Red Dragons : The origins
by L'ENSMACanSat

Sébastien Bardot, Camille Barnier, Pascal Bouda, Xuan Khanh Do, Sylvain Hochede, Eric Li, Trong Thuc N'guyen, Ismaël Paqueriaud, Adrien Polidano, Revyll-Jones Ratanga

***Abstract*– In this paper, we will introduce our team and the CanSat project. As students of ENSMA[1], a French engineering school, we have decided to attempt to achieve four missions: Atmospheric probing, photography, rover and "come back" missions. In order to meet volume restrictions for international class[2], our CanSat frame, a 3D printed structure, was designed to embed all needed components including motors, microcontrollers, transmitters… limited space has been a challenging issue.**

To control our CanSat in the air, we have chosen a "Nasa PARAWing"[3] and we will try a precision landing. On the ground, special wheels will be deployed to conduct a discovery mission. In both phases, photos will be taken and transmitted to the ground station.

So far, we have not completed integration tests, neither raised enough funds to afford housing during competition. However, we are confident that these issues will be solved by August.

I. INTRODUCTION

ONCE upon a time, Red Dragons : The Origins (RDTO)...

Last year at the origins of the project, a team of ENSMA students (IATeam[4]) chose the set of missions and corresponding components but they left the competition on March 2012. We inherited their work for the next year's competition...

Six month later, the class started and the choice of "Bureau d'étude" (BE) had to be made. Without hesitation, five third year students (A3) took over the project. Facing the work to be done, and considering the short time to do it (A3 class ends on March), they decided to recruit some first year students (A1) to help them. Five of them responded, and contributed ideas concerning the mechanical part of RDTO. The team was created, facing the challenge and ready to solve the arising problems (see Fig.1 : Picture of L'ENSMACanSat team).

[1]http://www.ensma.fr/front/page.php?id=1
[2]http://www.planete-sciences.org/espace/CanSat-France-Competition-2012
[3]http://freedom2000.free.fr/NPW_index.html
[4]https://www.facebook.com/pages/CanENSMASat/238349652888681

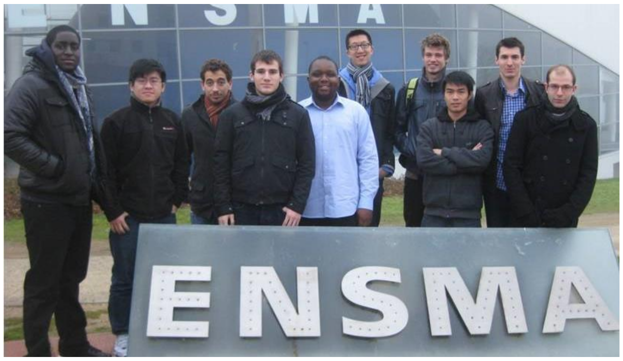

Fig.1 : Picture of L'ENSMACanSat team

This is the story of our team "L'ENSMACanSat"[5], but a question is yet to be answered. Why is the project named "Red Dragon : The Origins"?

First of all, why "Dragon"?

The dragon is the ENSMA mascot, for us this word must be in the CanSat name.

Secondly, why "Red"?

A dragon without a color is not a real dragon, nor is it terrifying, so our dragon needed a color. The red is in reference to the logo of ENSMA Space Project (ESP[6]), a student association in which we take part. It also refers to Thomas Harris' book and suggests the monster inside...

Finally, why "The Origins"?

RDTO is the first of a long series of ENSMA CanSat... we hope that it will be continued in the next few years (in any case we will help our successors).

For the first ENSMA participation in this competition, we are very ambitious: our objective is to design a CanSat that could complete four missions. The first mission is a telemetry mission. This one is now imposed. We believe that this mission is crucial, for when planning to colonize a new planet, the first things you need to know are atmospheric conditions. This is why, we chose to record the following data during the fall and on the ground: temperature, pressure, and hygrometry.

The second mission is a photography mission. We want to take some pictures during the descent and after landing. This is an essential mission because combining both photos (aerial and terrestrial ones) will enable us to make a cartography of the land.

[5] http://esp.ensma.fr/?page_id=64
[6] http://esp.ensma.fr/

The third mission is a “rover” mission. RDTO shall be able to move on the ground and perform a discovery mission by following some way-points (GPS coordinates).

The last mission is a “come back” mission. RDTO, thanks to a “Nasa PARAWing” controlled by two servomotors combining to current GPS coordinates, will be able to land in an area around drop point.

II. Context of Development

A. Club

The project « Red Dragons : The Origins » is led by L'ENSMACanSat. Our team is a member of ENSMA Space Project, which is an association of ENSMA students created in 2011 to regroup all the projects related to space at ENSMA, in particular the CNES projects. The club ENSMA Space Project includes the following project:

- 2 Experimental rockets (CNES project) ;
- 2 Minirockets (CNES project) ;
- 1 Experimental balloon (CNES project) ;
- 1 CanSat (CNES project) ;
- 1 mini satellite FahrSat[7], YourSpaceAvenue and Fahrenheit project.

Those projects are financed by the school, some of ENSMA research departments, the P'prime laboratory and we also raised funds from the department of Vienne and the region Poitou-Charentes.

During the last edition of the CanSat competition, the IATeam (ENSMA students) enrolled in the competition but dropped out of it before it actually took place.

The following table shows the member of the team, their levels in ENSMA and their participation in the project.

TABLE 1 : DISPATCHING OF THE WORK ON RDTO.

Name	Level	Participation
Camille Barnier	A1	Mechanics and Wings
Pascal Bouda	A1	Mechanics and Wings
Sylvain Hochede	A1	Mechanics and Wings
Eric Li	A1	Mechanics and Wings
Adrien Polidano	A1	Mechanics and Wings
Xuan Khanh Do	A3	Electronics and Embedded System
Ismaël Paqueriaud	A3	Electronics and Embedded System
Trong Thuc N'guyen	A3	Ground Station
Revyll Jones Ratanga	A3	Ground Station
Sébastien Bardot	A3	Mechanics, Electronics, Budget and Project Manager

[7] https://www.fahrenheit-pi.com/index.php/slideshow/37-fahrsat-la-grande-aventure-spatiale

B. Work plan

This year, five A3 students picked up where their predecessors had left the project. We planned the missions we wanted to achieve and we chose to recruit some A1 and A2 to help us on the mechanical parts and also to insure that after we graduate, some students will be able to take the project over during the subsequent years. Five A1 joined us in this adventure. They contributed a lot of ideas that concerned the structure and the wings. They made the wing and the wheels. The A3 focused on computer science and administration. Moreover, since they are knowledgeable in CAO, they designed the CanSat on Catia integrating A1's ideas. As a consequence, the project was shared as follow:

- A1 students designed the wing, brought some ideas on the mechanical parts including the shape of the wheels
- Ismaël and Khanh worked on the embedded software
- Revyll-Jones and TrongThuc developed the ground station;
- Sébastien is in charge of the administrative paperwork, worked on design with the CAO software Catia and on CanSat assembly.

This repartition is a rough view of the tasks because the team thinks as a group and members worked together when facing a problem.

C. Budget

Our budget is divided between three aspects: manpower, equipment and the costs of the competition (transport and accommodation).

The cost of manpower represents the time spent by the team on the project. It represents about 1000 hours of our time with the following repartition:

- 100 on choice and definition of missions;
- 500 on programming (source codes and ground station);
- 200 on integration test;
- 150 on design and realization of mechanical parts (including wings);
- 50 on administration (fund raising, presentation of the project, sheets for the organization...).

Equipment represents a budget of 600 Euros.

Transportation for all team members was estimated to approximately 2000 Euros. To this day, we don't have enough funds to pay for transportation and housing for all team members.

III. Definition of the Missions

A. Atmospheric probing mission

When we started to think about this mission, the new recommendations were not published yet. We choose to plan this mission because this is an essential mission that space probes have to complete. The role of the RDTO will be to measure pressure, temperatures and air humidity before and after landing. This data will enable us to know the atmospheric conditions at different altitudes and to establish a law which combines altitude, temperature and pressure: an atmospheric model. This was made possible thanks to:

- an Arduino Nano shared between Atmospheric and come back missions;
- an hygrometry sensor (based on Honeywell's HIH-4030 humidity sensor);
- a barometric pressure sensor (based on Bosch's BMP085 sensor) which also measured temperature;
- a modem (XBee Pro 60mW Chip Antenna) installed in the CanSat as a transceiver mounted on a board (XBee Explorer Regulated);
- a modem (XBee Pro 60mW U.FL Connection) implanted in the ground station as a receiver mounted on a USB connecting board(Xbee Explorer USB).

The Arduino Nano collects data and makes a frame which is sent via an XBee. That data is recorded and exploited in our ground station.

B. Photography mission

This mission is of equal importance. Indeed, having pictures of a new planet and its cartography, through to a discovery mission, will allow for a space probe to land in a safe and adequate environment. Furthermore, photographies can help distinguish the different elements of the planet. During this mission, we will shoot photos before and after landing, send them to the ground station and record them on an embedded mini-SD card and on the ground station. To achieve this mission, we will need to use :

- a dedicated Arduino Nano (not shared with other mission);
- a camera (LinkSprite JPEG Color Camera TTL Interface);
- one support of microSD Card (MicroSD card breakout board+) with a microSD card of 1 Go (Flash Memory - microSD 1GB);
- a modem (XBee Pro 60mW Chip Antenna) in the CanSat as transceiver mounted on a board (XBee Explorer Regulated);
- a modem (XBee Pro 60mW U.FL Connection) in the ground station as receiver mounted on a USB connecting board (Xbee Explorer USB).

C. Rover mission

This mission starts after landing when the devise is on the ground. The objective of this mission is to enable the RDTO to move on land in order to do a discovery mission. To do this mission, the material needed is:

- an Arduino Nano shared with Atmospheric and "come back" missions;
- a driver motor board (Motor Driver 1A Dual TB6612FNG);
- two motors (Micro Metal Gearmotor 100:1);
- two wheels (spiral stress).

D. "Come back" mission

This mission is essential in guaranteeing the safe return of the RDTO by giving it a GPS position. To do this mission, RDTO will use a wing controlled by servomotors. The following material will be needed to performe this precision landing:

- an Arduino Nano shared with Atmospheric and Rover missions;
- two servomotors (a low cost of Hitec HS-55);
- a wing type Nasa PARAWing;
- GPS sensor (LS20031 GPS receiver).

All this missions have been programmed on Arduino's board thanks to Sparkfun[8] and Arduino[9] websites and with AVR Studio[10].

IV. RDTO Architecture

A. Electrical architecture

RDTO is powered by a Polymer Lithium Ion Battery - 1000mAh 7.4v. This battery is directly connected to the Arduino. We used voltage regulator to obtain two more voltage, 3.3V and 5V. The following diagrams show the two Arduino-Modem along with their accessories.

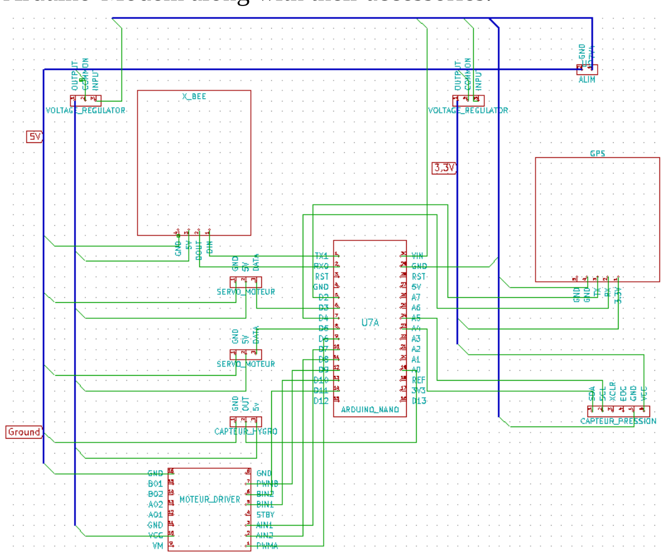

Fig.2: Kicad[11] diagram of Arduino-Modem 1 which provide Atmospheric probing, Rover and Come back missions

[8] https://www.sparkfun.com/tutorials
[9] http://www.arduino.cc/
[10] http://www.atmel.com/microsite/avr_studio_5/
[11] http://www.kicad-pcb.org/display/KICAD/KiCad+EDA+Software+Suite

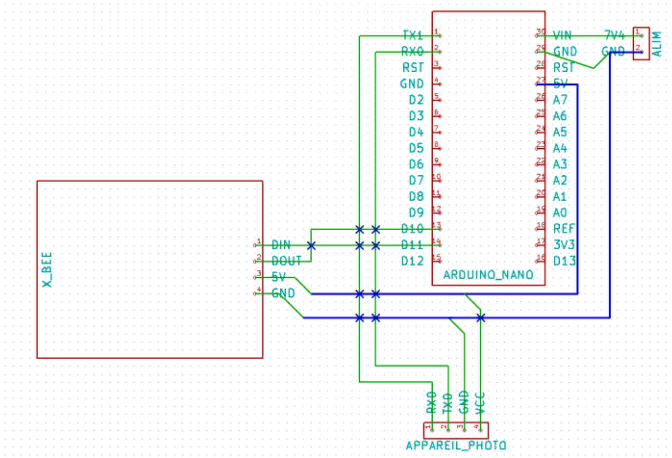

Fig.3 : Kicad diagram of Arduino-Modem 2 which provide Photography mission

The battery will support all the embedded system. To save energy, we are discussing starting the Arduino-Modem 2 at the beginning of RDTO fall. Furthermore, we do not need wheels on air and wing on ground, so we are thinking about ejecting the wing when RDTO lands and put the motor driver cards (and the motor) in standby mode during the fall.

Furthermore, Fig.2 and Fig.3 show the two different embedded system connections but not their geometry in RDTO. In fact, inside RDTO there is one Polychlorinated biphenyl (PCB) board by Arduino, one PCB board to link motors and the drive motor board and one another to link the servomotor's container with the rest of RDTO before the landing.

B. Mechanical parts

The RDTO is printed. In fact, as we want to run four missions on two XBee-Arduino Nano, the structure has been designed to fit the onboard equipment. The following picture shows RDTO made with a CAO software (without the wheels).

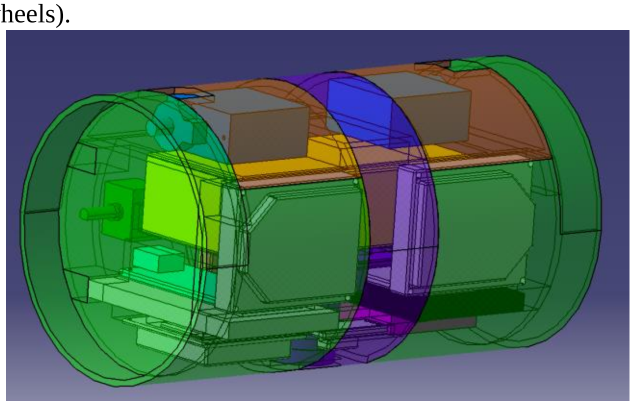

Fig.4 : RDTO made with a CAO software

The mechanicals parts and onboard system are linked via the frame. The Arduino are mounted on PCB (one for each, red and black on the picture). The following view shows all disassemble elements without electronic equipments and wheels.

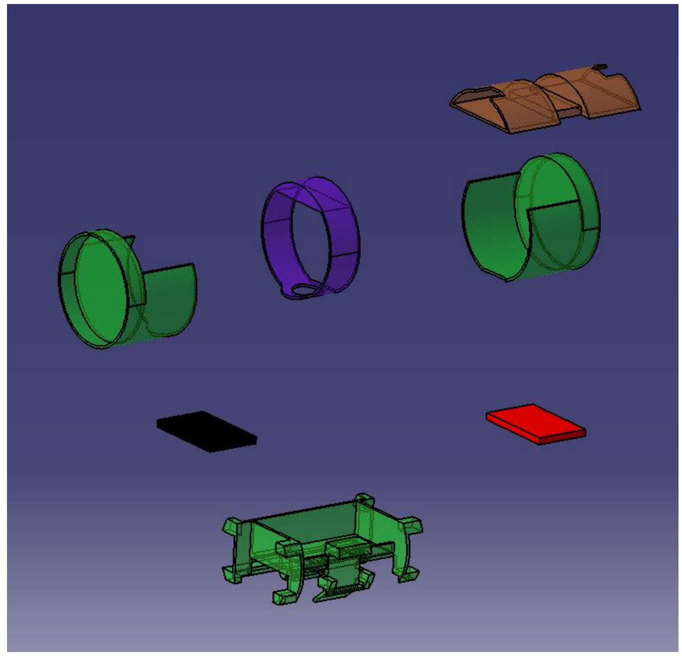

Fig.5 : RDTO disassembly

In orange, this part RDTO retains a servomotor's container. This one is link with the pink piece to the frame. When RDTO touch the ground, the pink part opens and frees the servomotor's container. Furthermore, the pink part prevents RDTO rolling over during the rover mission.

On each side, a carter protects the equipment and forms a hole to include the wheels.

C. Telemetry

We use two modems XBee with an USB connector. Last year two XBee had already been ordered and the antenna was built on chip. We want to get two identical modems to fit into the architecture (two in the CanSat and two in the ground station). However, this year we accidentally ordered a model that does not have a built-in antenna.

Fig.6 : On the left a XBEE with an antenna on chip and on the right a XBEE with an U.FL connector

As shown in the figure above, the model we ordered has a miniature connector U.FL.

We ordered then a U.FL - RP SMA adapter, and have to find an antenna with such connections and adapted to 2.4GHz XBee. Resulting from the addition of these antennas, two new modems will be used in the ground station.

For the connection between CanSat and the ground station, we used a cJSON-like form, different types of signals were defined to connect efficiently CanSat and the ground station. For example, information about temperature and pressure is transmitted on the following format:

{"id"="data","temp"=262,"pres"=100852,"hygro"=36,"lt"=3024254,"ln"=-9782634,"sp"=1,"al"=0}

V. Ground Station

The ground station is a computer device that initiates a private communication between ground operators and the can.

A. Presentation

The Ground Station we designed consists in a simple HMI that has two main objectives. Firstly, it presents data acquired by the can sensors in multiple and organized views. For example, the user can choose whether to focus on camera acquisitions or telemetric measurements. It can also display any received frames to ensure the communication with the satellite remains active.

Secondly, we wanted the ground operator to be able to control the CanSat while in the air. Therefore we implemented functions that allow the user to trigger or change the functional mode or to stop and turn off the CanSat.

Since the previous bureau provided the frame to build the ground station, our job mostly consisted in designing the system that links the ground station with the CanSat. The picture below gives a first sight of our ground station. It emphasizes the data viewer (the "DonneesCansat" tab) and the frame viewer (the "SuiviTrames" tab).

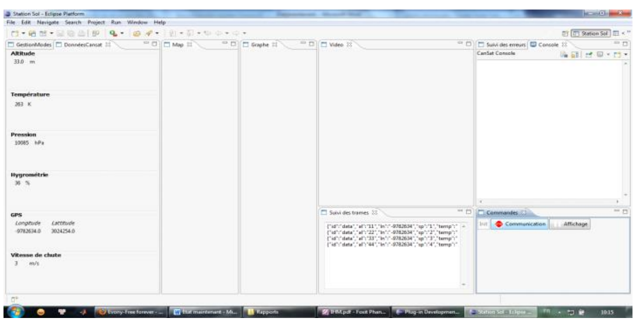

Fig. 7: Data perspective

Speaking of conception, the Ground Station is based on the MVC (Model-View-Controller) architecture. This design pattern decouples data access, control code, and data presentation-user interaction. It was designed using EMF (Eclipse Modeling Framework) plugin of Eclipse. This plugin allows to create the meta-model via UML schema. Then it can generate the corresponding Java implementation classes and enable data persistence with XMI standard. That part was once again done by the previous Cansat team. We did not modify the station model but we added many contributions to make it work properly. We'll now expand on our contributions.

B. Our contributions

As mentioned earlier, the main goals were to be able to display CanSat data on the ground station interface and control the can by sending it predefined orders. The second task was completed by our predecessors. They added buttons on the HMI in order to control the CanSat.

Then they implemented ready-to-use methods to transmit orders to the CanSat. Yet, there were no communications established between the ground and the can. To achieve our first goal necessary to achieve our other goals, we simply had to implement communications between the can and the ground station. The communication system required a ground modem that could receive data from the remote modem embedded in the can. Since this ground modem was linked to our ground station by a USB cable, we simply had to read the frame sent by the Can, extract useful information and transfer it to the appropriate view on our HMI.

a) Frame model

Before we expand on our process to retrieve, store and display information from the Cansat, we will start by focusing on frame type. We are using JSON frame type to convey data from the can to the ground station. Basically, the data we would like to access are values of the parameters in a JSON frame. As we predicted we would like to extract these values from the frame, we created Java classes for each type of Frame, whose attributes correspond to the parameters. For example, a "Data" frame has a "pressure" parameter for telemetric. Then a "Data" frame is translated as an object of data frame class "Trame_data" class which holds an attribute "pres" as illustrated in the following picture (fig 8. details our frame model). Thus by using the associated getter, we can access the value of pressure. We'll stress that the frame model was also provided by the previous bureau. Yet, we modified it and make it lighter (for embedded purpose) and much simpler to understand understanding.

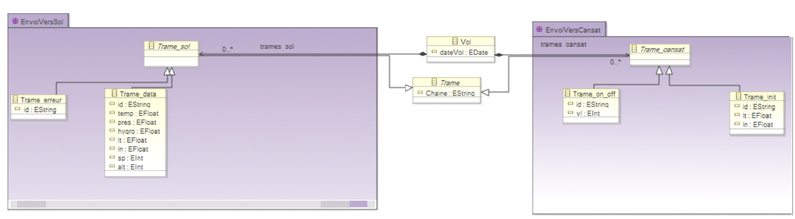

Fig. 8: UML diagram of our frame model

b) Read a data from USB port & translate them

The reading part was the first thing we worked on. Since the whole application was developed with the Java programming language, we had to find the most suitable libraries to read data from a serial port. We finally used the "comm" library and its features to build a class that automatically reads any data once it arrives on the port. The received data (a JSON type frame) was then translated in one Java object class from frame model. For example, an error frame shall be translated in an object of the "Trame_erreur" class of previous illustration. The result object is immediately forwarded to the extraction process.

c) Extract data from the frame & display them

Once the object was built, we still had to access the information it contains. It is because of this phase that we had the frame model re-designed. The current extraction phase now consists in calling a "getter" method to retrieve the desired data.

Once extracted, the data is stored in a data module. This data module enables its subscribers (HMI views) to access the data without letting it be updated concurrently by the arrival of another frame. Since the data module is a frame array, only the last frame recorded can be accessed by the subscribers. It's a basic way to ensure we always have the latest information from our CanSat. The views therefore get updated the instant a new frame arrives.

We tested the ground station communication with the CanSat. It currently meets our expectations. We are able to send/receive and display information to/from our can. We can as well record any flight data in a file and list our flight errors...

Yet, the views are not all filled. We envisage integrating many other features like videos, maps, altitude diagram in a near future.

VI. Conclusion

We can divide our project in four parts:

- Electrical-components
- Mechanical;
- Programming;
- Ground station.

Each part will be developed on this following.

A. Electrical-components part

At this moment, about 30 Euros of components are missing: the two antennas (2,4 GHz), one switch for the modems in RDTO and one XBee Explorer (a problem with the previous order, we received an Xbee USB Explorer). This order will be passed before the end of June.

Concerning the electrical part, the PCB board is almost finished drawing it and it will be made before the end of the month of June too.

The electrical test, doing to determine the autonomy of RDTO, will be take place when the ordering will arrived. If the RDTO battery is not sufficient, a new one will be made to provide enough energy for RDTO.

B. Mechanical part

The drawing of printing pieces are almost finished. We are thinking about transmitting the movement between the motor and the wheels because they cannot be coaxial in RDTO. When this problem will be solved, the drawing will be achieved and the printing will follow. This part of the work should be achieved before the fifteen of June to be sure that the structure will be ready before the fifteen of July.

The wing is already ready.

C. Programming part

The different functions of our RDTO are programmed but structure is not yet assembled. The integration of all the codes will be made after receiving the rest of the components. But, a theory scheduler analysis shows that the two Arduino will correctly work with the charge per unit given, just a bit slower for the photography mission Arduino (four second to take, record and send a picture).

This part will be finished before the end of July.

D. Ground station part

Now RDTO and its ground station communicate. Our ground station shows the data it has received. The following steps are integrating pictures after receiving them and making a diagram for each data.

Nevertheless, as the ground station prints the data, we will not focus on it before the competition and we do not know if we can spend some time on it.

Finally, we hope to assemble RDTO before the end of July to have time to make some tests of all the missions before the competition and check if our CanSat can fly and roll correctly.

VII. Acknowledgment

We would like to thank our school and teachers for enabling us to be part of this adventure. We would especially like to thank Mickaël Baron, Michaël Richard and Emmanuel Grolleau for helping us on programming (Arduino and Ground station), Jean-Marie Roncin and Olivier Ser to for checking our CAO work, Jean-Marie Petit and Thierry to for letting us use the 3D printer.

We would like to also thank our work placement mentors for letting us participate in the competition.

Also all members of ESP for welcoming our team and sharing their knowledge about the C'space.

ENSMA departments for providing us electronics equipment and the means to do the structure parts, and also ENSMA direction, for granting us funds for the other parts of project (especially the wings).

Last but not least, we would like to extend our thanks to all the people who organize the C'Space and the DGA for welcoming us at Biscarrosse (a military missile test center (DGA EM)).