Final Design Review

for **CanSat** competition

Team

SobieskiSat

Changelog 2

Introduction 3

Team composition 3

Available resources and time management 4

Mission overview 4

Primary mission 4

Secondary mission 5

Probe description 6

Introduction 6

Components description 6

Microcontroller board 6

Batteries 6

Atmospheric pressure and temperature sensor 6

Particulate matter sensor 7

Humidity sensor 7

GPS module 7

IMU module 8

Gases sensor 8

Buzzer 9

Photoresistor 9

Measurements 9

Recovery system 10

Mechanical design 11

Weight 11

Case 12

Parachute 12

Transmitting antenna 13

Electrical design 14

Daughterboard design 14

Power consumption 15

Software design 16

System structure 16

Data gathering 17

Event logging and timers 17

Data compression during radio transmission 17

Ground station 18

Radio receiving 18

Ground Station Interface 19

Landing prediction feature 20

Data Analyzing Software 20

Test campaign 21

Radio transmission tests 21
Drone drop tests 21
Thermovision tests - heat visualisation 24
Project summary **25**
Resource management 25
Budget and costs 25
CanSat Value 26
Sponsors 27
Documentation 28
Social media & promotion 28
Facebook 28
Radio promotion 29
Articles 29
Summary 30
Education 30
GitHub Repository 30
Timeline 30
Cansat Characteristics **40**

1. Changelog

- PDR
  - Initially we planned to use an outside communication module with a chip supporting LoRa, but since it is already in CanSat kit provided to us, it was deemed unnecessary and we opted to came with some other ideas for the secondary mission.
  - As of the PDR, our main goal in the secondary mission has changed. Now we are going to collect humidity and video data alongside PM density.
  - In addition to the main goal of the secondary mission we also added plans to develop a descension assistance system (DAS).
- CDR
  - We decided to leave out the camera because we could not find any suitable one for our needs and compatible to our board.
  - We've updated our component list with final components we chose according to our financial capabilities and electrical requirements.
  - Mission stages from PDR were modified due to software development.
  - DAS (Descension Assistance System) was delayed due to prototyping restrictions and ultimately its goal was changed to predict the landing site using linear regression.

- We found that LM35 thermometer was outputting unusable disturbed data while sending via radio. Because of that we use BMP280 as our main thermometer.
- We changed GPS module to Adafruit Ultimate GPS, it includes RTC module so one planned in PDR is not needed anymore.
- Particulate matter sensor was changed to SPS30.

- FDR
    - Added a new MQ-9 sensor to help analysis in secondary mission.
    - We decided to increase promotion of our project in both social and traditional media.
    - We finished a lot more drone drop tests than we expected to carry out at the beginning, for better analysis of data
    - Descension Assistance System was finally examined and implemented into our main code.
    - No more big changes were executed - we focused on developing the systems we decided to use before writing CDR

2. Introduction

2.1. Team composition

Our team consists of 5 students and our tutor, all of us are attending High School No. 2 named after King John III Sobieski in Kraków. Aleksandra Musiał - Our tutor, CS teacher and also an administrator at our school's 3D printing laboratory.

- Jakub Czarny - Team leader & Electro-mechanical designer. His primary role is managing our progress and mission but he also specializes in electronic design and is responsible for choosing components according to our financial and electrical constraints. He is also our 3D designer.
- Andrzej Szablewski - PR spokesperson, he takes care of all PR related missions like finding sponsors, media as well as managing and updating our social media. He also works on programming and testing.
- Maciej Gamrot - High-Level Developer, he is responsible for main system development and communication with ground station. He also works on machine learning, look after our repository and develops our website.
- Damian Legutko - Embedded Developer, his job is to write code for managing data retrieved from sensors, designing backup procedures. On top, Damian designs graphical parts of our project when needed.
- Krzysztof Parocki - Digital Media Creator, he makes a photo-documentation of our progress. He also records and cuts movies based on our creation process. His work is used to promote

our project in social media. He also helps with facebook site managing, testing prototypes and organizing.

2.2. Available resources and time management

We’ve established cooperation with our school that provided us a classroom, where we hold our meetings and 3D printing lab that is under our tutor’s management. We have access to all the tools needed for the project in a public workshop - Hackerspace Kraków. Outside of the school we work on our own as well as in group through communication apps like Messenger and Discord. We meet once a week in our classroom to summarize the previous week and divide tasks for the next few days. A few weeks before completing FDR we were meeting once a week and testing our CanSat with professional drone operator.

2.3. Mission overview

2.3.1. Primary mission

CanSat mission is divided into few stages. Because of the importance of primary mission, mission stages are dependent on the time from hardware oscillator clock and photoresistor, secondary mission readings are used only as reference.

- **Initialization**
  During *Initialization,* software will diagnose every component and save received data in log files on SD card. Checking if everything is working properly would prevent from errors that might occur during the mission. In this stage Can should be standing still on a flat

surface for about 3 seconds because few sensors inside would be calibrating.

- **Await**
  The Can will wait for being placed inside the rocket. Probe will occasionally collect sensor data and *photoresistor* readings will indicate if is put in tray and trigger next stage.
- **Launch**
  Except for light intensity, acceleration values from IMU module would be used as reference to mission stage. Sudden acceleration during lift might cause electrical components to behave in unexpected way, to avoid possible errors most of them will go into sleep mode. If Can could not receive any data after estimated time goes into next stage.
- **Fall**
  As a main part of primary mission, in this stage probe will collect data from all sensors, save them to SD card and send via radio.
- **Recovery**
  Probe goes into recovery stage after estimated time or when it detects landing via accelerometer, gyroscope or GPS. It waits to be found by searching team but still is collecting data and sending radio signals.

2.3.2. Secondary mission

Our main goal in the secondary mission is to gain first information about atmosphere quality, disasters, and also volcanic activity. Using sensors like temperature, pressure, humidity, particulate matter density sensor as well as flammable gases and carbon monoxide sensor we can initially determine if the local atmosphere is suitable for the existence of life. Our idea in case of volcanic activity is based on dropping the satellite inside the volcano, while it’s dormant.

Particulate matter density and humidity can be used for controlling and predicting volcanic activity. Particulate matter is a very important factor in our secondary mission. Dust intensity determines chances of survival for both animal and human beings. High densities are very dangerous for life and are a direct information that the region needs evacuation. Other factors like temperature and humidity are also important as an information about life conditions. Our system can be used in fires as well as other natural disasters. The whole probe is supposed to output the collected data for the processing to happen on the ground. This allows us to quickly adapt to whatever is needed and analyze the data thoroughly.

As for launching the probe in finals we expect to test all systems and check for errors as we don't predict the army range will output dangerous data signalyzing any disasters. The whole point of the probe is life-sensitive data collection which can be interpreted in many different ways.
In addition we will use landing site prediction software to help us understand where we are measuring and assist the primary mission recovery.

3. Probe description

3.1. Introduction

The whole probe fits in our 3D printed case. We decided to name it "Znicz1" because of the shape of our case. Our list of electrical components is now final when it comes to primary mission components and matches our needs and requirements. Every planned sensor and its function is described below.

3.2. Components description

3.2.1. Microcontroller board

We managed to test the CanSat Kit microcontroller board and decided to use it inside the probe due to its big capabilities. It completely fulfills our needs and no board change is needed.

Current status: Tested and ready

3.2.2. Batteries

We chose to go with 2600mAh modules (ICR18650-26FM) because we found a particular seller which would pre-weld them to a tin strip resulting in a 3P configuration of 7800mAh. This is still a massive battery pack that after a few hours of testing communication its voltage went down just 0.04V and will surely last long enough for the whole flight. Probe includes battery charger and protection module - TP4056.

Current status: Tested and ready

3.2.3. Atmospheric pressure and temperature sensor

This sensor fulfills all primary mission data gathering requirements, from received data we are able to calculate altitude of falling Can. The BMP280 provided with the CanSat Kit works just fine through the I^2C interface and we haven't encountered any problems during tests. It’s also working as our main thermometer, because it’s much more stable and calibrated than the LM35. Beside its obvious function, we will use it for getting exact altitude.

Expected results: Change in temperature and pressure gives information about basic air properties and altitude. We will combine temperature with altitude. Also by this we can get information about potential disaster.

Current status: Tested and ready

3.2.4. Particulate matter sensor

The particulate matter sensor is our main focus in the secondary mission. Basing on data collected from this sensor it is possible to examine life conditions. We opted for the SPS30 which provides us high sampling rate of 1Hz that erases the need for slower descension.

Expected results: We expect some air pollution during final competition. It is really hard to predict the values of the pollution, but we know that on significant heights air pollution is low.

Current status: Tested and ready

3.2.5. Humidity sensor

Air humidity values would be used in secondary mission to predict possible fires. Very low humidity is often main cause of those disasters.
We found DHT22 sensor most suitable for our needs, it combines high precision and fast update compared to other humidity sensors.

Expected results: We expect change in humidity due to different weather. We will also analyze it in terms of altitude.

Current status: Tested and ready

3.2.6. GPS module

We decided to buy the Adafruit Ultimate GPS module based on MTK3339 chip. It was more expensive than our initial choice but its capabilities and big community support made it a best option. It also provided us with a RTC powered by a backup battery. We use it not only to get the exact location but also to get the altitude. It's the second source of this variable, by that we can check our data and make sure that it's correct.

Expected results: From our GPS module we will get exact position and altitude. We restricted expected values of longitude, latitude and altitude in order to get only correct values.

Current status: Tested and ready

3.2.7. IMU module

Accelerometer readings from this module could be used to predict earthquakes and avalanches.
Additionally IMU module will give us information about Can momentum additionally it is very useful during tests. We plan to use it when determining landing spot.
With use of special library we managed to get it working as intended. Kalman filters and Magdwick algorithm (inside DMP) used in the library give us very accurate data.

Expected results: The IMU is currently used as a backup in calculating a wind vector and we expect to see at least the full flight recorded in 9 axis.

Current status: Tested and ready

3.2.8. Gases sensor

Sensor responsible for this functionality is the MQ-9 module. It is sensitive for carbon monoxide (CO) as well as flammable gases. We predict that we won’t exceed the alarm levels of these gases during the mission, but this sensor is a supplement for the theoretical aspects of our secondary mission.

Expected results: Probably we won’t exceed the alarm levels of flammable gases or carbon monoxide concentration, we will get information about it live.

Current status: Tested and ready

3.2.9. Buzzer

The buzzer was implemented on our first PCB with a transistor as to not overload the digital pins of the microcontroller. It's working just as expected, helping us to locate it after landing(It's hearable from about 7-10m) and signal that the Can is working.

Expected results: We hope it will help finding our satellite after landing. It's hearable from about 7-10m.

Current status: Tested and ready

3.2.10. Photoresistor

We can get information about CanSat mission stage in a very tricky way, thats by implementing photoresistor. When probe is outside the rocket it detects big amounts of light, if is inside a tray and is covered, no light shines onto photoresistor. We proved this concept during tests, Cansat was elevated inside 'Pringles' can and ejected with special mechanism attached to drone.

Expected results: We expect that photoresistor will give us information about our can. We will know is it inside or outside the rocket.

Current status: Tested and ready

3.3. Measurements

Table below presents data that will be gathered during the mission. **Minimum** and **maximum** columns represent predicted values to be received. Readings precision is presented in **Significant figures** column with format: *[before ; after decimal point]* in base 10.

Sensor	**Reading variables**	**Minimum**	**Maximum**	**Significant figures**	**Sampling time [ms]**
BMP280 (I^2C)	Pressure	600 *hPa*	1100 *hPa*	4 ; 1	40
	Temperature	-10*°C*	40*°C*	2 ; 1	40

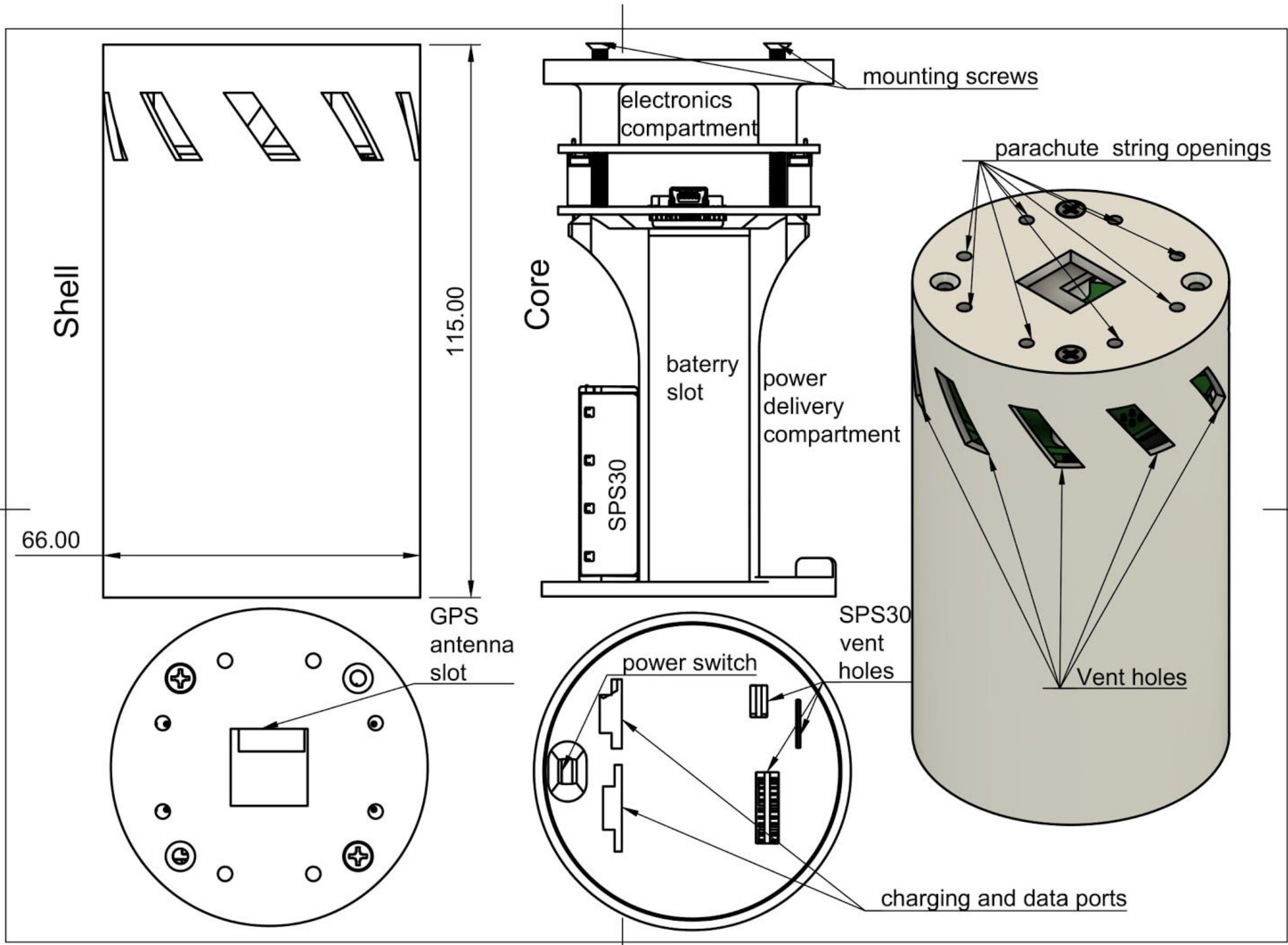
Adafruit Ultimate GPS (UART)	Latitude[1]	49,00°	52,00°	2 ; 7	1000
	Longitude[1]	14,07°	24,09°	2 ; 7	1000
	Altitude	0 *m*	3000 *m*	4 ; 1	1000
	Clock	mission start time	mission end time	-	1000
DHT22 (Digital communication)	Humidity	0 %	100 %	2 ; 0	2000
	Temperature	-10°*C*	30°*C*	2 ; 1	2000
SPS30 (UART)	Mass concentration	0.3 $\mu g/m^3$	100 $\mu g/m^3$	2 ; 1	1000
	Number Concentration	0.3 $1/cm^3$	10 $1/cm^3$	1 ; 1	1000
MPU 9250 (I^2C)	Tilt	0°	360°	3 ; 2	20
	Acceleration	0 *g*	4 *g*	1 ; 2	20
	Magnetic orientation	0°	360°	3 ; 0	30
Photoresistor (Analog)	Light intensity	0 *lux*	10 *lux*	1 ; 2	10
MQ 9 (Analog)	Flammable gases and carbon monoxide	0	1024	4; 0	10

[1]Minimum and maximum values of GPS location are set to Poland boundaries. During the real mission they would not reach further than few kilometres from start. This allows us to compress data more effectively

3.4. Recovery system

As described in 3.5 and 3.5.3 we are using a parachute made by us and then calculated and tested for falling with a speed of 9m/s. It's attached to the body with 8 strings with a holding strength of 60kg. They are attached by pinching them through 8 holes and tying a knot. So far our test have proven this design successful.
The main power switch is located at the bottom and is a simple bistable slide switch. (See 3.5)

3.5. Mechanical design



Open link to the full assembly: https://a360.co/2CFqlhm

3.5.1. Weight

Our full assembly was weighed firstly on a kitchen scale then in our school chemistry lab and both reported it being safely under 350g, about 336g.

This picture shows the final weigh without the MQ-9, but its weight is only about 5g.

3.5.2. Case

The Case is made of a ABS based material called Z-UltraT and 3D printed in our school. The iterative design lead us to a very efficient and strong assembly, which is held together by 2 M3 steel screws. The center of mass is located fairly low which helps stabilize everything during flight. During our tests, the design withheld falling with a tangled parachute(v_{max}~15m/s) without any structural or electrical damage.

3.5.3. Parachute

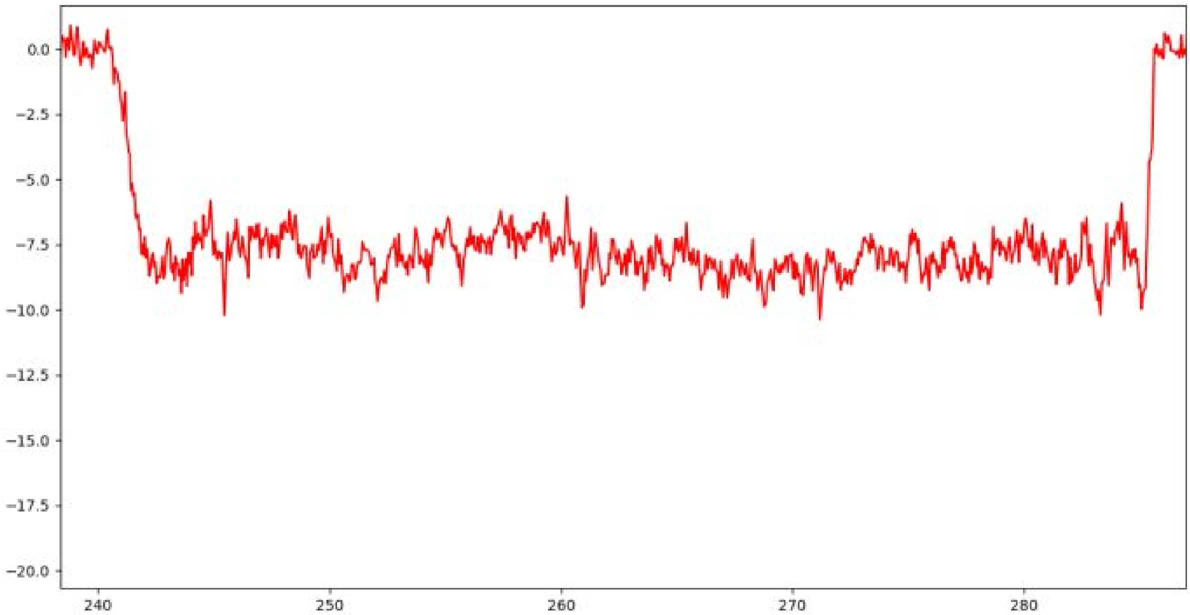
Our first real parachute prototype made out of Skytex 27 Classic had a radius of 20cm with an 8 cm in diameter center hole. This was an overestimation(theoretical r=18 cm, C_d=0.78) for test purposes and resulted in v_{max}=5.5m/s.

$$r = \sqrt{\frac{2 \cdot 9,81\frac{m}{s^2} \cdot 0,35kg}{\pi \cdot C_d \cdot 1,28\frac{kg}{m^3} \cdot \left(8\frac{m}{s}\right)^2}}$$

After adjusting the drag coefficient according to our test flights (C_d=1.26)

$$C_d = \frac{2 \cdot 9.81\frac{m}{s^2} \cdot 0.35kg}{1.28\frac{kg}{m^3} \cdot \left(6\frac{m}{s}\right)^2 \cdot \left(\pi \cdot (0.2m)^2 - \pi \cdot (0.04m)^2\right)}$$

the parachute was modified to a radius of still 20 cm but a center hole with a diameter of about 13cm. Testing the new version resulted in safe v_{max}=9m/s which meets the required range of 8 to 11 m/s.



This graph, generated from data gathered during drone mission, represents speed calculated between 2 heights of 10 pressure readings apart, in relation to time. The distortion is caused by a low distance between readings in the calculation program, but shows the overall speed more accurately.(it’s showing a negative value, because it’s calculated by a decreasing height)

3.5.4. Transmitting antenna

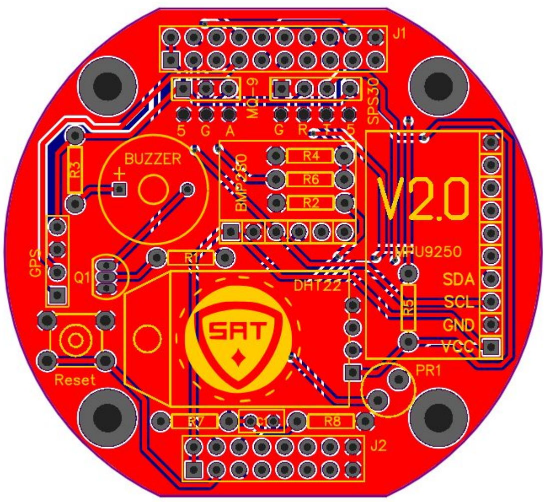
For the transmitting antenna on the can we used a 17 cm long solid core wire soldered directly into the microcontroller board. So far it suited its purpose but in case of it being not enough we can replace it with a manufactured one.

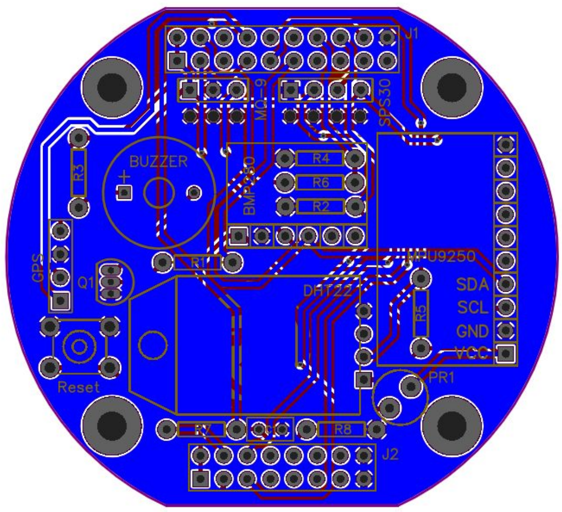
3.6. Electrical design

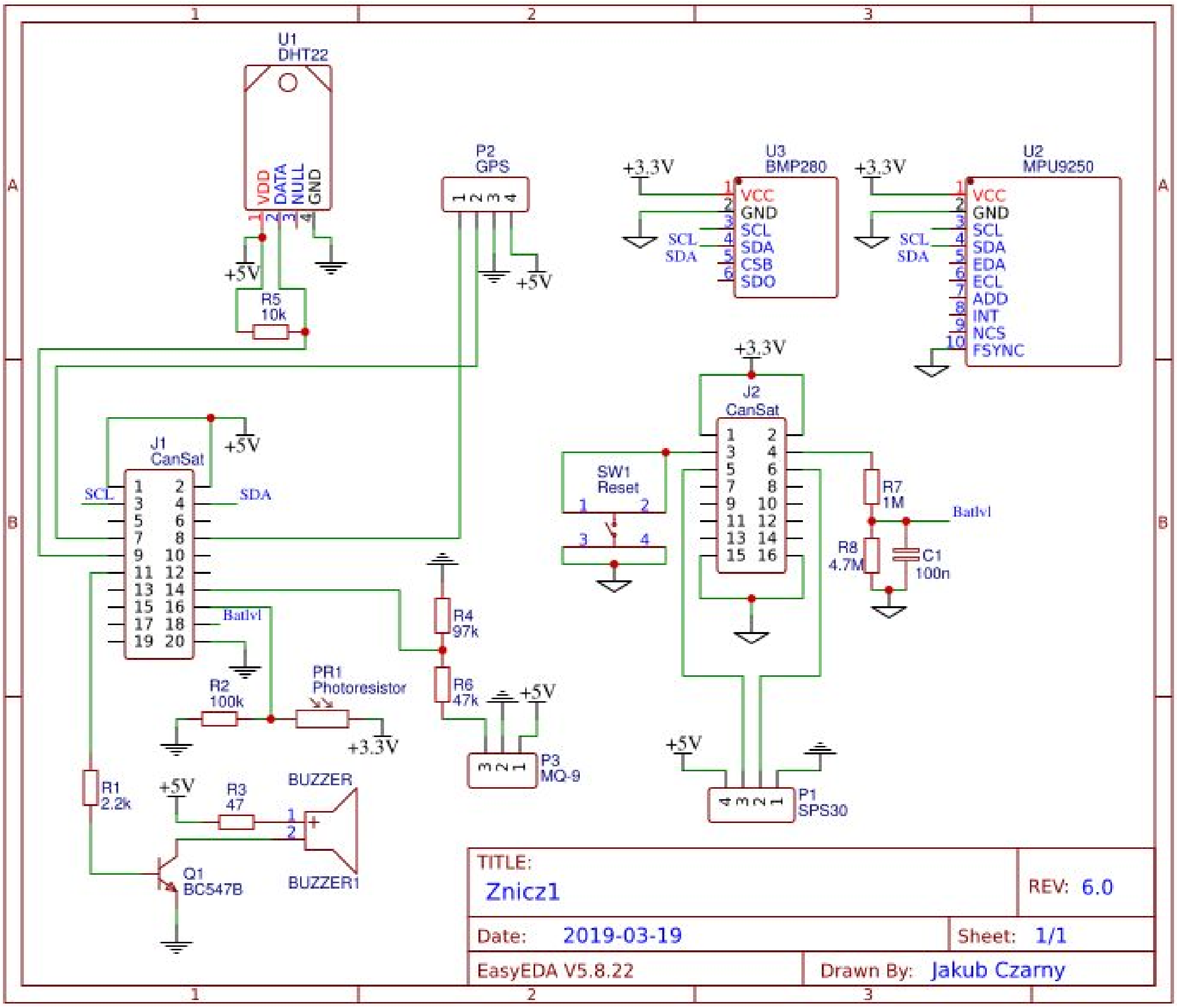
3.6.1. Daughterboard design

We designed and ordered a printed circuit board for all of our main sensors and components as a shield for the CanSat kit. As it was made in a hurry so it had some bugs, but after soldering a jumper wire to connect the buzzer transistor to a correct port it was fully functional and behaved as expected in all of our tests. After adding a new sensor to help our secondary mission and additional functionality like battery voltage monitoring, reset switch, stress relief holes and better spatial arrangement we ordered a new pcb and are expecting it to arrive shortly. It's important to note that the new PCB has retained its core functionality with the old one and even in case the added features not working, it should behave completely the same in regards to our tested design.

New PCB schematic and design:







3.6.2. Power consumption

Here is the theoretical maximum power consumption e.g. worst case scenario.

Component	**Maximum current draw [mA]**	**Voltage [V]**	**Efficiency[2] [%]**	**Effective power [mW]**
Microcontroller	20[1]	3.3	50	132
SX1278	120	3.3	50	792
BMP280	1	3.3	80	7
SPS30	80	5	80	500
DHT22	1.5	5	80	9,4
Adafruit Ultimate GPS	25	5	80	156,3

MPU9250	3.7	3.3	50	24,4
Buzzer	20	5	80	125
MQ-9	150	5	80	937,5
Total	**421,2**	-	-	**2683,6**
Photoresistor is drawing an unnoticeable amount of current so we decided to skip it in the calculations				

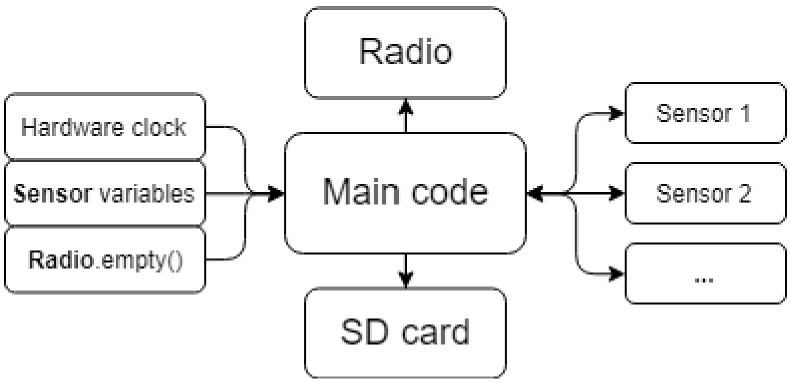
[1] - Here we used the average values provided in the CanSat Kit documentation, because we think they are more suitable and we couldn't find the maximum power draw for the microcontroller.
[2] - Efficiency is based on which power source is used by the component and it's either 80%(Efficiency of the step-up converter) for +5V or ~50%(80%*66% - combined efficiency of the step-up converter and line regulator) for +3.3V

Battery capacity = 2600mAh*3*3,7V = 28860 mWh
Total power consumption = 2683,6 mW
Theoretical maximum operational time = ~10,8

We do not have an onboard battery voltage meter yet so we couldn't make a continuous battery drain test, but since getting our batteries we only charged them once to full capacity and after using them for all tests during the last 2 months the voltage dropped to only about 3.9V making them safe for the required 6 hours of ontime.

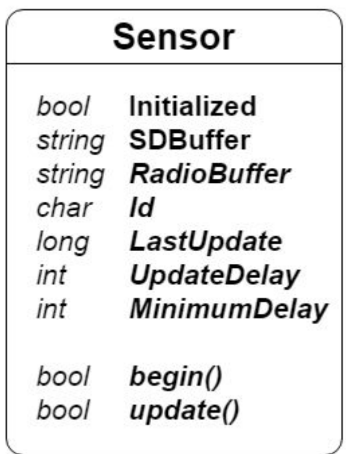
3.7. Software design

3.7.1. System structure



3.7.2.

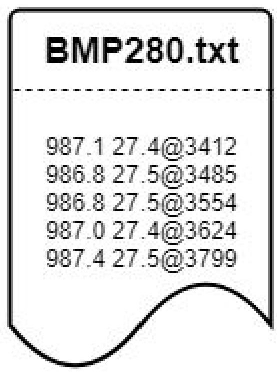
Whole functionality of all the components included inside probe is contained inside separate library. As a result main code uses only objects of probes components and their basic methods like: *begin, update etc.*



For every sensor inside probe, there is a class with defined data reading methods that inherits from abstract **Sensor** class.
Sensor class consists of initialization and update methods, radio and SD buffers, diagnosis variables and update time. Software structure like this allows to use every sensor in unified way without worrying about it's other methods required for proper working.

3.7.3. Data gathering

Every sensor has different specifications and by that every has its own update time, to not lose data while waiting for those with low sampling rate, sensors are invoked asynchronously when their update time expires. With every measurement sensors input new reading into their *SD buffer* and every reading is marked with hardware clock time. For example *BMP280* saves pressure and temperature floats and *GPS* saves last received *NMEA* sentence. Methods responsible for data gathering are different for every one of them but their outputs are later unified to fit **Sensor** base.
Every session of CanSat being active is saved to different directory on SD card. Inside those directories there are files belonging to every sensor. *Example file is presented on image on the left.*



3.7.4. Event logging and timers

Our CanSat software implements functions responsible for logging too. Every important action inside probe will be saved with time marker on *.txt* log file.
By **time marker** (number after '@' char) in CanSat we understand value of *millis()* function that returns amount of milliseconds elapsed since board start counted by the hardware clock. Additionally software gets current time from *RTC* module inside *GPS* board. At the beginning of log file there is an information about real time and time marker during this moment.
Example heading of a log file: *#110119161503412@1302*
This string lets us know exact day, month, etc.. and *time marker* during this moment. Every action after this line will be marked only with time marker because we can simply add difference of elapsed milliseconds to get event time.

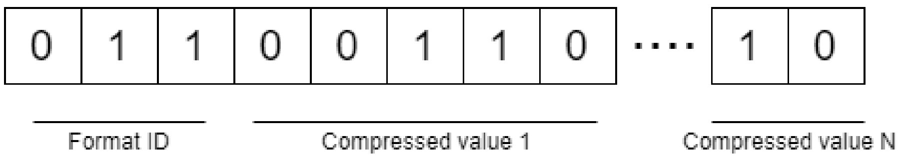
3.7.5. Data compression during radio transmission

Probe software implements simple yet effective data compression. In connection with optimal LoRa variable settings we are able to transmit real-time readings with sending frequency of 5,3 transmissions per minute (*look at: drone tests 3 & 4*) and could be freely changed to any value above by sending queued data (but in reality only pressure sensor is able to update so fast). Our compression bases on mapping to known region of numbers and transmitting only changed variables.

Measurement values collected from sensors would be mapped to some binary number with length specified by reading precision. For example knowing that temperature is in range of -10 to 30°*C* and reading precision is: two digits before and one after decimal point, we can map value from 0 as -10.0°*C* up to 400 as 30.0°*C*. Numbers in that range can be represented as binary number with length of 9 bits. During compression a data format is created.
Example data format (must be identical for receiver and transmitter):

```
1 0_26_Latitude_49.0000000_54.5000000_2_7
26_53_Longitude_14.0699997_24.0900002_2_7 53_67_Altitude_0.0_1000.0_4_1
67_90_Pressure_600.0000_1200.0000_4_4 90_103_Temperature_-20.00_50.00_2_2
103_113_PM25_0.0_100.0_3_1 113_123_PM100_0.0_100.0_3_1 123_128_PackNo_0_32_2_0 X
```

Example data send:



Later received data will be decompressed on ground station and presented in human-readable form. Our first tests of compression during real mission achieved shortening packet from 54 Bytes (chars) to only 16 Bytes. Algorithm still can be improved but for current phase we have already reached the point where not many sensors could be updated before there is time to send another packet.

4. Ground station

4.1. Radio receiving

Antenna:
Our base station antenna (LPRS YAGI-434A) was ordered from UK and works without any problems. It already was equipped with an SMA connector so no modifications were necessary. Its aluminium construction makes it very light and easy to handle which helps make it handheld. in order to verify the connection with our satellite without having to observe computer screen we redesigned and 3D printed the antena handle adding

small OLED screen that displays the most important data - RSSI, height, etc. After installation it performs exactly as intended enabling operating antena by only one person.

Data received by the receiver is decompressed, saved to SD-Card and outputted to a serial port. Software ran on the computer connected to the receiver via USB will listen for input. Read data will by parsed, displayed and subjected to further analysis. Each input will be also saved by saved by DataManager and logged by EventManager.

GroundStationLog.txt

```
Tue Jan  8 12:34:40 2019 Radio __init__ {'kwargs': {'kwargs': {'baudrate': '115200', 'port': 'COM9', 'timeout': '1',
'kwargs': {'event_manager': <Modules.EventManager.Manager object at 0x0369CA90>}}}}
Tue Jan  8 12:34:40 2019 Radio open_serial {'kwargs': {'kwargs': {'baudrate': '115200', 'port': 'COM9', 'timeout': '1'}}}
Tue Jan  8 12:34:40 2019 Radio readline:call
Tue Jan  8 12:34:40 2019 Radio readline:received {'kwargs': {'kwargs': {'data': b'15_23.13_980.62_50.0684_19.9318_231.6\r\n'}}}
Tue Jan  8 12:34:41 2019 Radio readline:call
Tue Jan  8 12:34:41 2019 Radio readline:received {'kwargs': {'kwargs': {'data': b'14_23.13_980.59_50.0684_19.9318_231.6\r\n'}}}
Tue Jan  8 12:34:42 2019 Radio readline:call
```

4.2. Ground Station Interface

We divided main window of ground station graphical interface into a few groups:

- Information Board display live information from satellite (temperature, humidity, etc)
- Interactive Gyroscope Widget informs about the inclination of satellite relative to the ground by processing data from GPS
- Interactive Map Widget plots current and past location of satellite creating the flight path of our satellite. Each point is colored depending on RSSI and can be clicked to display further information about satellite at that moment. Additional feature we developed is Landing Zone Prediction System, which forecasts landing spot based on current wind vector measured during flight and wind data obtained from internet.
- Group of buttons used to control DataSaver and change information displayed on plots.
- Interactive graphs plotting live data from satellite.

4.3. Landing prediction feature

We succeeded to implement Landing Prediction Zone System which receives GPS data and presents the probable landing spot. We calculate this zone depending on current wind vector, live wind data from internet and terrain elevation. The prediction isn't very accurate (the miscalculation equals approximately 10% of the drop altitude).

We plan to implement landing prediction feature into our ground station software. So far we found linear regression methods as the best candidate. From received *GPS* data we are able to represent the Can's route as a line in three dimensional space. Having known ground altitude we can solve the linear equation and get the predicted landing position. The results definitely would not be very accurate, in the future we plan to also use IMU data providing we find proper methods to use them.

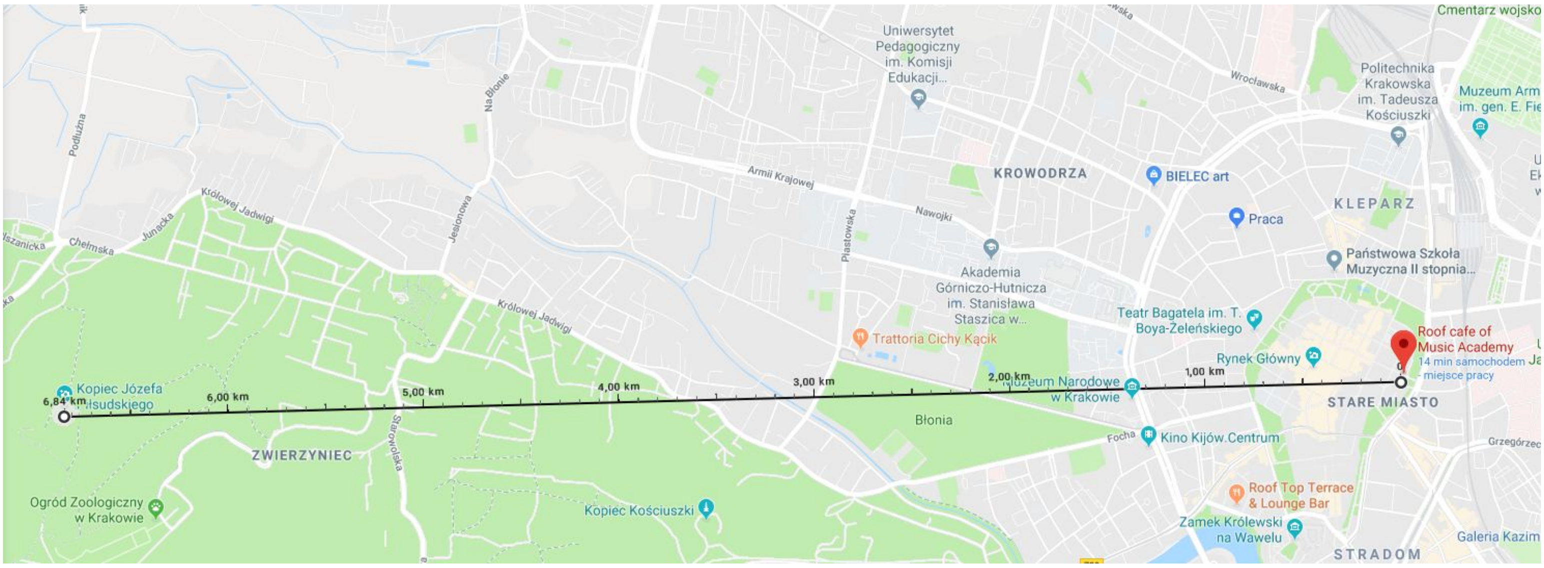
4.4. Data Analyzing

We have finished implementation of analyzing features to our ground station software. The main ideas are that after flight probe can be plugged to the computer for analyzing information saved on SD Card as well as data saved on computer by DataManager. Software will enter Debug Mode in which every measurement will be presented in a window similar to the one with live data, but with feature to reenact flight history in a real time and readout by readout. All information will also be compared and analyzed with regard to irregularities and interferences. After analyzing all plots can be saved as pictures, simplifying the preparation for the results presentation.

5. Test campaign

5.1. Radio transmission tests

- 18 December - 2.8 km of range
- 26 February - 6.8 km of range
  We tested our radio transmission between Piłsudzki's Mound and the Roof cafe of Music Academy in Krakow, as shown below.



  At this distance we measured a max RSSI of -119 and the connection was stable. We were sending uncompressed packets with all gathered sensor data.
- 6 March - first model of compressed transmission allowed us to send same data about 3 times faster as before.

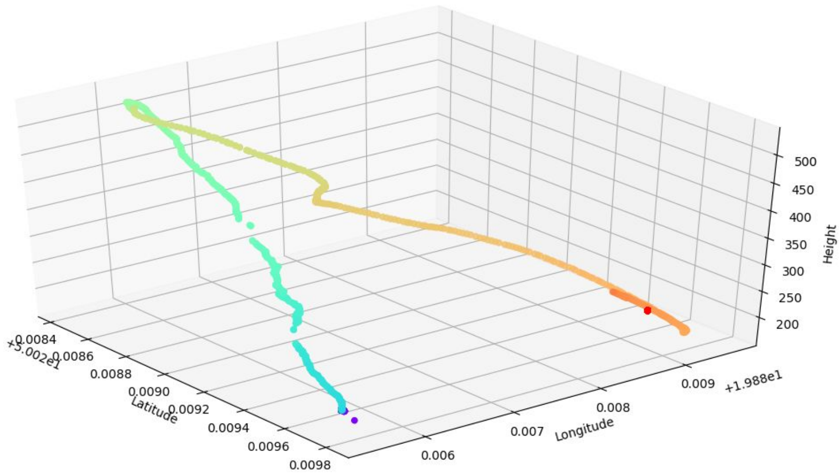
5.2. Drone drop tests

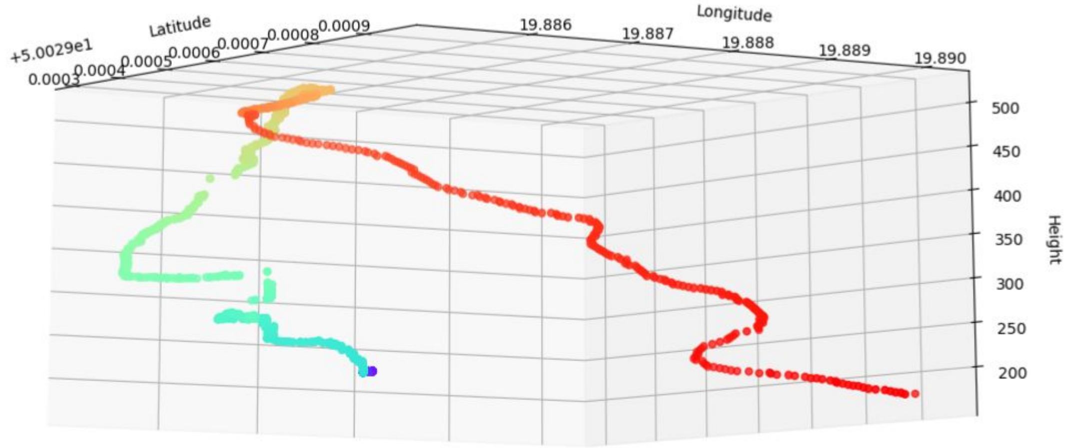
We cooperated with FlyTech to perform our drop tests.

- 11 February - Our first paratuche prototype achieved 5.5 m/s fall speed. Probe was sending pressure, temperature and position data, that's 33 Bytes with about 2.5 transmissions per minute. Unfortunately we were not able to use radio and SD at the same time, board was crashing in random moments. We managed to make 5 drops, one with only radio transmitting and rest with only SD logging.

- 26 February - Our daughter board arrived, so we could implement few other sensors and test new mechanical design. This time probe was sending additionally particulate matter density values. Still problem with crashing wasn't solved, our main goal this day was to test ground station interface so we made 4 flights with only radio transmission. It was the first time when the buzzer was used, because it wasn't making enough noise we managed to lower resistance of safety resistor.

- 6 March - This time SD card and radio finally worked, we build new parachute with about 10 m/s falling speed. New data compression algorithm allowed us to send much more precise GPS data and with higher speed.During the last flight that day our can landed in a small river and we focused on recovery and testing every part. After drying we have found two of our sensors to be broken - SPS30 and DHT22, after cleaning the SPS30 got fixed and only the DHT22 remains to be replaced.





Two of our drops during this day, blue line shows drone ascension and red one is fall.

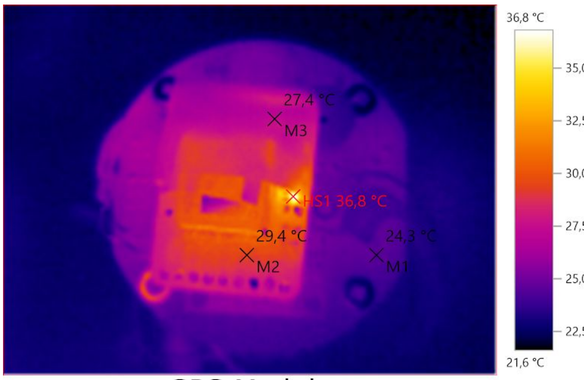
- 22 March - Water damage recheck, final software testing. During this drop we were testing the final software and making adjustments. We also discovered inconsistencies with our GPS height readings so to be safe, we’ll replace the module and switch to the barometric formula as our primary height calculation.

5.3. Thermovision tests - heat visualisation

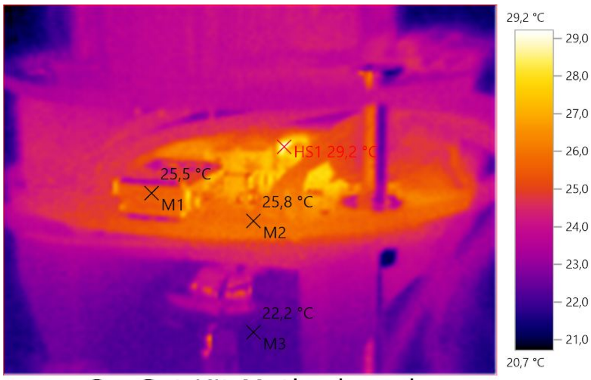
We decided to check our satellite in terms of heat emission. It is important factor because heat emission can disturb measurements like temperature or humidity. We checked all modules which can emit heat. The measurements were made with a professional thermal imaging camera. Our probe was running for over one hour before taking these thermal photos but we were also checking its state during this hour. The environment temperature is about 22 °C, so it was initial for all sensors. The temperature of the most important spots after one hour is presented in table below. (+ fotki)

Module	Average temp.	Hottest point
Adafruit Ultimate GPS	28,1 °C	36,8 °C
CanSat Kit Motherboard	25,6 °C	29,2 °C
SPS30	23,7 °C	24,8 °C
Batteries	22,2 °C	22,4 °C

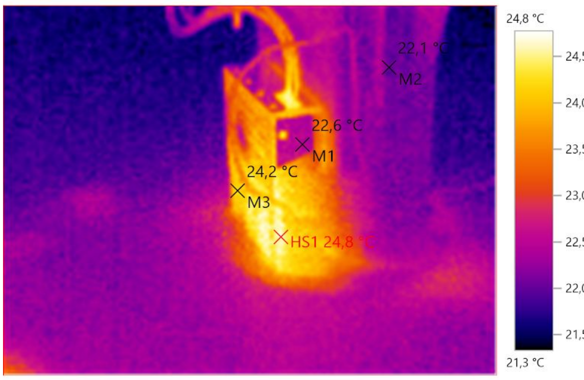
We also present some images from the camera. All sensors were visible to the camera - cylindrical cover was taken down.



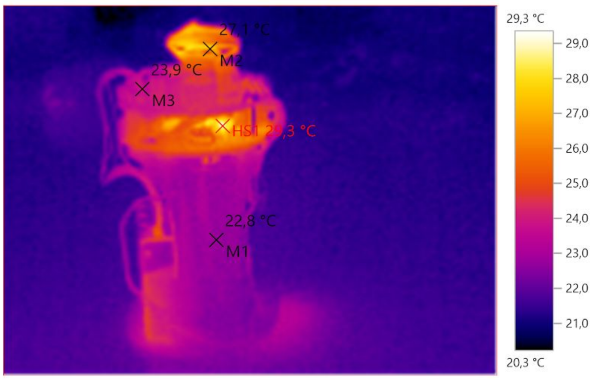
GPS Module



CanSat Kit Motherboard



SPS30 Module



Whole satellite without cover

6. Project summary

6.1. Resource management

6.1.1. Budget and costs

Note: All prices in this section to the 8th point are converted from PLN to EUR with exchange rate 1 PLN = 0.232996887 EUR (05.01.2018 - Google)

Note: All further prices in this section are converted from PLN to EUR with exchange rate 1 PLN = 0.232503288 EUR (22.03.2019 - Google)

We have started with 267,95 EUR. We hope we will get more money and partners. Now we are on a good way to get a sponsorship from Kraków Airport. On this day (22.03.2019) our spending is presented in the table below.

No.	Our expenses	Price (EUR)	Category
1.	website hosting	4,66	Promotion
2.	Facebook advertisement	5,82	Promotion
3.	2x Charger TP5056, 2x Buzzer, 1x DHT22 and shipment	7,47	Modules
4.	1x MPU9250 sensor, 3x batteries, welding service and shipment	21,64	Modules
5.	1x Adafruit Ultimate GPS and shipment	38,09	Modules
6.	Antenna and shipment	53,42	Modules
7.	Electrical tape, sandpaper	1,05	Modules
8.	Food, snacks, drinks for meetings	31,43	Meetings
9.	Stickers	6,97	Promotion
10.	First PCB, screws and M3 rod, OLED screen and shipment	32,63	Modules
11.	Sockets, Nanoprotec spray, MQ-9 sensor and shipment	23,63	Modules
12.	New PCB (planned to insert, in shipment)	12,24	Modules

	SUM	**239.05**	

Now we have 267,95 - 239,05 = **28.9 EUR**. Our main planned expense is a new GPS sensor, in case of a possible breakdown of the prior one during the finals. The next thing is the eventual cost of the participation in the final for the fifth member of a team. We are clear to sign the contract with Cracow Airport, so we will possibly get some additional financial resources for our project from this source.

6.1.2. CanSat Value

Note: All prices in this section to the 9th point are converted from PLN to EUR with exchange rate 1 PLN = 0.232996887 EUR (05.01.2018 - Google)

Note: All prices shown further are converted from PLN to EUR with exchange rate 1 PLN = 0.232503288 EUR (22.03.2019 - Google)

The whole value of our CanSat is presented in the table below.

No.	CanSat part	Value (EUR)
1.	Adafruit Ultimate GPS	34,92
2.	Batteries and welding service	11,87
3.	MPU9250	7,69
4.	DHT22	3,69
5.	Buzzer	0,47
6.	Li-Ion Charger	1,21
7.	SPS30	40,93
8.	Case - 80g Z-ULTRAT material	4,66
9.	Parachute and cords	5,00
10.	PCB	0,17
12.	Screws, M3 rod	0,55
13.	MQ-9 sensor	6,89
14	USB Socket	0,45
	SUM	**118,50**

6.1.3. Sponsors

Note: All prices in this section are converted from PLN to EUR with exchange rate 1 PLN = 0.232996887 EUR (05.01.2018 - Google)

We wanted to amp up our chances for getting a sponsor so we have written to the following companies:
Comarch, Asseco, Cd projekt, Orange, Play, Coca cola, Pepsi, Rmf fm, Ailleron, Shell, Orlen, Lotos, Arge, Tvn, Radio zet, Tok fm, Kraków Airport, Codewise, Flytech UAV, Kraksky, botland.pl, School parental council, Company which want to be anonymous.

We have several sources of money as well as partnerships. They are presented in the tables below.

No.	Source of money	Amount (EUR)
1.	School parental council	186,40
2.	Team contribution	58,25
3.	Team advisor contribution	23,30
	SUM	**267,95**

No.	Partnership	Our benefits
1.	Our school	We have unlimited access to our school's 3D printing lab
2.	Botland.pl	We got a 7% discount on all their items
3.	Company which wants to be anonymous	We got parachute material and parachute cords - The material is called DOKDO-N20DMF-WR-HD38
4.	Kraków Airport	We are clear to sign the contract with the airport. After signing it we will get 800 PLN. The contract has been accepted by both sides - our school and Kraków Airport.

6.2. Documentation

6.2.1. Social media & promotion

6.2.1.1. Facebook

We have over 400 likes on our Facebook account (linked below). Posting regularly about 35 posts, we have on average 400 views under every single one. We have also several posts with about 1000 views, and also one, promoted, with nearly 8000. This post is a movie-type which is a vlog from our first radio transmission tests. By doing this, we wanted to promote our project as well as the whole competition among the local and school community. Our facebook profile contains valuable educational information and presents the whole timeline and advance of our project. Recently we also published a very special movie - our advertisement. Our predictions say that it will be the most viewed post ever on our profile - 3 days after release it has over 8000 views.

6.2.1.2. Radio promotion

We have also written emails to local radio stations. Many of them replied and invited us for short broadcasts. During these conversations we were talking about our project and its educational values. Table below shows radios we have already visited and our plans for future.

Radio name	Status
Radio Kraków	**Visited 03/12/2019**
Radio Eska	**Visited 03/18/2019**
Plus Radio	**Visited 03/18/2019**
Radio Wawa	**Visited 03/18/2019**
Antyradio	**Visited 03/21/2019**
Melo Radio	**Visited 03/21/2019**
UJOT.FM	**Planned live on 03/25/2019**
Radio Pryzmat	**Planned live on 04/10/2019**

6.2.1.3. Articles

We have also written many articles for magazines and newspapers. Our main target is to show that technology is interesting and encourage people to follow our project. Table below shows magazines we have already written for.

Magazine/Newspaper	Status
Life In Kraków	**Published**
PC World	**Published**
Gazeta Krakowska	**In progress**
AntyWeb	**In progress**
21. Wiek	**In progress**

6.2.1.4. Summary

Due to the project promotion we got huge hype. Beside these actions, we promoted ourselves on our schools facebook group which has over 3000 members as well as school webpage with over 2000 daily entries.

Facebook site:
https://www.facebook.com/SobieskiSat/

6.2.2. Education

We also plan to make some workshops for other students from our school. We are going to make a presentation about our project and present how our satellite is working. These workshops will probably take place during the next week after completing FDR. Probably we will also have a chance to speak in front of the students applying for our highschool during the "open day". We plan to show basics of our project as well as introduce students to the CanSat world.

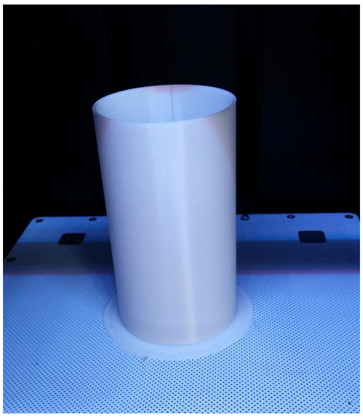
6.2.3. GitHub Repository

We decided to create a public organization in github, so that everyone interested in our project could follow our progress. You can find there the entire code of the satellite and ground station. We also include all 3D models, diagrams and documents of our project. This helps to organize and archive all of our work, enabling easy tracking. Satellite, Antenna and ground station softwares are updated there regularly.

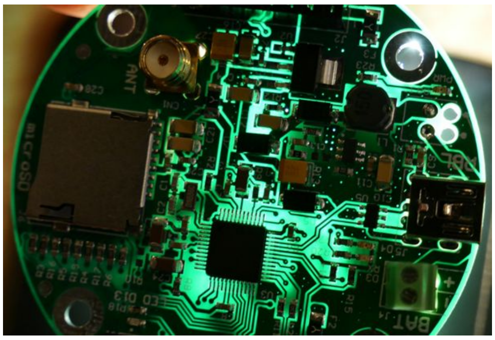
Our GitHub: https://github.com/SobieskiSat

6.2.4. Timeline

26 September – The date of one of our first actions taken to develop our micro-satellite. This day we finally got access to the school 3D lab, where we printed the first case to see exactly how much space we do have to fit in all components. We have also mentioned this action in PDR, as one of the Project Summary.

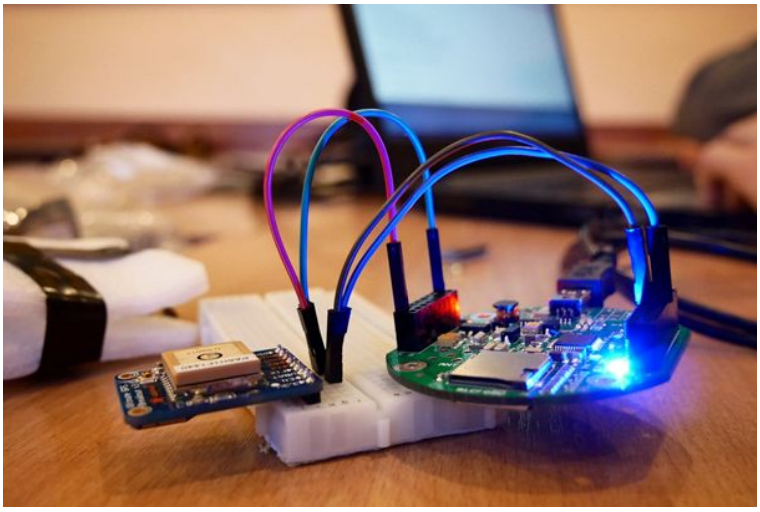


30 September – that day we met with the airplane pilot, who helped us to design and plan a lot of thangs attached to aerodynamics and electronic circuits. We elaborated a project of components composition inside the case, spoke a lot about secondary mission idea and chose the way which our marketing and advertising actions will follow in the future.



20 November – our first meeting during which we were testing modules we had received a couple days before. We have tested components such as accelerometer, gyroscope and magnetometer, simultaneously with other essential modules needed to complete the primary mission. All gages of these we have done on a breadboard, but after that we still needed to wait for missing parts.

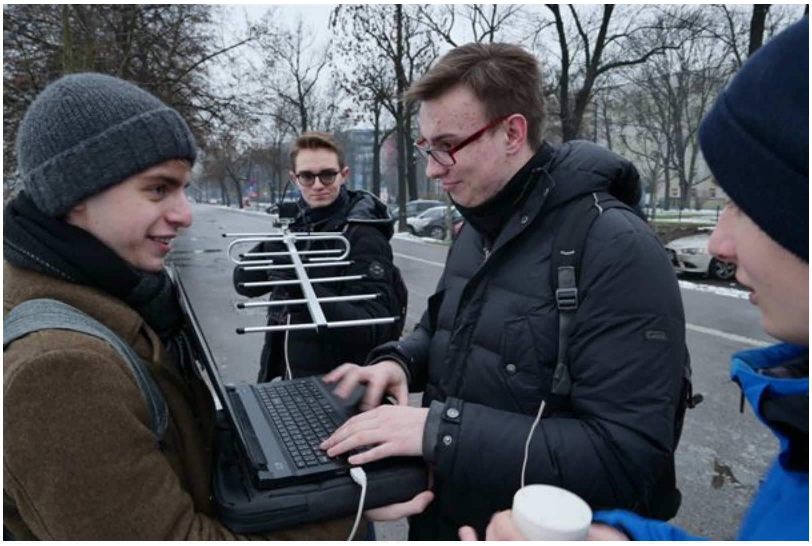
27 November – that was our second meeting during which we worked with modules, provided to us a few days after the first meeting. We finished the attunement of the GPS module, and we plugged in the same time every other module to see if it will work correctly – the only one missing was particulate matter module, which we planned to add and trial at one of our future meetings.



11 December – Our measures to design a first prototype were over. We printed a new case for our modules, this time it was more durable and contained a top, so that we could close components up inside. We fitted in all modules needed to complete primary mission and some from secondary and wrote a code to be able to send data from the can and receive it in our ground station, so the first prototype was finally ready to try.

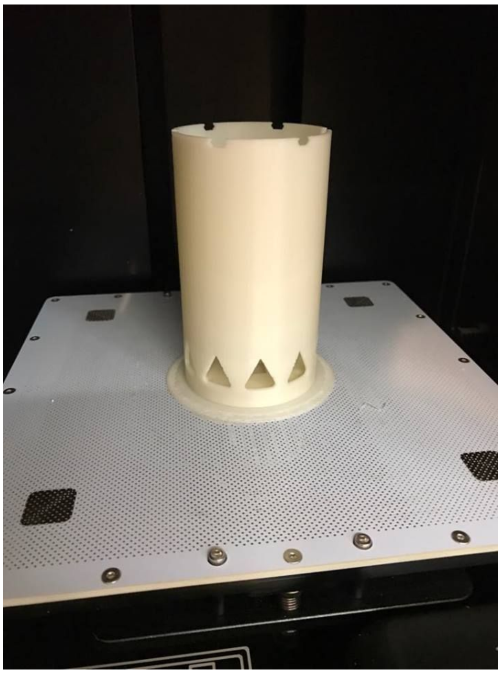
13 December – we got an high-range aerial ordered a week before, that we needed to receive data from our can and relay it further to the ground station. We did a few adjustments in our code to meet the aerial requirements and make everything work correctly, and decided that we need to probe the real aerial range to check if the signal will work during the final competition.

18 December – The day of trials. We were testing our aerial range for about 4 hours, by leaving the ground station in the northernmost part of Błonia, and walking away as far as we could with our sender. We have reached 2,8 km distance, without a big loss of signal, hence we were certain that out satellite will be transmitting data all the time during the plummeting at the finals. That day we have also completed our stab at cansat falling with parachute. Full report of our efforts was posted on our facebook page as a vlog and animation of our repositioning.



23 December – The first usable prototype of the parachute was done. All measurements and calculations had been completed, and afterwards the plastic foil was cut and formed.
Work on the code was also going forward that day.

8 January – We have 3D printed a new version of the case with some improvements, for example holes to provide better air-circulation and new latch system to test new ways of joining the two halves. We have also thrown it with a parachute out of a balcony to check if it will fall properly without destroying itself.

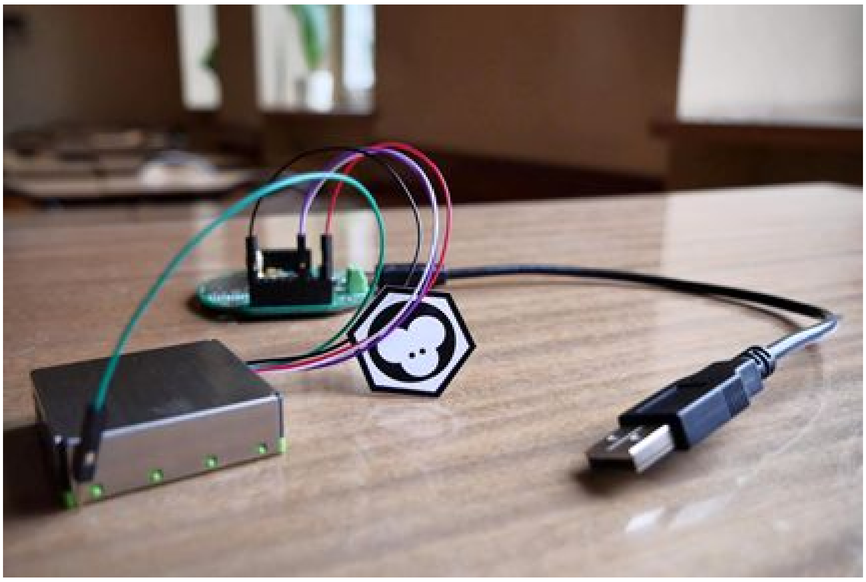
 

11 January – Parachute material was finally with us, provided by the company which wants to stay anonymous – we got over six times more material than we wanted, which enabled us to make multiple versions and test each one separately. We've also set up our Github page.

29 January – We had an opportunity to test our parachute for the first time. Before we could do it, we had had to give a parachute material to the cobbler. We needed to conform and cut cleanly our material. Scissors wasn't enough, because we wanted to do that as precisely as we could.

We threw it from the 4th floor of a residence hall of one of our team members.

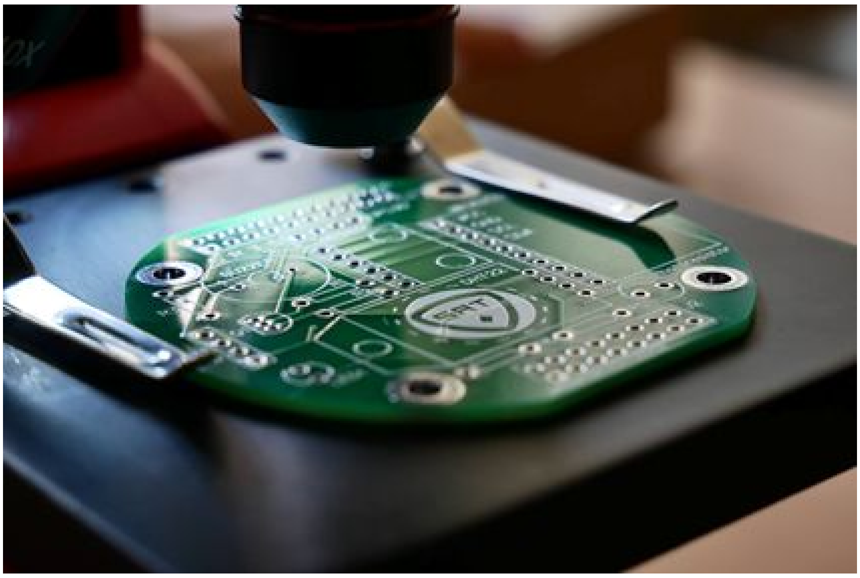
5 February – We got a particulate matter (PM) meter from our friendly cansat group from Cracow – Plastic Monkeys! We checked if everything works correctly, and then we fitted it to our satellite.



11 February – We tested our satellite by throwing it from a drone with full number of sensors inside for the first time. We had some difficulties with the parachute, what caused one fall without any air resistance. However, it turned out our unit is almost unbreakable, because there weren't any grooves on a casing. During these tests we had a still radio connection with our satellite and we were constantly receiving data. We were also saving readings gathered by all of the componentry on a

SD-card inserted into the unit. All that we could achieve due to a courtesy of the FlyTech UAV company.

18 February – After a long time of a delivery from China and an adventure with a lost package, we finally got our self-designed, printed circuit board! It solved a problem of ubiquitous wires, which furthermore have their weight. It was fitting in perfectly in our casing, so we attached all the components to it as soon as we could, and prepared our satellite for further drops from the drone.



26 February – One of the most important days for our project. It was the second time when we were dropping satellite from the drone.

However, that day, we elevated our piece of work to over 300 (before, it was only 170) meters! We were simultaneously logging and live-displaying the readings gathered by sensors working inside our satellite. Everything was working correctly, the only things we had to do were a few software adjustments. Later that day, we eventually examined the signal of our aerial – we divided ourselves into two groups, one travelled to the Piłsudski’s mound with the satellite, and the other went on the roof of Cracow Music Academy. The distance between these two areas totals 7 km, but transmitting was working perfectly. We were almost sure, that our unit is in the final stage of production. Moreover, a part of this day was shown in the second Vlog!

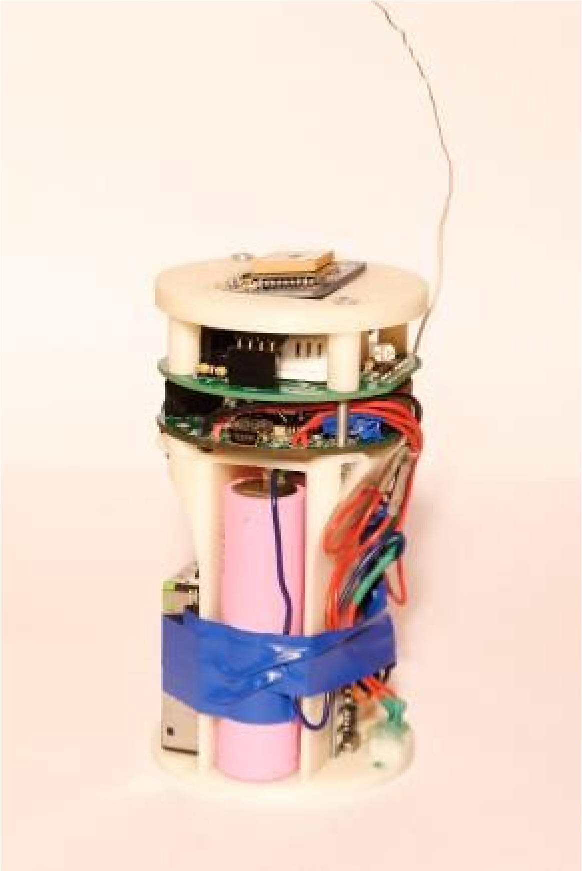
6 March – The third day of satellite drone test. We were working on the landing prediction feature we want to implement, as it was said in the secondary mission description, but we weren’t lucky – during the trials our satellite fell into a river, which has only one meter in width. We were terrified, but it turned out that our gear is also waterproof – just a few of the components were destroyed!

10 March – We started considering about promoting ourselves in the traditional media, not only on a facebook page. We wrote a lot of e-mails to the most of radio stations in Cracow, and also to internet newspapers. We got a lot of positive answers inviting us to take part in radio broadcasts, as well as propositions of writing articles for the newspapers, such as PC World and LifeInKraków.

14 March – The day of recording a short film, which we want to promote and reach as many people as we can. We recorded brief interviews and then cut a movie – We posted and promoted it on Monday. For now, it has about 11.000 of reach and 4.000 views.

Cansat Characteristics

Characteristics	Figure
Height of the CanSat	115mm
Diameter of the CanSat	66mm
Mass of the CanSat	341g
Estimated descent rate	9m/s
Radio transmitter model and frequency band	CanSat Kit SX1278
Estimated time on battery	~10h
Cost of the CanSat	118,50 €



CanSat layout without cylindrical cover