Development of the Research Nano-Satellite RauSAT 2.0

Institution: Nazarbayev Intellectual School of Science and Mathematics, Kyzylorda

Author: Nadira Atymtay

Grade: 11

City: Kyzylorda

Year: 2025

Table of Contents

I. Abstract..3
II. Introduction..3
III. Main..4
1. What is "CubeSat"?..4
2. What is inside a CubeSat and what is its application?.....................................5
3. CubeSat Pitfalls: Space Debris and Data Security..6
IV. Rausat 2.0 Tasks..7
V. Advantages of RauSAT 2.0...7
VI. Technical characteristics of the satellite...8
VII. Development of RauSAT 2.0...9
1. Selection of major components..9
2. Developing a motherboard for the components..13
3. Assembly of the ground station satellite..14
VIII. Conclusion..18
IX. References..18
1. List of used literature..18
2. Application..19
XI. Mentor's review...24

I. Abstract

The research project “RauSAT 2.0,” developed by 11th-grade student Nadira Atymtay from NIS Kyzylorda, aims to enhance environmental monitoring through a CubeSat-standard nanosatellite. The primary objective is to collect real-time environmental data, including illumination, temperature, atmospheric pressure, altitude, and geolocation, while capturing Earth images from space. A hypothesis was formulated that an educationally designed CubeSat could provide cost-effective, reliable, and high-precision data comparable to more expensive satellite solutions.

The research involved several stages: conceptualizing the satellite design, developing a 3D model, assembling components such as sensors and communication modules, integrating a ground station for data transmission, and testing the system’s functionality. The methodology focused on leveraging an Arduino-based system, modular design, and open-source resources to ensure affordability and adaptability. Key experiments included testing data relay via radio waves to a ground station and visualizing outputs on a custom web platform.

The novelty of this research lies in its focus on combining educational accessibility with environmental applications, allowing students to independently design and assemble a fully operational satellite prototype. RauSAT 2.0 successfully demonstrated its capabilities by transmitting precise environmental data and images.

The project’s practical applications extend to monitoring climate patterns, aiding disaster response, and providing communication solutions in remote regions. By merging innovation with sustainability, RauSAT 2.0 serves as a model for advancing Kazakhstan's space technology and education sectors.

I. Introduction

Despite being home to the world’s first cosmodrome, Kazakhstan's space industry remains in its early stages of development, playing a limited role on the global stage. According to statistics, by 2022, the USA had launched approximately 3,500 satellites, China 535, and Russia 170, while Kazakhstan had only launched five satellites [1]. To advance Kazakhstan's space infrastructure and improve environmental monitoring, the scientific project *CubeSat – RauSAT 2.0* has been proposed.

CubeSat is a format of small artificial satellites for space exploration, with dimensions of 10 × 10 × 10 cm. The primary goal of CubeSat development is to offer a highly efficient and affordable means of obtaining space data critical for advancing various scientific fields. CubeSats have several advantages: compact and cost-effective instrumentation, rapid satellite development, high launch accessibility, and reduced space debris [2].

The main functions of CubeSat include monitoring the environment, tracking changes in the Earth's atmosphere, and transmitting data crucial for scientific research.

Objectives of the research project:

1. Development of a 3D model;
2. Enabling data transmission, including illumination, temperature, altitude, pressure, orbital photos, coordinates, and the CubeSat’s position in space;
3. Creating a user-friendly interface for data analysis;
4. Developing a ground station for receiving data from the CubeSat.

III. Main

1. What is "CubeSat"?

CubeSat is a standardized format for nanosatellites, first proposed by Professor Bob Twiggs of Stanford University in 1999. The initial goal was to design a satellite that could fit into a box designed to hold a Beanie Baby plush toy. Today, CubeSats represent an entire industry, and by August 2022, nearly 2,000 such satellites had been launched into Earth orbit [3].

CubeSat belongs to a subcategory of small artificial satellites classified by weight. While small satellites can weigh up to 1 ton, CubeSats have strictly defined dimensions and mass:

- A 1U CubeSat has dimensions of 10×10×11.35 cm and a mass of up to 1.33 kg.
- Larger variants include 1.5U, 2U, 3U, 6U, and 12U, which are combinations of multiple 1U units.

The advantages of CubeSats lie in their standardized size and weight, which significantly reduce development and launch costs. Standardization simplifies the mass production of components, streamlines satellite assembly, and lowers transportation and launch expenses. Multiple CubeSats can work together as a “single organism,” enabling the creation of complex systems for tasks such as coordinated space exploration, atmospheric monitoring, Earth observation, and signal relay in remote areas.In 2009, a smaller version of CubeSat, called the PocketQube, was introduced. This format measures 5×5×5 cm and weighs around 250 g, but the number of PocketQubes remains limited.

Building a CubeSat from scratch can be costly, but most projects use off-the-shelf components or assembly kits, which reduces development expenses. A ready-made 1U CubeSat typically costs under $100,000, while launch costs average around $30,000 per kilogram. This affordability makes CubeSats accessible not only to universities, research labs, and commercial companies but also to individual enthusiasts.

For comparison, launching a satellite weighing up to 200 kg can cost approximately $1 million, with development timelines stretching over several years. As a result, larger satellites often carry outdated equipment by the time they are launched.

In some countries, CubeSats have become the first national satellites. For example, Bhutan successfully launched its CubeSat aboard a SpaceX Falcon 9 rocket in 2018, spending just $280,000 on its creation, launch, and specialist training.

2. What is inside a CubeSat and what is its application?

There are four specific types of aluminum alloy that are used to make CubeSats. Their purpose is to ensure that the coefficient of thermal expansion of the satellite matches that of the launch vehicle. A protective oxide layer covers all surfaces of the satellites to prevent cold welding under high pressure.

Several microcomputers and onboard computers are housed inside the CubeSat. Their function is to perform calculations and control the operation of the rest of the electronics. CubeSats have different missions, which can be classified into four main types:

1. **Technology testing**
   Some CubeSats are used to test new technologies in conditions as close to reality as possible before using them in more complex space projects. For example, testing photographic systems and studying new materials.
2. **Scientific research applications**
   CubeSats collect data to monitor the planet's magnetic field for early detection of earthquakes. Biological experiments are also possible, such as NASA's BioSentinel project, which sent a CubeSat with yeast to the Moon to study the effects of radiation.
3. **Educational projects**
   CubeSats can provide opportunities for students to participate in the development of unmanned spacecraft. Applications include photography, atmospheric and climate studies, and others.
4. **Commercial projects**
   The information obtained by CubeSats can be sold. For example, Planet Labs' SkySat satellites provide photographs and videos for commercial use, such as monitoring agricultural lands and cities.

CubeSats are often equipped with multiple onboard computers to perform various functions, such as attitude control, thrusters, and communications. Microcomputers can perform auxiliary functions, while the main onboard computer delegates tasks to other devices, including attitude control, orbital maneuver calculations, and mission planning. The main computer can also be used to perform payload-related tasks, including image processing, data analysis, and data compression. CubeSats use antennas that operate in the VHF, UHF, L-, S-, C-, or X-bands to provide communications. The available power of the antennas must not exceed 2W due to the small size and limited capabilities of the satellites. Depending on the communication requirements, different types of antennas can be used, such as helical, dipole, monopole, and more complex models.

CubeSat power systems most often combine batteries and means of recharging them. Batteries are small, the weight of which is difficult to minimize. In most cases, solar panels are fixed to the edges of the satellite, with the help of which recharging is carried out. However, there are also deployable panels, which increase the cost and complexity of the design.

There are many different methods of movement in space that CubeSats rely on. One option is chemical propulsion systems, which are associated with chemical reactions and are simple and reliable. Cold gas propulsion is the most useful and safest system; however, these systems offer limited efficiency and do not allow for proper acceleration. There are also electric solutions, but these require additional power and larger solar panels. Solar sails are also used, which have the advantage of not requiring fuel, but add mechanical complexity due to their relatively large size.

3. CubeSat Pitfalls: Space Debris and Data Security

CubeSat launches are typically carried out via ride-sharing on a launch vehicle or delivery to the ISS. Since 2012, the number of nanosatellite launches has increased dramatically, reaching hundreds annually. However, this growth increases the risk of space debris and the Kessler syndrome, which threatens the safety of spacecraft. Although CubeSats typically operate at altitudes of 400–600 km and burn up in the atmosphere, their deployment in orbits above 700 km raises concerns.

Cybersecurity is another issue, as protection against satellite hacking remains weak, despite the lack of confirmed incidents. Nevertheless, cybersecurity technologies and practices are actively evolving, and experts are working on solutions to reduce space debris and enhance

CubeSat protection. These developments open up prospects for the safe and efficient use of small satellites in the future [3].

IV. Rausat 2.0 Tasks

1. Scientific research:

- Astronomy and astrophysics: cubesats to observe space and search for exoplanets.

- Geophysics: monitoring the upper atmosphere.

- Meteorology: collecting climate and weather data.

2. Communications and navigation:

- Testing new methods of navigation.

3. Space geodesy:

- Measuring changes in the Earth's surface: sea level rise and crustal deformation.

4. Education and Training:

- Educational missions: training students and scientists in space technology.

5. Commercial Uses:

- Earth observation: collecting geodata for commercial purposes.

- Communications and Internet: providing communications in remote regions.

6. Environmental research:

- Monitoring of the health of forests, oceans, and the atmosphere.

V. Advantages of RauSAT 2.0

1. Accessibility and low cost: cheaper than large satellites, affordable for universities and companies.

2. Fast preparation: launch is carried out faster, which is important in natural disasters.

3. Flexibility: can be customized for different tasks using sensors.

4. Global coverage: covering the entire Earth in all conditions.

5. Data collection: continuous collection of information for research.

6. Innovation: stimulates the development of space technology.

7. Disaster Prevention: helps to identify disaster threats in advance.

8. Scientific discovery: access to new data for research.

9. Sustainable resource utilization: better management of resources such as water and energy.

10. Environmental friendliness: reduced impact on nature compared to large satellites [4-5].

VI. Technical characteristics of the satellite

The satellite includes:

- 1U frame with protective panels and sensors;

- expandable solar panels;

- electromagnetic coils;

- attitude system controllers and VHF transceivers;

- battery pack and X-band transmitter.

Orbit parameters:

- Altitude: 550 km [6].

- Type: sun-synchronous

- Inclination: 98 degrees

- Ellipticity: circular

- Time of equatorial transit: 10:30

- Full revolution time: 96 min

Operating time:

The expected lifetime of RauSAT 2.0 is 1-2 years.

VII. Development of RauSAT 2.0

1. Selection of major components

RauSAT 2.0 is a nano satellite that will fly in earth orbit. Arduino board was chosen for its development because it is affordable and has a large number of measurement sensors.

The main requirements for building a nano satellite are

1. Development of an on-board computer

2. Design the communication system (Communication module)

3. Develop a board with payloads in the form of sensors (magnetometer, gyroscope-accelerometer, light sensor, temperature, pressure, GPS navigation)

4. Calculate the current consumption of all components and design the power system.

Based on the main requirement, the following components were selected (see Table 1).

Component name	Characteristics
Arduino Nano	Microcontroller: ATmega328 Operating voltage (logic level): 5 B Input voltage (recommended): 7-12 B Input voltage (limit voltage): 6-20 B Digital I/O: 14 (6 of which can be used as PWM outputs) Analog Inputs: 8 DC current through I/O: 40 mA Flash memory: 32KB with 2KB used for bootloader RAM: 2 Kb EEPROM: 1 Kb Clock frequency: 16 MHz Dimensions: 1.85 cm x 4.2 cm
GPS module NEO-6M	GPS module: U-Blox NEO-6M-0-001;

	Built-in battery for fast, cold start; Built-in EEPROM; Sensitivity: -161 dBm; Refresh rate: 5 Hz; Interfaces: UART (output), SPI, DDC, IIC; Transmits coordinates in NMEA format; Default connection speed by UART: 9600 baud; Included active antenna; Power supply voltage: 3 - 5 V; Ability to work with programs: U-Center, etc.; Board Dimensions: 57 x 25 x 15 mm; Kit weight: 18 g.
GY-521 (MPU6050)	Power supply: 3.5 - 6 V; Current consumption: 500µA; Accelerometer measuring range: ± 2 ± 4 ± 8 ± 8 ± 16g, Gyroscope measuring range: ± 250 500 500 1000 1000 2000 ° / s,
HMC5883 magnetometer board	Power supply voltage: 3-5V Rated current consumption (in measurement mode): 2.5 mA Nominal current consumption (in sleep mode): 0.1 mA Interface: I2C (IIC) protocol

	Measuring range: ± 1.3- 8 Gauss Operating temperature range: 0°C - +55°C Relative humidity: 5% - 95%
Light Intensity Module GY-302 BH1750	Power supply voltage - 5 V; Interface: I2C; Chip: BH1750FVI; ADC: 16 bit; Accuracy: 1 lux; Sensitivity: 65536 gradations; Calibration: not required; Dimensions: 19 x 13 x 2 mm; Weight: 5 g.
Micro SD for Arduino	Power supply voltage 3.3V/5V Power dissipation Standard Operating temperature -40-+85
Wireless modules- NRF24L01+PA+LNA	Transmission power more than + 2 0d Bm , 5 0 Ω ,

2.4G, 1100 m	supports 7-channel reception, transmission rate up to 2 Mbps, single packet size: 32 bytes Operating voltage: 2.7 - 3.6 V, but control pins are tolerant to 2.7 - 3.6 V. pins are tolerant to 5 V (i.e. we supply strictly to 3.6 V, but control pins can be connected to Arduino pins without problems). Receiving range: ~1 km Number of channels: 127 Operating frequency: 2.4 GHz Modulation: GMSK Receiver sensitivity-95 dBm Maximum transmit current: 115 mA Maximum current at reception: 45 mA Operating temperature range-45°C...+85°C PA gain: 20 dB LNA gain: 20 dB
ESP32-CAM Wi-Fi module with OV2640 camera	Dual-core 32-bit microprocessor: Xtensa LX6 Wireless modes: WiFi 802.11 b/g/n/e/i/ up to 150 Mbps

	Bluetooth 4.3: BR/EDR/BLE Built-in SRAM memory: 520 KB External Flash memory PSRAM: 2 MB SPI Flash: 32 Mbit OV2640 Camera: 2 MP Camera Resolution: 1600x1200 GPIO I/O Ports: 9 Interfaces: UART/SPI/I2C/PWM/CAN/ADC/DAC/I2S Camera support: OV2640 and OV7670 Built-in flash LED: Yes Support image download to TF card: Yes Built-in 2.4G WIFI antenna: 2dB gain Default serial port speed: 115200 bps Video streaming: Yes Current consumption without flash: 180mA Current consumption in deep sleep: 6mA Pin pitch: 2.54mm Operating temperature range: -20℃ ... +85℃
Solar panel	Solar panel: 0.15W Material: polysilicon epoxy board Operating current: 0-50mA Voltage: 3V Quantity: 1pcs/2pcs/4pcs Product size: 40 x 30mm

Table 1. Selected components

2. Developing a motherboard for the components

Connecting the various components to the Arduino via cables poses some challenges. This is due to the fact that the mass of our satellite should not exceed 1.33 kg. Therefore, it was decided to develop a motherboard for the on-board computer, which will optimize the integration and simplify the connection process. The motherboard for the on-board computer and sensor connection was developed in the EASYEDA development environment. The board is made in a two-layer design and has dimensions of 10x10 cm, which allows to place all the listed components (see Fig.1-2).

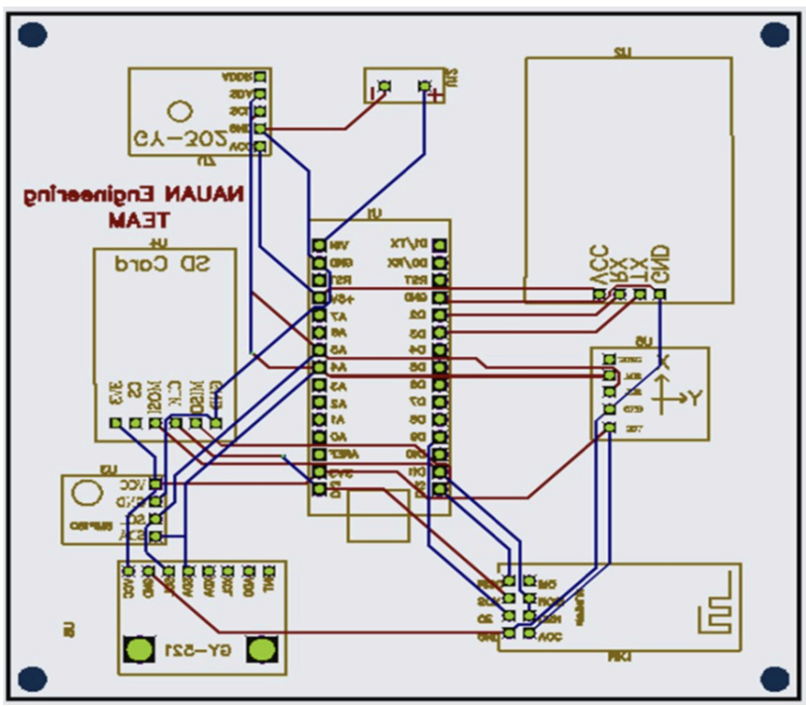


Figure 1. Schematic diagram of the motherboard

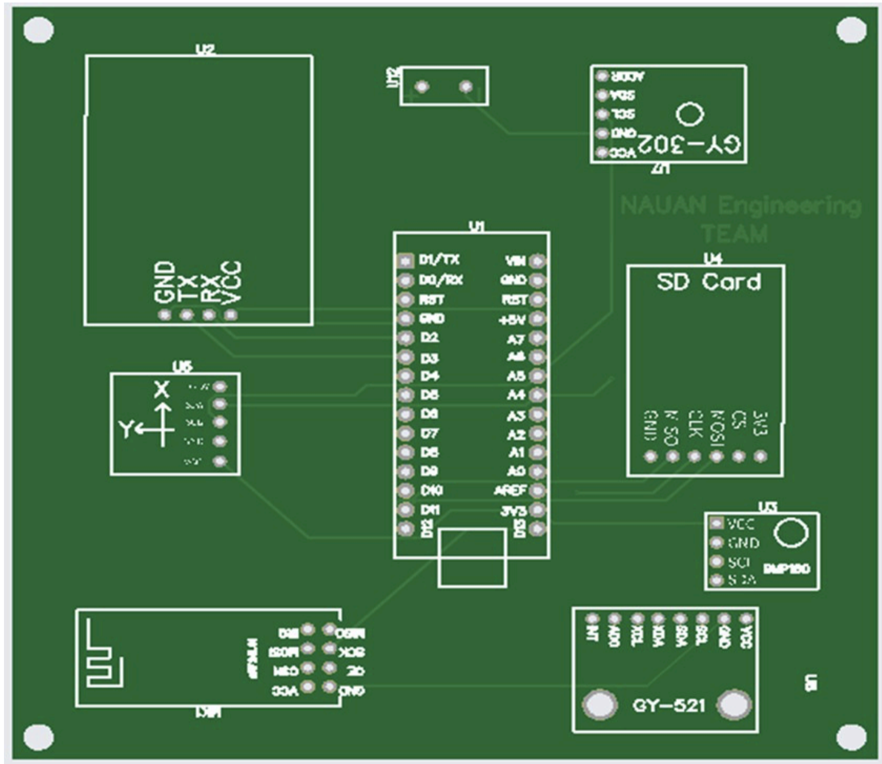
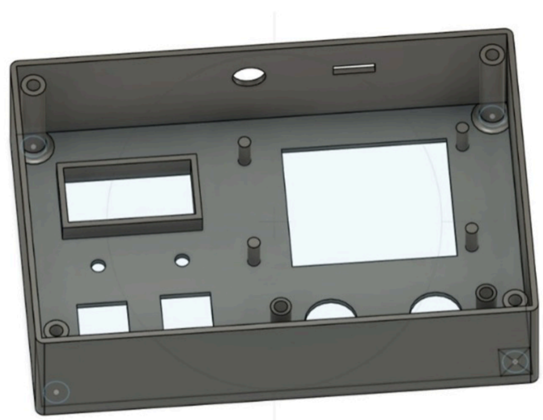


Figure 2. Finished board

3. Assembly of the ground station satellite

The RauSAT 2.0 enclosure was designed in the Fusion 360 environment. In real satellites, the body is made of aluminum alloys. However, for the school prototype it was decided to use 3D printing. PLA plastic was chosen as the printing material (see Figure 3-6).



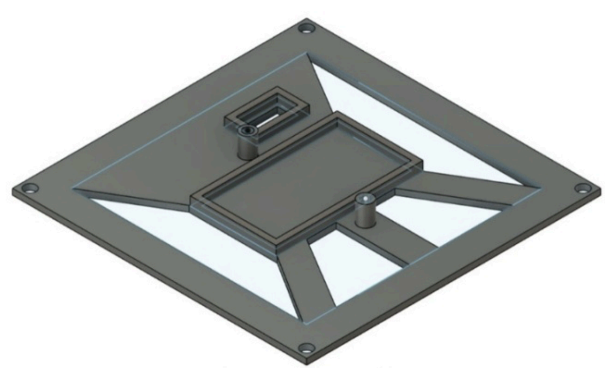


Figure 3. The process of developing the 3-D model of the satellite and transponder

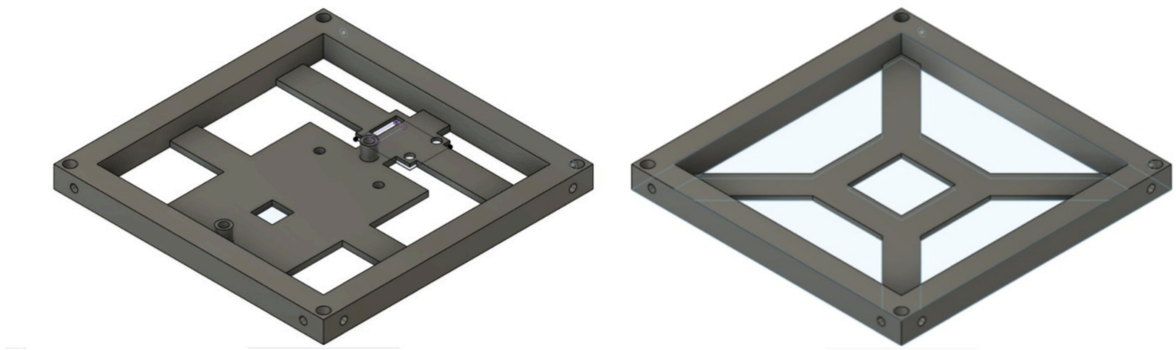


Figure 4. The process of developing the 3-D model of the satellite and transponder

The assembly of RauSAT 2.0 was performed at NIS Engineering. The assembly process resulted in a three-layer satellite:

1. Camera and luxmeter;

2. On-board computer with sensors;

3. Power cells (battery compartment);

4. Principle of operation.

The ground station is represented as a radio receiver, which retransmits the received data from the satellite to the telegram bot. The ground station receives radio signals. The data array created by the satellite is transmitted by radio waves to the ground station. At the same time, the ground station receives this data array and transmits it via Internet communication to the telegram bot (see Figure 7-8).

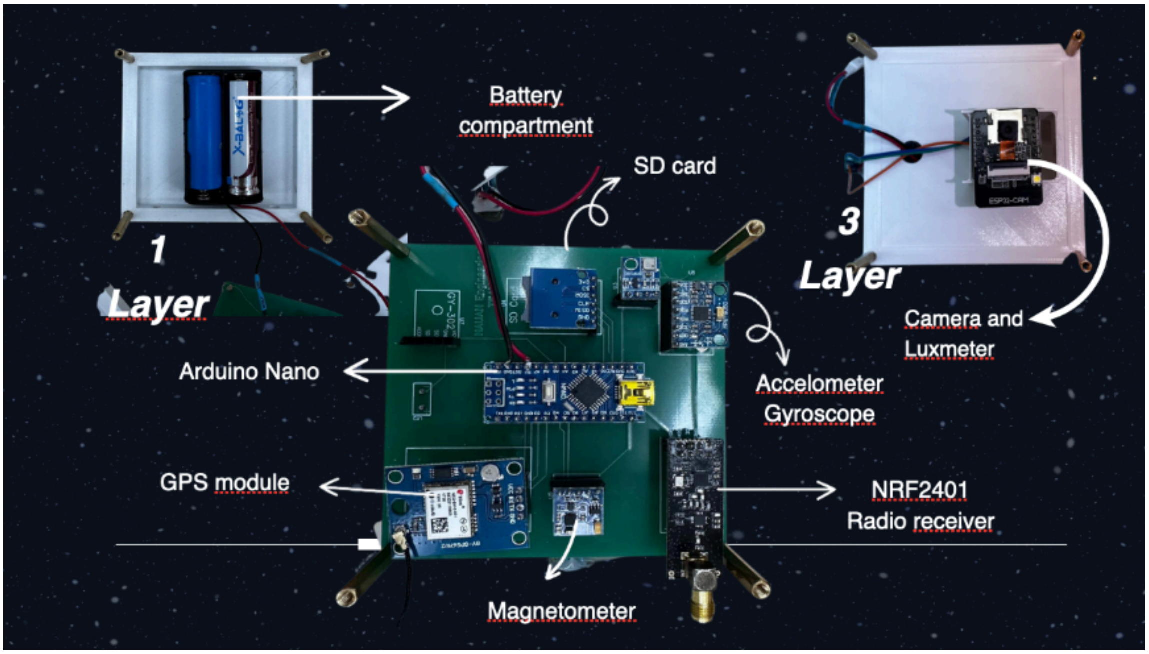


Figure 7. RauSAT 2.0 layers

Figure 8. Principle of operation of RauSAT 2.0

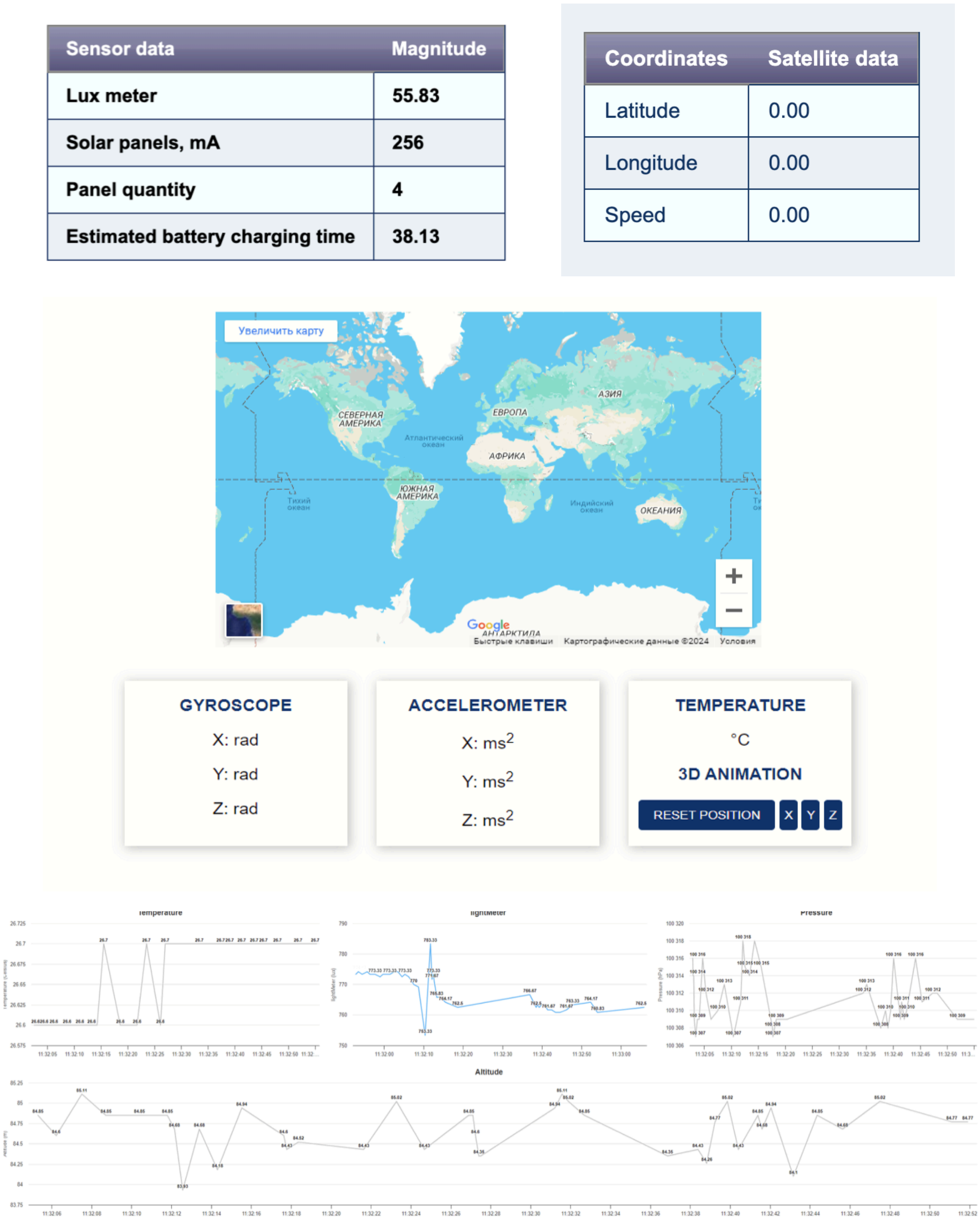


Figure 9. Platform on which the acquired data is displayed

At the moment our satellite successfully transmits data on its location in space, temperature, pressure, altitude, current on solar panels, number of panels and estimated battery charging time, which depends on the current illumination to the server (Fig.9), as well as photos to the Telegram bot. Considering the results achieved, it is safe to say that all the objectives have been met and the RauSAT 2.0 satellite fully meets the requirements of the prototype.

VIII. Conclusion

This project successfully developed the RauSAT 2.0 satellite, designed for environmental monitoring and data transmission to devices via repeaters. It represents an important step forward in the development of space technologies and scientific research, with a wide range of practical applications.

The RauSAT 2.0 mission will collect valuable real-time information about the environment and conditions. This data can be used for climate monitoring, weather forecasting, environmental control, and other tasks. Additionally, RauSAT 2.0 has made a significant contribution to the advancement of communication and data transmission technologies, providing access to information in remote and hard-to-reach regions.

In summing up the project, we emphasize the importance of collective efforts and innovative solutions in the field of space technologies. We are confident that the data obtained during the RauSAT 2.0 mission will be actively utilized and will form the basis for future projects aimed at expanding our knowledge and advancing space technologies.

IX. References

1. List of used literature

1. Statista. (n.d.). *Countries with the most satellites in space*. Retrieved from https://www.statista.com/chart/17107/countries-with-the-most-satellites-in-space/
2. SPUTNIX. (n.d.). *Запущенные миссии* [Launched missions]. Retrieved from https://sputnix.ru/ru/sputniki/na-orbite/

3. RBC Trends. (2022, October 26). *Космос, доступный студентам: что могут спутники Cubesat и сколько стоят* [Space accessible to students: What Cubesat satellites can do and how much they cost]. Retrieved from https://trends.rbc.ru/trends/futurology/635954ba9a7947466adc13b7
4. Heidt H., et al. CubeSat: A new Generation of Pico satellite for Education and Industry Low Cost Space Experimentation // 14 Annual/USU Conference on Small Satellites, Logan, Utah, 14, August 21–24, 2000. Logan, 2000. — SSC00-V-5. — 19 p.
5. Hoyt R., et al. The RETRIEVE microsatellite tether deor bit experiment // AIAA Paper. — 2002. — N 3893.
6. Oehrig J. H., et al. TU Sat 1 — An Innovative Low Cost Communications Satellite // 15th Annual AIAA/USU Conference on Small Satellites Logan, Utah, August 13–16, 2001. — Logan, 2001. — SSC01VIIIb4. — 15 p.

2. Application

Listing of the Satellite program:

```
#include <Wire.h>
#include <BH1750.h>
#include <Adafruit_BMP085.h>
#include <TinyGPS++.h>
#include <SoftwareSerial.h>
#include <nRF24L01.h>
#include <RF24.h>
/***********************/
#include <Arduino.h>
#include <Adafruit_MPU6050.h>
#include <Adafruit_Sensor.h>
/*************/
Adafruit_MPU6050 mpu;
```

```
/*************************************************/
RF24 radio(9, 10);
Adafruit_BMP085 bmp;
BH1750 lightMeter;
TinyGPSPlus gps;
#define S_RX    3              // Вывод RX
#define S_TX    2              // Вывод TX
SoftwareSerial SoftSerial(S_RX, S_TX);
float data1[7];
/***************************************************/
/**********************************************/
void setup(){
   Serial.begin(115200);
   SoftSerial.begin(9600);
   Wire.begin();
   lightMeter.begin();
   if (!mpu.begin(0x68)) {
   Serial.println("Failed to find MPU6050 chip");
   while (1) {
     delay(10);
   }}
   bool status = bmp.begin(0x77);
  if (!status) {
   Serial.println("Could not find a valid BMP180 sensor, check wiring!");
   while (1);
  }
  radio.begin();
   radio.setChannel(0);// задаем канал для передачи данных (от 0 до 127)
   radio.setDataRate (RF24_1MBPS);
```

```
    radio.setPALevel (RF24_PA_HIGH);
    radio.openWritingPipe (0x0987654321DD);
 /* mpu.setHighPassFilter(MPU6050_HIGHPASS_0_63_HZ);
  mpu.setMotionDetectionThreshold(1);
  mpu.setMotionDetectionDuration(20);
  mpu.setInterruptPinLatch(true);  // Keep it latched.  Will turn off when reinitialized.
  mpu.setInterruptPinPolarity(true);
  mpu.setMotionInterrupt(true);*/
}
void loop(){
  data1[0] = bmp.readTemperature();
  data1[1] = bmp.readAltitude();
  data1[2] = bmp.readPressure();
  data1[3] = lightMeter.readLightLevel();
  while (SoftSerial.available() > 0) {
    if (gps.encode(SoftSerial.read())) {
      if (gps.location.isValid()) {
        data1[4] =gps.location.lat(), 6;
        data1[5] = gps.location.lng(), 6 ;
        data1[4]=data1[4]*10000;
        data1[5]=data1[5]*10000;
      }
      if (gps.speed.isValid()) {
        data1[6]=gps.speed.kmph();
      }
    radio.write(&data1, sizeof(data1));
    /* Get new sensor events with the readings */
   /* sensors_event_t a, g, temp;
    mpu.getEvent(&a, &g, &temp);
```

```
  data2[0]=a.acceleration.x;
  data2[1]=a.acceleration.y;
  data2[2]=a.acceleration.z;
  data2[3]=g.gyro.x;
  data2[4]=g.gyro.y;
  data2[5]=g.gyro.z;
```

Ground Station Program Listing:

```
#ifdef ESP32
  #include <WiFi.h>
  #include <ESPAsyncWebServer.h>
  #include <SPIFFS.h>
#else
  #include <Arduino.h>
  #include <WiFiClientSecure.h>
  #include <UniversalTelegramBot.h>
  #include <ESP8266WiFi.h>
  #include <ArduinoJson.h>
  #include <Hash.h>
  #include <ESPAsyncTCP.h>
  #include <ESPAsyncWebServer.h>
  #include <FS.h>
  #include <SPI.h>
  #include <nRF24L01.h>
  #include <RF24.h>
  #include <SoftwareSerial.h>
#endif
```

```
#include <Wire.h>
RF24 radio(4, 5);//CE, CSN
float data1[7];
float Temp1=0,Press1=0,Alt1=0,Hum1,lat1=0,lng1=0,speed1=0;
float t;
uint8_t  pipe;
// Replace with your network credentials
const char* ssid = "WiFi_Users";
const char* password = "ILoveNIS";
// Create AsyncWebServer object on port 80
AsyncWebServer server(80);
String Tempr2() {
   return String(Temp1);
}
String Humm2() {
   return String(Hum1);
}
String Presss2() {
   return String(Press1);
}
String alt2() {
   return String(Alt1);
}
String lat12() {
   return String(lat1);
}
String lng12() {
   return String(lng1);
}
```

```
String speed12() {
   return String(speed1);
}
String bat12() {
   return String(t);
}
void setup(){
  pinMode(D5, OUTPUT);
  // Serial port for debugging purposes
  Serial.begin(115200);
  Serial.setTimeout(50);
  bool status;
  // Initialize SPIFFS
  if(!SPIFFS.begin()){
    Serial.println("An Error has occurred while mounting SPIFFS");
    return;
  }
  // Connect to Wi-Fi
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
    Serial.println("Connecting to WiFi..");
  }
  // Print ESP32 Local IP Address
  Serial.println(WiFi.localIP());
  radio.begin();
  radio.setChannel(0);
  radio.setDataRate (RF24_1MBPS);
  radio.setPALevel (RF24_PA_HIGH);
```

```
radio.openReadingPipe(1,0x0987654321DD);
radio.startListening();
// Route for root / web page
  server.on("/", HTTP_GET, [](AsyncWebServerRequest *request){
  request->send(SPIFFS, "/index.html");
});
 server.on("/temperature1", HTTP_GET, [](AsyncWebServerRequest *request){
  request->send_P(200, "text/plain", Tempr2().c_str());
});
server.on("/humidity1", HTTP_GET, [](AsyncWebServerRequest *request){
  request->send_P(200, "text/plain", Humm2().c_str());
});
server.on("/pressure1", HTTP_GET, [](AsyncWebServerRequest *request){
  request->send_P(200, "text/plain", Presss2().c_str());
});
server.on("/alt1", HTTP_GET, [](AsyncWebServerRequest *request){
  request->send_P(200, "text/plain", alt2().c_str());
});
server.on("/lat1", HTTP_GET, [](AsyncWebServerRequest *request){
  request->send_P(200, "text/plain", lat12().c_str());
});
server.on("/lng1", HTTP_GET, [](AsyncWebServerRequest *request){
  request->send_P(200, "text/plain", lng12().c_str());
});
server.on("/speed1", HTTP_GET, [](AsyncWebServerRequest *request){
  request->send_P(200, "text/plain", speed12().c_str());
});
 server.on("/t", HTTP_GET, [](AsyncWebServerRequest *request){
  request->send_P(200, "text/plain", bat12().c_str());
```

```
});
```

XI. Mentor’s review

The project *“Development of the Research Nano-Satellite RauSAT 2.0,”* presented by student Nadira Atymtay is an excellent example of integrating theoretical knowledge with practical skills within the educational process. Working on the development of a CubeSat enabled the student to fully immerse themselves in the world of space technologies, marking an important milestone in mastering modern engineering disciplines.

This project stands out for its comprehensive approach to solving the set objectives. The student demonstrated a high level of responsibility and determination at every stage of the project—from selecting components to designing the motherboard and assembling the satellite's structure. During the work, the student not only learned how to operate microcontrollers, sensors, and radio modules but also successfully implemented a data transmission system through a server and a Telegram bot, a modern and efficient solution.

An essential aspect of the project is its educational value. While working on this project, the students showcased their ability to apply theoretical knowledge in practice, solve engineering problems, and analyze complex technical processes. Their initiative and independence, which played a key role in the project’s success, are particularly noteworthy.

The project holds significant scientific and technical value and demonstrates impressive educational outcomes by fostering the skills necessary for future careers in the aerospace industry. The RauSAT 2.0 satellite successfully transmits data on its position, temperature, pressure, altitude, solar panel current, number of panels, and estimated battery charging time, along with photos to a Telegram bot, confirming its practical relevance and operational readiness.

The project by Nadira Atymtay fully meets the high standards required for scientific research competitions and deserves high recognition. I am confident that her work will be appreciated at both national and international scientific platforms.

Mentor,

Adilzhan Zhumagazyuly Nauan

Computer Science Teacher, NIS Kyzylorda