
🕒 21 nov. 2023 - 4 déc. 2023
- Technique**
recherche pliage parachute
👍 2 🕒 23 nov. 2023 - 30 nov. 2023 **RL**
- + Ajouter une carte 📄

Appendix 2 - Backlog

Sprint 4 ☆ 👤 000 Tableau ▾

À faire ...
+ Ajouter une carte 📅

En cours ...
+ Ajouter une carte 📅

Terminé ...

Wiring
cablage
🔖 2 👁 ⌚ 11 janv. - 14 janv. RL

Conception
méthode pliage parachute
⌚ 15 janv. - 16 janv. EK

Conception
découpe parrachute + 1ère conception
🔖 2 👁 ⌚ 15 janv. - 17 janv. RL

Conception
CAO améliorée
⌚ 11 janv. - 16 janv.

Technique / Calcul
recherche théorique diriger parachute
+ Ajouter une carte 📅

Terminé ...

🔖 1 👁 ⌚ 13 janv. - 15 janv. RL

Conception
impression Cansat
⌚ 17 janv. - 18 janv. JC

calcul resistance
🔖 1 👁 ⌚ 15 janv. - 15 janv. RL

Code
avancer coder+
⌚ 12 janv. - 15 janv. AM

Sprint 4
canva prez
⌚ 11 janv. - 15 janv.

Commande
bon de commande nappe
⌚ 14 janv. - 14 janv. EK

+ Ajouter une carte 📅

Technique / Calcul
calcul condensateur
🔖 1 👁 ⌚ 15 janv. - 15 janv. RL

Technique / Calcul
RCE2
🔖 1 ⌚ 15 janv. - 15 janv. B

+ Ajouter une carte 📅

Appendix 2 - Backlog

Sprint 5 ☆ 👤 000 Tableau ▾

à faire

+ Ajouter une carte

en cours

+ Ajouter une carte

terminé

Technique

Borrow drone

🕒 14 févr. - 15 févr.

Parachute

sewing the part of the parrachute

🔖 3 🕒 4 févr. - 7 févr. RL

Code

Adjust code to launch test

🕒 19 févr. - 19 févr. AM

Test

2nd launch test

🕒 9 févr. - 9 févr.

Technique

try launching with the real sail

🔖 3 🕒 17 févr. - 19 févr. RL

+ Ajouter une carte

terminé

Parachute

cutting real parrachute fabric

🔖 3 🕒 13 févr. - 19 févr. RL

Conception

adjust details CAO

🕒 29 janv. - 30 janv.

Wiring

wiring on the raspberry given by the professor

🔖 1 🕒 9 févr. - 9 févr. RL

Conception

print right legs

🕒 13 févr. - 13 févr. JC

Parachute

calculation of the area for the parachute

terminé

calculation of the area for the parachute

🔖 2 🕒 11 févr. - 13 févr. RL

Code

code for the sensor

🕒 9 févr. - 11 févr.

Technique

image recognition

🕒 3 févr. - 9 févr. AM

Conception

make design

🕒 12 févr. - 16 févr. AM

Code

keep coding

🕒 19 janv. - 19 févr. B

terminé

Wiring

order components or find them in the lab (Wires, convertor, SD card, gear adaptator, paint for the target, sponge)

🕒 8 févr. - 10 févr.

Parachute

decisional matrix parrachute

🕒 9 févr. - 15 févr. EK

Conception

inner rack done

🕒 16 févr. - 16 févr. JC

Parachute

cutting parachute prototype

🔖 1 🕒 16 févr. - 18 févr. RL

Technique

terminé

Parachute

cutting parachute prototype

🔖 1 🕒 16 févr. - 18 févr. RL

Technique

RCE2

🕒 10 févr. - 16 févr. B

Conception

pulley system

🕒 19 janv. - 20 janv. RL

Parachute

folding way parachute

🕒 16 févr. - 17 févr. EK

Technique

gather the previous things from orders

🕒 15 févr. - 16 févr. EK

+ Ajouter une carte

Appendix 2 - Backlog

Sprint 6 ☆ 🧑 000 Tableau ▾ 🔍 ⚡ ⚙ Filtres RL AM B EK +1

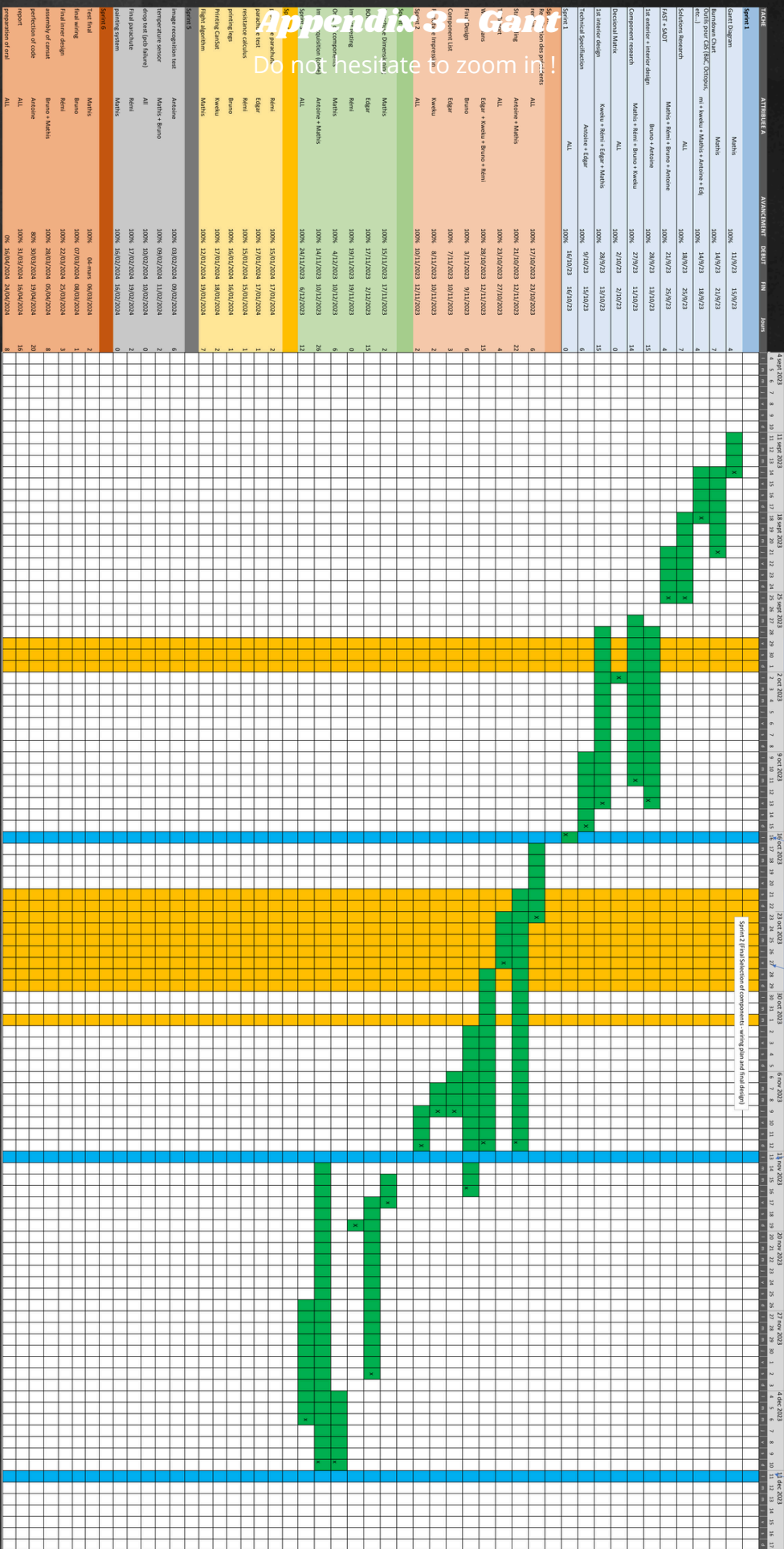
To Do	Doing	review	Done
+ Ajouter une carte 📄	+ Ajouter une carte 📄	+ Ajouter une carte 📄	
			documents modification fichiers pour logique des pieces 🕒 5 févr. - 12 févr. EK
			amelioration stencil creation 🕒 8 avr. AM
			amelioration design of cansat 📝 15 avr. AM
			documents question coudreuse 🕒 16 avr. B
			software code amelioration 🕒 20 - 15 AM
			+ Ajouter une carte 📄

Done	Done	Done
code amelioration 🕒 29 mars - 16 avr. AM	hardware wiring parachute 📈 3 🕒 7 mars - 18 mars RL	amelioration final sewing 📈 3 🕒 19 févr. - 6 mars RL
hardware cansat assembly 🕒 28 mars	hardware printing parts 🕒 21 mars - 22 mars JC	documents question ragot 🕒 12 avr. B
hardware soldering wires 🕒 7 mars - 8 mars	software finalisation algo 🕒 17 févr. - 12 mars B	documents final report 🕒 12 avr. - 16 avr. EK
tests drop test 1 🕒 4 mars - 6 mars AM	tests drop test 2 🕒 7 mars - 8 mars EK	hardware batteries soldering 🕒 8 avr.
hardware assembly inner parts 📈 3 🕒 22 mars - 25 mars RL	amelioration final sewing 📈 3 🕒 19 févr. - 6 mars RL	hardware servomotor fixing 🕒 8 avr. EK
hardware		+ Ajouter une carte 📄

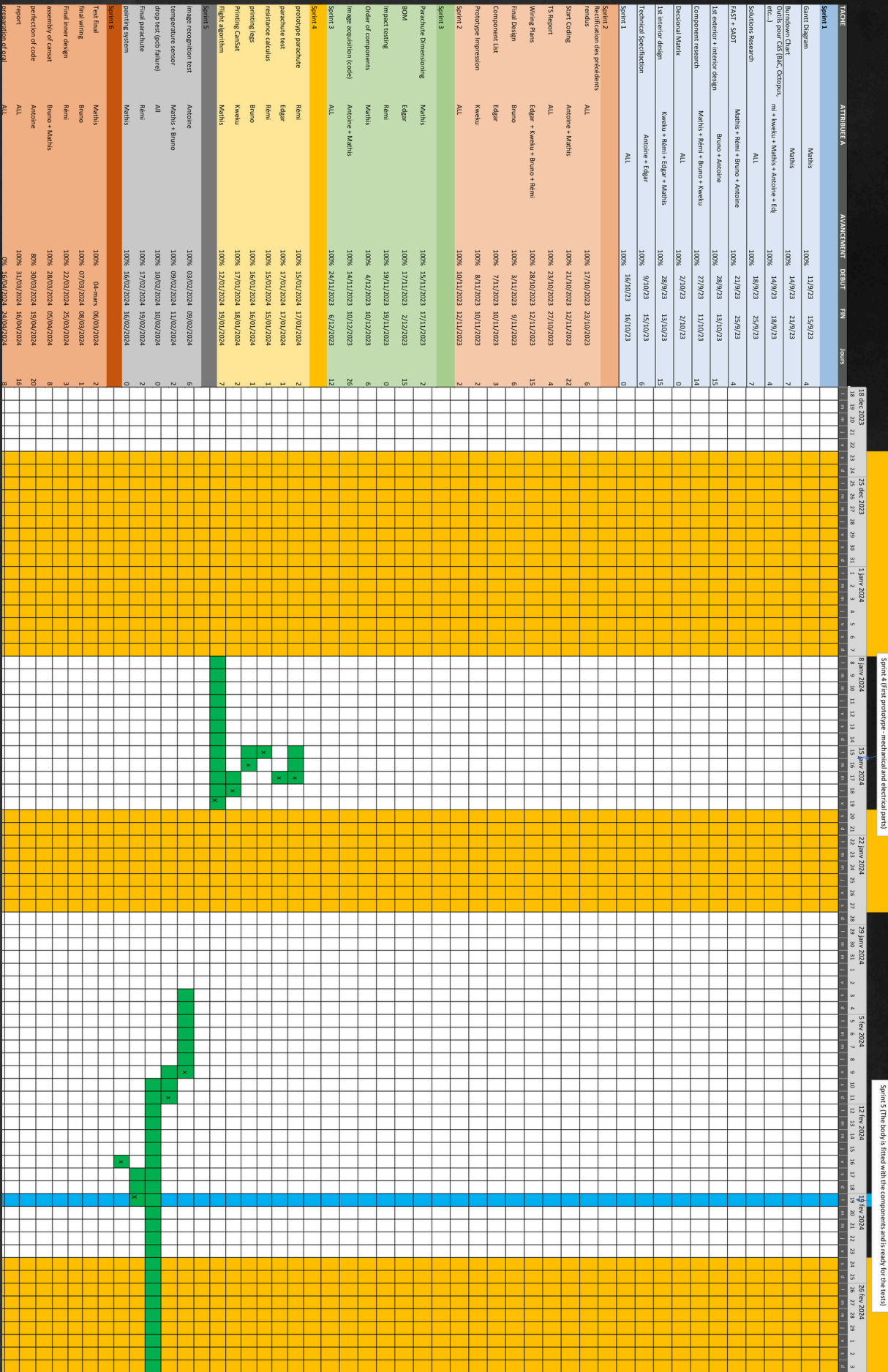
hardware batteries soldering 🕒 8 avr.
hardware servomotor fixing 🕒 8 avr. EK
amelioration adding of bar support 📈 2 🕒 11 avr. RL

Appendix B - Gantt

Do not hesitate to zoom in!



Appendix 3 - Gant



Appendix 3 - Gant



Appendix 4 - Code

```
1 import numpy as np
2 import cv2
3
4 # Video capture initialization
5 cap = cv2.VideoCapture(0)
6
7 while True:
8     ret, frame = cap.read()
9     if not ret:
10         break
11
12     # Convert to HSV color space
13     hsv = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)
14
15     # Define color range for white
16     lower_white = np.array([0, 0, 134])
17     upper_white = np.array([180, 43, 255])
18
19     # Create a mask to isolate the white color
20     mask = cv2.inRange(hsv, lower_white, upper_white)
21
22     # Detect circles using the mask
23     circles = cv2.HoughCircles(mask, cv2.HOUGH_GRADIENT, 1.2, 100)
24
25     if circles is not None:
26         circles = np.round(circles[0, :]).astype("int")
27         for (x, y, r) in circles:
28             cv2.circle(frame, (x, y), r, (0, 255, 0), 4)
29             cv2.rectangle(frame, (x - 5, y - 5), (x + 5, y + 5), (0, 128, 255), -1)
30
31     # Split the frame into 4 quadrants
32
33     height, width, _ = frame.shape
34     top_left = frame[:height//2, :width//2]
35     top_right = frame[:height//2, width//2:]
36     bottom_left = frame[height//2:, :width//2]
37     bottom_right = frame[height//2:, width//2:]
38
39
40
41     # Detect circles in each quadrant
42
43     circles_tl = cv2.HoughCircles(cv2.inRange(top_left, lower_white, upper_white), cv2.HOUGH_GRADIENT, 1.2, 100)
44     circles_tr = cv2.HoughCircles(cv2.inRange(top_right, lower_white, upper_white), cv2.HOUGH_GRADIENT, 1.2, 100)
45     circles_bl = cv2.HoughCircles(cv2.inRange(bottom_left, lower_white, upper_white), cv2.HOUGH_GRADIENT, 1.2, 100)
46     circles_br = cv2.HoughCircles(cv2.inRange(bottom_right, lower_white, upper_white), cv2.HOUGH_GRADIENT, 1.2, 100)
47
48
49     # Draw circles on the original frame
50
51     if circles_tl is not None:
52         circles_tl = np.round(circles_tl[0, :]).astype("int")
53         for (x, y, r) in circles_tl:
54             cv2.circle(frame, (x + width//2, y), r, (0, 255, 0), 4)
55
56
57     if circles_tr is not None:
58         circles_tr = np.round(circles_tr[0, :]).astype("int")
59         for (x, y, r) in circles_tr:
60             cv2.circle(frame, (x + width//2, y), r, (0, 255, 0), 4)
61
62
63     if circles_bl is not None:
64         circles_bl = np.round(circles_bl[0, :]).astype("int")
65         for (x, y, r) in circles_bl:
66             cv2.circle(frame, (x, y + height//2), r, (0, 255, 0), 4)
67
68
69     if circles_br is not None:
70         circles_br = np.round(circles_br[0, :]).astype("int")
71         for (x, y, r) in circles_br:
72             cv2.circle(frame, (x, y + height//2), r, (0, 255, 0), 4)
73
74
75     # Stock the result
76     cv2.imshow('Frame', frame)
77
78     if cv2.waitKey(1) & 0xFF == ord('q'):
79         break
80
81 cap.release()
82 cv2.destroyAllWindows()
83
84 image_path = "icameteor"
85 img = cv2.imread(image_path)
86 if img is not None:
87     cv2.imshow("Display Window", img)
88     cv2.waitKey(0)
89     cv2.destroyAllWindows()
90 else:
91     print("Error: Image not loaded.")
92
93 while True:
94     ret, frame = cap.read()
95     if not ret:
96         break
97
98     import os
99
100 import glob
101 import time
102
103 # import necessary libraries for servo control
104 import RPi.GPIO as GPIO
105
106 def initialize_servo(pin):
```

Appendix 4 - Code

```
107     GPIO.setmode(GPIO.BCM)
108     GPIO.setup(pin, GPIO.OUT)
109     pwm = GPIO.PWM(pin, 50) # 50 Hz frequency
110     pwm.start(0)
111     return pwm
112
113 def servo_angle(pwm, angle):
114     duty = angle / 18 + 2
115     GPIO.output(pin, True)
116     pwm.ChangeDutyCycle(duty)
117     sleep(1)
118     GPIO.output(pin, False)
119     pwm.ChangeDutyCycle(0)
120
121 # servo is connected to GPIO pin 17
122 servo_pin = 17
123 pwm = initialize_servo(servo_pin)
124
125 while True:
126     ret, frame = cap.read()
127     if not ret:
128         break
129
130     frame_center = (frame.shape[1] // 2, frame.shape[0] // 2)
131     hsv = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)
132     mask = cv2.inRange(hsv, lower_white, upper_white)
133     circles = cv2.HoughCircles(mask, cv2.HOUGH_GRADIENT, 1.2, 100)
134
135     if circles is not None:
136         circles = np.round(circles[0, :]).astype("int")
137         # Find the circle closest to the center
138         closest_circle = min(circles, key=lambda c: (c[0] - frame_center[0])**2 + (c[1] - frame_center[1])**2)
139         x, y, r = closest_circle
140
141         # Calculate error from the center of the frame
142         error_x = x - frame_center[0]
143
144         # Assuming a simple proportional control
145         correction = -error_x / 10
146         new_angle = 90 + correction
147         servo_angle(pwm, new_angle)
148
149     else:
150         # No target detected, perform a default maneuver
151         # Rotate in place or circle search pattern
152         servo_angle(pwm, 45) # turn left
153         # Alternatively, implement a more complex search pattern
154
155     cv2.imshow('Frame', frame)
156     if cv2.waitKey(1) & 0xFF == ord('q'):
157         break
158
159 cap.release()
160 cv2.destroyAllWindows()
161 GPIO.cleanup() # clean up GPIO on exit
162
163 # Function to read the raw temperature data from the sensor
164 def read_temp_raw(device_file):
165     with open(device_file, 'r') as f:
166         lines = f.readlines()
167     return lines
168
169 # Function to parse the raw temperature data and convert it to Celsius
170 def read_temp(device_file):
171     lines = read_temp_raw(device_file)
172     while lines[0].strip()[-3:] != 'YES':
173         time.sleep(0.2)
174         lines = read_temp_raw(device_file)
175     equals_pos = lines[1].find('=')
176     if equals_pos != -1:
177         temp_string = lines[1][equals_pos+2:]
178         temp_c = float(temp_string) / 1000.0
179         return temp_c
180
181 # Find the path to the sensor device file
182 base_dir = '/sys/bus/w1/devices/'
183 device_folders = glob.glob(base_dir + '28*')
184
185 if not device_folders:
186     print("No DS18B20 sensors found!")
187     exit(1)
188
189 device_file = device_folders[0] + '/w1_slave'
190
191 # Continuously read and print the temperature every 10 seconds
192 while True:
193     temperature = read_temp(device_file)
194     print(f"Temperature: {temperature:.2f}°C")
195     time.sleep(10)
196
```