Article

TinyGS vs. SatNOGS: A Comparative Analysis of Open-Source Satellite Ground Station Networks

João Sá Gomes [1] and Alexandre Ferreira da Silva [1,2,*]

1   Center for MicroElectroMechanical Systems (CMEMS-UMinho), University of Minho, Campus de Azurem, 4804-533 Guimaraes, Portugal
2   LABBELS—Associate Laboratory, 4710-057 Braga, Portugal
*   Correspondence: asilva@dei.uminho.pt

Abstract: In recent years, two of the largest open-source ground station (GS) networks capable of enabling Earth–satellite communication have emerged: TinyGS and SatNOGS. These open-source projects enable anyone to build their own GS inexpensively and easily, integrate into a GS network, and receive data from satellites listed in the database. Additionally, it enables satellite developers to add satellites to the databases of these projects and take advantage of this GS network to receive data from the satellites. This article introduces the TinyGS and SatNOGS projects and conducts a comparative analysis between them. Generally, the TinyGS project seems to have simpler implementation as well as lower associated costs. In a deeper analysis, it was observed that on the 29 July 2023, the TinyGS project had a higher number of online GSs and a more favorable geographic distribution. On the other hand, the SatNOGS project managed to communicate and decode a larger number of satellites up to 29 July 2023. Additionally, in both projects, it was noted that frequencies between 436 and 437 had the highest number of satellites with decoded data. Ultimately, the choice between these projects depends on critical parameters defined by the reader.

Keywords: SatNOGS; TinyGS; ground station network; LEO satellites

1. Introduction

In the current era, marked by technological advances and innovations in the space sector, we have seen a notable increase in the presence of satellites in orbit [1]. This growth is driven, in part, by the ease and reduced cost of the development and launch of satellites, enabling a variety of entities, such as universities [2–4] and small companies, to actively participate in this ecosystem.

However, beyond construction and launch, there is also a crucial challenge: satellite monitoring. To address this, ground stations (GSs) are needed, as they play an important role in communicating with satellites, serving as the link between the "ground" and these devices orbiting Earth several kilometers above us. This communication is vital to receiving/sending data from/to the satellite, providing access to scientific data, weather data, satellite telemetry data, and other valuable information [5]. However, if the satellite is not in a geostationary orbit (GEO), that is, not over the same location, there are times when satellite–GS communication becomes impossible, particularly for low-Earth orbit (LEO) satellites (which are the main satellites with which both SatNOGS and TinyGS GS communicate), as they are in low orbits with high speeds, as the GS has only a few minutes to communicate with the satellite [6].

To extend this communication time and make it as extensive as possible, a set of GSs must be built in strategic positions. However, studying the number of required GSs and their positions is not an easy task. This is because there are various factors affecting communication between satellites and GSs, such as the satellite's elevation angle and rain attenuation. Nevertheless, there are methodologies that can be employed to study this number and these positions [6,7]. Additionally, to facilitate multiple ground station

operations remotely, the creation of a global management interface, a network [5], is often required. Consequently, the creation of such infrastructure can be a challenging, costly, uninteresting, and time-consuming task. In many cases, such as in universities and small businesses, the primary interest and focus lie solely in the development of satellites, not in the development and study of a GS network. Therefore, the existence of an already built and financially accessible infrastructure would be ideal. Examples of open-source projects with such infrastructure include SatNOGS [8] and TinyGS [9] and are characterized by being non-profit projects, easily accessible, and suited for educational, research, non-profit experimental, and amateur work. Both projects are among the largest open-source ground station networks, aiming to provide an easy and low-cost alternative for communication between Earth and satellites, particularly LEO satellites. Designed for tracking, receiving, and decoding satellite signals, they seek to democratize access to space by providing a decentralized infrastructure. Besides their accessibility and simplicity, they boast an active and collaborative community capable of providing support and promoting participation from anyone.

Consequently, for satellite developers whose sole objective is satellite development or even for space and communication enthusiasts, these two projects emerge as effective and accessible solutions. Thus, if a satellite developer integrates into one of these networks, they can focus their efforts on building and launching satellites without the need to create their own network infrastructure. They can also leverage existing GSs and design the satellite in accordance with the bands and frequencies most commonly used by the project's GSs or those better distributed geographically, in accordance with the radio modulations (e.g., FSK and GFSK) supported by each project [8,9], etc. Additionally, if they join and build their GS, while it is not receiving data from the intended satellite, it can communicate with other satellites and thereby contribute to the community. Regarding the aspect of adding the satellite to one of these projects, since it is also of interest, it is necessary first to provide all the satellite details to the project. These details include the downlink frequency, upload frequency, and, if not confidential, the decoder. To provide this relevant information, Telegram is used for TinyGS and GitLab for SatNOGS (All steps can be found in the Satellite Operator Guide [8]). Once the satellite is added to the projects' databases, ground stations can immediately start communicating with and receiving data from it. This data can then be visualized, for example, in the respective web applications.

Moreover, network collaboration is not exclusive to satellite developers. Space and communication enthusiasts also find valuable opportunities to participate in these projects. First, they can communicate with satellites solely through the simple and inexpensive construction and integration of a GS into the network. Second, in addition to contributing to the community, they also have the chance to deepen their knowledge of satellites and communication. Despite previously emphasizing the need to construct ground stations (GSs) for participation in these projects, it is also feasible to do so by leveraging existing GSs. This approach offers advantages such as cost reduction and rapid integration. However, it is crucial to initially verify whether the components of the GS are compatible with the software used in the projects. For instance, if one intends to utilize a non-fixed GS (equipped with rotators that can change the antenna's direction) for integration into SatNOGS, it is imperative to ascertain whether the computer can run the software and whether the antenna rotator is compatible with the specified software.

Since TinyGS and SatNOGS are two of the largest open-source ground station networks in recent years, in order to assist in a potential decision regarding which project to join, in this article, these two GS networks, SatNOGS and TinyGS, will be introduced and compared. The structure of this article is as follows. In Section 2, the methodology employed in this study is comprehensively addressed, outlining the systematic approach utilized. In Section 3, the constituent parts of SatNOGS, its operation, and the block diagram of SatNOGS's architecture and GS are presented. Section 4 addresses the aforementioned topics, but in the context of TinyGS. Section 5 delves into a detailed discussion of the comparisons mentioned in Section 2. The comparisons are between the block diagrams

of both projects, the difficulties in constructing and configuring the ground stations (GS), software uploading and updating, costs, support, membership count, the number and distribution of online GSs as of 29 July 2023, the number and frequencies of satellites they (TinyGS and SatNOGS) managed to communicate with up until 29 July 2023, the percentage of satellites (among the operational and supported satellites that TinyGS successfully communicated with) that SatNOGS was able to communicate with, the timeline of GS creation, and limitations and implications. Finally, Section 5 concludes the article.

2. Methodology

During this article, the SatNOGS and TinyGS projects were studied, analyzed, and compared. In the sections "SatNOGS" (Section 3) and "TinyGS" (Section 4), a presentation of the corresponding projects is made both in terms of general architecture and operation, as well as in the block diagram of the projects' GSs. In the block diagram of the GSs, the minimum components necessary for the system's operation are addressed as well as additional components (e.g., LNA) used for better signal reception. Additionally, the usefulness of each of these components is discussed in order to better understand their function.

Then, in Section 5, a comprehensive comparison is made between SatNOGS and TinyGS on various topics. They can be seen in Figure 1.
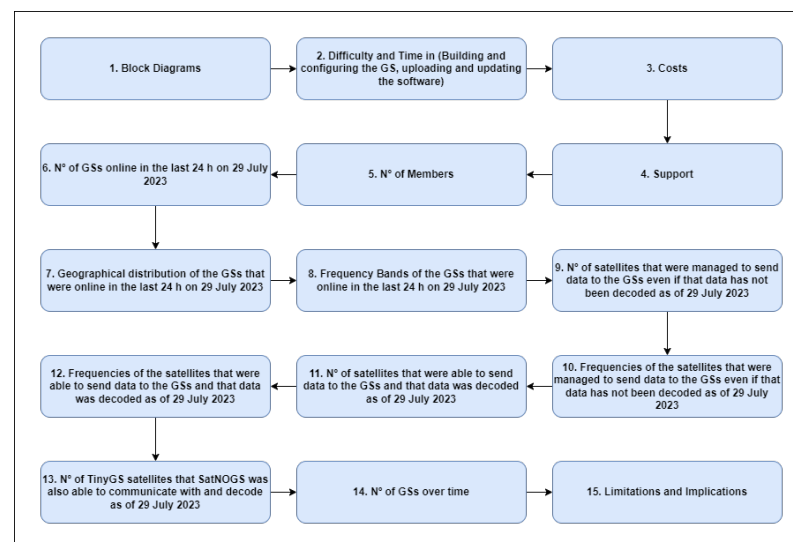
Figure 1. Comparisons to be conducted and discussed in Section 5 between TinyGS and SatNOGS.

First, in Topic 1, the block diagrams of both projects are analyzed and compared. This comparison aims to identify additional constituent parts in one project relative to the other, since these may not only contribute to increased costs but also potentially elevate the difficulty in constructing the GS.

Transitioning to Topic 2, a comparison is drawn regarding the difficulty and time required in building and configuring the GS and uploading and updating the software in both projects across various scenarios. This includes considerations such as with and without antenna construction in both projects, as well as with and without the construction of the rotator and controller components (solely for SatNOGS).

In the subsequent cost comparison (Topic 3), costs for constructing TinyGS and SatNOGS (both fixed and rotator) GSs were compared. This comparison took into account the recommended components (e.g., LNA, filter) specified by each project, aiming to showcase the varying levels of investment required, since this plays a vital role in an enthusiast's decision. The costs presented in the following section were based on August of 2023 for Portugal.

Moving to the comparison of support and membership (Topics 4 and 5), the support used by both projects was compared at the organizational and practical levels. This consider-

ation was based on the author's experience, which may differ from the reader's perspective. Additionally, the number of members in each project was observed and compared. This involved examining the number of sign-ups on the Libre Space Community for SatNOGS and the number of members provided by TinyGS in its web application. The rationale for this comparison is rooted in the belief that projects with a larger community may offer more resources for addressing user queries.

For Topics 6 to 14, all TinyGS data, including GSs details and satellite information, were collected from TinyGS's web application. However, for SatNOGS, the GSs details were obtained from the SatNOGS network API [10] and the satellite details were retrieved from the SatNOGS DB API [11]. GSs' data encompassed their coordinates (latitude and longitude) and frequency ranges, while satellite data covered operational status, downlink frequency, and the number of packets received by GSs from that specific satellite.

In Topic 6, the number of GSs online in the last 24 h on July 29th was compared between both projects. At the time of writing, the TinyGS web application only displays the GSs that were online in the last 24 h, so all GSs in that web application were selected. In SatNOGS, the "status" parameter of various GSs was observed as well as the last time online for GSs that were offline, and only online GSs and those offline in the last 24 h were selected. Topic 7 compared the geographical distribution of the GSs from the previous section based on their coordinates (latitude and longitude). Topic 8 originally intended to compare the frequency ranges of different GSs mentioned in the last two sections of each project. However, due to the multitude of varied frequency ranges, representing them in a histogram became impractical. Instead of focusing on frequency ranges, the analysis shifted to comparing the bands used by these GSs in both projects, such as UHF. This approach enables satellite developers interested in utilizing these GS networks to identify the bands with the highest number of GSs. Consequently, they can make informed decisions about which bands to utilize for their specific needs.

In Topic 9, the number of operational satellites in each project's network which had at least one data reception by GSs as of 29 July 2023 was compared, even if the data were not decoded. It is important to highlight that an operational satellite for the network is a satellite listed as capable of communication, not necessarily a satellite that is literally operational and in orbit. In other words, there could be satellites that are literally operational and in orbit, but if they are not "operational" for the network, it is as if that satellite does not exist, and the GSs will never be configured by the network to receive data from that satellite. In Section 10, the downlink frequencies, also simply referred to as frequencies, of the satellites mentioned in the previous section are observed for each project. Unlike TinyGS, SatNOGS created a histogram with 50 MHz frequency ranges for better visualization.

In Topics 11 and 12, the analysis is very similar to Topics 9 and 10, respectively. However, now the focus is on operational satellites with decoded data for the same date, 29 July 2023. The objective of this analysis is to identify the project with the highest number of satellites having their data received by GSs, as well as the frequencies most commonly used by the satellites. This information helps enthusiasts decide which frequency or frequencies to utilize for their GS when considering joining one of the projects.

In Topics 13, it was investigated whether SatNOGS could receive and decode data from the same satellites as TinyGS. This was aimed at determining whether the project that received and decoded data from more satellites could do the same for satellites from the other project. Therefore, for example, if SatNOGS is capable of receiving and decoding data from most or all of the satellites that TinyGS managed, it would allow an enthusiast whose goal is to receive data from various satellites to decide whether it is worth joining the latter.

Finally, in Topic 14, only the growth of the number of ground stations (GS) in SatNOGS from 2015 to 2023 was observed, as in TinyGS, during the time of writing, it is not possible to access the creation date of GSs that have been offline for more than 24 h. In Topic 15, the limitations and security implications of both projects were mentioned.

This methodology ensured a comprehensive and systematic comparison of various aspects between SatNOGS and TinyGS, offering readers a thorough understanding of

the projects' differences, strengths, and limitations. The next section will provide a more detailed description of each of the aforementioned topics.

3. SatNOGS

3.1. SatNOGS Introduction

SatNOGS is an open-source (hardware and software) and a global network of satellite ground stations used to communicate and receive data from LEO satellites. It is divided into four parts [12]:

- SatNOGS ground station;
- SatNOGS client;
- SatNOGS network;
- SatNOGS DB.

The SatNOGS ground station is relative to all hardware used in the communication from antennas, software-defined radio (SDR), low noise amplifier (LNA), filter, rotator, and rotator controller to the Linux computer (e.g., Raspberry Pi or a Linux desktop) [12]. The rotator serves to vary the azimuth and elevation angles so that the antenna is pointed in the direction of the satellites. It is important to note that here we are considering a rotating ground station; otherwise, we would be talking about fixed ground stations, and in this case, there is no rotator or rotator controller.

The SatNOGS client is a software that runs on a Linux computer and is responsible for periodically checking the SatNOGS network to identify the observations it has to make in order to execute them later. When an observation job is received, it is responsible for controlling the ground station hardware according to the observation. When the observation ends, it sends all of the information (waterfall, audio, etc.) back to the network [12,13].

The SatNOGS network is responsible for allowing the scheduling of observations, storing them so that the GSs know if they have observations, and, if so, executing them at the appropriate time. Furthermore, it is responsible for receiving the observation data from the GS and making it available to everyone on the web [12–14].

The SatNOGS DB or SatNOGS database [15], as the name suggests, is an open database with satellite information like NORAD, mode of modulation, downlink frequency (also mentioned in this article simply as "frequency"), etc. This is useful because, when the user is scheduling an observation and selecting a specific satellite, the SatNOGS network goes to the database to identify which ground stations are able to communicate with that satellite [12,13].

Figure 2 shows the high-level relationships between users, the SatNOGS network, the SatNOGS database, GSs, the SatNOGS client, and the satellites. Bidirectional arrows in Figure 2 represent bidirectional data flow. For example, between the SatNOGS network and the SatNOGS client, installed in this case on a Raspberry Pi, parameters will be sent from the network to the client so that the GS is configured to receive telemetry data. In the opposite direction, that is, from the client to the network, this data will be uploaded. Compared to the SatNOGS database, in the database located within the network, other information is stored, such as future observations and all GSs and their parameters (e.g., location, ID, and frequency range). If one wants to collect or access any information, one can do so through the SatNOGS network [14], SatNOGS DB [15] SatNOGS DB API [11], or SatNOGS network API [10] depending on the data one wants.
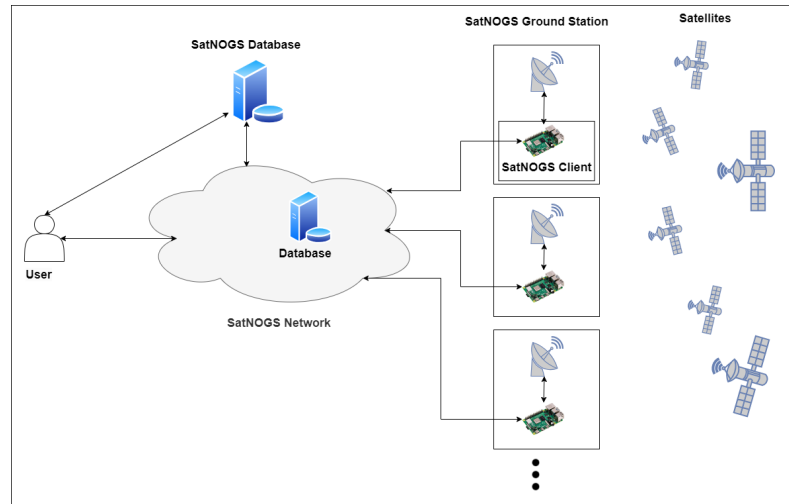
Figure 2. High-level relationship between SatNOGS parts where the bidirectional arrows represent the sending or receiving of data. Satellites represent supported satellites found in the SatNOGS database [15]. The ground station symbol next to the raspberry symbol represents all the hardware, for example, the antenna, LNA, filter, SDR, rotator, and rotator controller.

Regarding the operation of SatNOGS, when a user wants to communicate with a satellite, first he needs to have a GS online and functioning, otherwise he will not be able to schedule observations across the network. Then, he can start scheduling, and he has two options: He can do this by accessing the SatNOGS network through a web interface [14] or using the SatNOGS auto-scheduler [16], which allows one to perform observations automatically. However, this will be determined considering that the observations are carried out on the web since it is easier for people who are not familiar with the command line. The first step is to provide all the necessary details from the satellite one wants to communicate with, the transmitter in case the satellite has more than one (each transmitter has a certain transmission mode, downlink frequency, etc), and the time interval in which one wants to perform the observation. Afterward, the network will be responsible for observing the characteristics of the satellite present in the SatNOGS DB and selecting the most suitable GS to carry out this communication according to its frequency range and availability within the selected time interval. Then, it will present the most suitable GS to the user. From all of the GSs presented, the user selects the desired GS, GSx (the user can select more than one GS), and then a "Job", Jobx, is placed in the queue of GSx jobs according to the order of arrival on the network. Then, the Linux computer periodically asks the network if it has observations scheduled and, if so, it executes them in order. After all of the tasks/jobs that were in front of Jobx are executed, Jobx will start executing, and the GSx will communicate with the satellite selected by the user at the beginning. Subsequently, the GSx will receive the data and send it to the network. In the end, the user who requested the GSx and any other user can observe the data collected from the satellite, for example, via the web network [12,13].
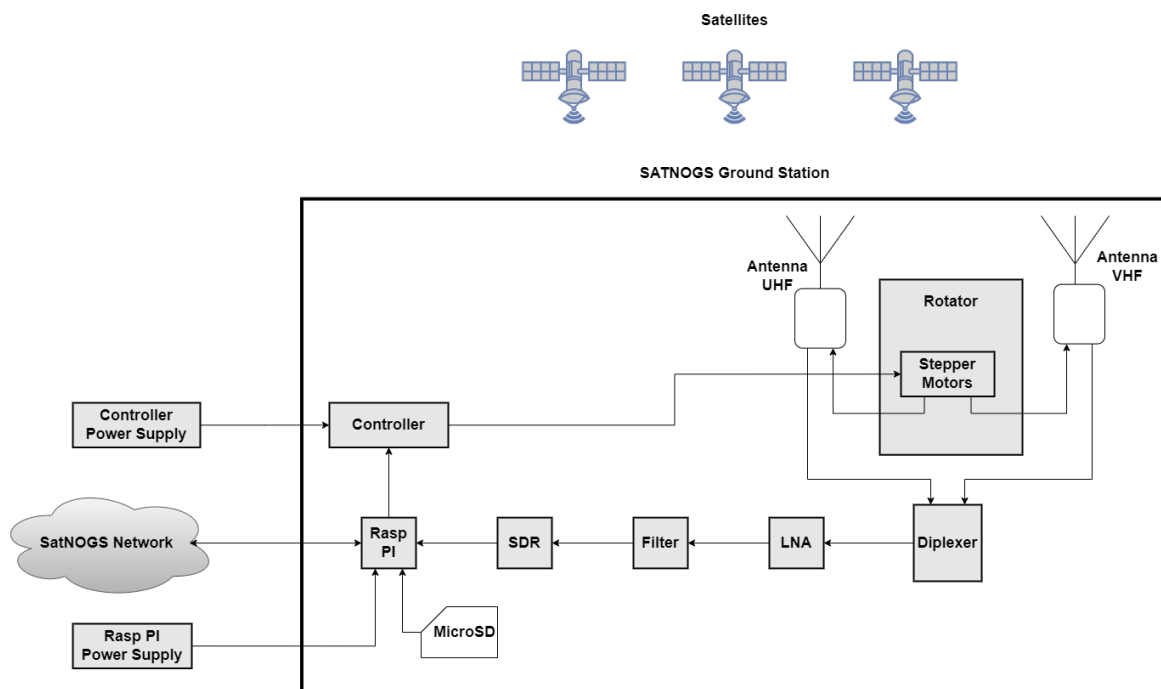
3.2. SatNOGS Ground Station

There are two types of SatNOGS ground stations, the non-rotator ground station (or fixed ground station) and the rotator ground station (or non-fixed ground station). The first is a simpler fixed GS that requires an omnidirectional antenna (e.g., Turnstile, Lindenblad, or QHF). Consequently, it is unable to receive the weakest and most distant signals coming from the most distant satellites. The second is a more complex and expensive GS, mainly due to the presence of a rotator and a rotator controller. It is able to vary elevation and azimuth angles, allowing for the use of directional antennas and always pointing them in the direction of the satellite during its cross between horizons. In this way, it allows one to receive weaker and more distant signals compared to the non-rotator GS.

Starting with the simplest case, if you intend to build the simplest ground station (a fixed ground station) with the minimum number of components, simply construct a GS with:

- Linux computer;(If your Linux computer is a Raspberry Pi, you also will need an SD card)
- Software-defined radio (SDR);
- Antenna.

Despite being the simplest and most cost-effective ground station, it is likely to capture few or no signals. Conversely, if you aim to capture more signals within the fixed ground station, it is advisable to add a low-noise amplifier (LNA) and a filter if you are in a location with a strong presence of FM signals. However, as the rotator ground station with LNA normally allows one to receive a larger number of signals, its block diagram will be focused, and it is presented in Figure 3.

Figure 3. Block diagram of a SatNOGS rotator ground station with two antennas (VHF and UFH antennas).

This block diagram is constituted by a Raspberry Pi (models 3 or 4 are recommended [17]), but it could be any Linux computer (It is noteworthy that the Linux computer in question can range from a Raspberry Pi employing the Raspbian distribution to a desktop Linux system utilizing, for instance, the Debian distribution. Although alternative Linux distributions can be employed, it is recommended to adhere to the recommended ones, as outlined in the SatNOGS client Ansible guide [17] for installing the SatNOGS client. Otherwise, you will need to manually install the SatNOGS station software, a procedure explicitly described as a "tedious and brittle process" and not recommended [17].), where the SatNOGS client will be installed.and is connected to the SatNOGS network, for example, by an Ethernet cable or Wi-Fi. This Raspberry Pi with the SatNOGS client will be responsible for periodically checking the network to see if there are observation jobs to execute. Furthermore, it is connected to the controller, and this is responsible for controlling the motors present in the rotator, in this case, the stepper motors (DC motors are also possible to be implemented), in order to position the antennas in the desired position. If one wants a commercial rotator controller, do not forget to check if it is supported by hamlib-rotctl,

otherwise it will not work, since the SatNOGS client uses hamlib. If one wants a homebuilt rotator controller, one can opt for the SatNOGS rotator controller [18].

The rotator is a mechanism composed of stepper motors (or DC motors), which are connected to the antennas and move them by varying their elevation and azimuth angle, thus allowing the antennas to point in the direction of the satellite. The rotator can be obtained commercially or be built at home. If one wants a commercial rotator, one could consider, for example, Yaesu G-5500 and the Alfa Spid XY [17]. If one wants a homebuilt rotator, one could choose the SatNOGS rotator v3, and there is already a list of the required materials, source files for 3D-printed components(e.g., shaft gear, shaft side, and motor mount), as well as a guide that details how to put all of the components together [17,19,20].

Mechanically connected to the rotator are the antennas, which are responsible for receiving the signals from the satellites. As a large number of satellites work in the VHF and UHF bands, a UHF antenna was placed on the block diagram to receive UHF frequencies and a VHF antenna was added for VHF frequencies. There are several types of antennas, and their selection should be based on the signals with which one intends to work. If one is new and does not have much knowledge about antennas, to start with, one can choose one of the commercial antennas present on the SatNOGS wiki [17]. If, despite being new, one wants to build their own antenna, below are two references one might want to look at [20,21].

Electrically connected to the antennas is the diplexer, and it is responsible for allowing the signals from both antennas to share the same communication channel. If there were no diplexer, it would be necessary to have two communication channels, one for the UHF antenna and another for the VHF. Consequently, the number of components (at least LNA, SDR, and coaxial cables) would be doubled. The filter does not necessarily have to be duplicated because, if it is applied only to block the FM frequencies (88–108 MHz), then it could be placed in the VHF antenna communication channel (30–300 MHz). One can buy a commercial diplexer, but if one wishes to build their own, in order to save some money, one may want to look at the SatNOGS diplexer [22].

Then we have the LNA, which allows us to amplify small signals with minimal noise introduction. It can be powered by a bias tee approach and should be positioned as close as possible to the antenna. If it is not powered by a bias tee, one needs to add its power supply to the block diagram. An example of a recommended and widely used LNA in SatNOGS GS is the LNA4ALL [17].

Regarding the filter, it aims to block unwanted frequencies, such as blocking the FM band in places where there is a strong presence of FM signals. It is important to take into consideration that if the filter is between the SDR and the LNA, and the LNA is powered by a bias tee, it is necessary to make sure that the current supplied by the bias tee passes through the filter and arrives at the LNA.

Then, there is the SDR dongle connected to the computer, a radio communication system consisting of radio components (e.g., amplifiers, filters, and ADC) implemented via software. If one intends to buy an SDR that is not on the list presented on the SatNOGS wiki [17], be careful to ensure that it is supported, since SatNOGS uses the gr-soapy module. Furthermore, it is important to take into consideration that if one uses an LNA powered by a bias tee, one must use an SDR with an activatable bias tee circuit. In terms of recommended and widely used SDRs, there is the RTL-SDR [17].

Finally, one may need to use coaxial cables, which are responsible for transporting the signals between the different components. When using coaxial cables, it is necessary to be careful to verify if one needs connectors and, if so, which types.

4. TinyGS

4.1. TinyGS Introduction

TinyGS is a recent project, which started, more precisely, in 2019 under the name ESP32 Fossa GroundStation. This project takes advantage of the existence of a global network of ground stations distributed all over the world and aims to communicate mainly with

LoRa satellites in order to receive data from them [9]. The advantage of using LoRa is that it allows the user to carry out long-range transmissions with low power consumption, which is why it has been increasingly used, even on satellites where energy consumption is decisive.

TinyGS, as with SatNOGS, can be seen as a project divided into a number of parts, specifically the following three parts:
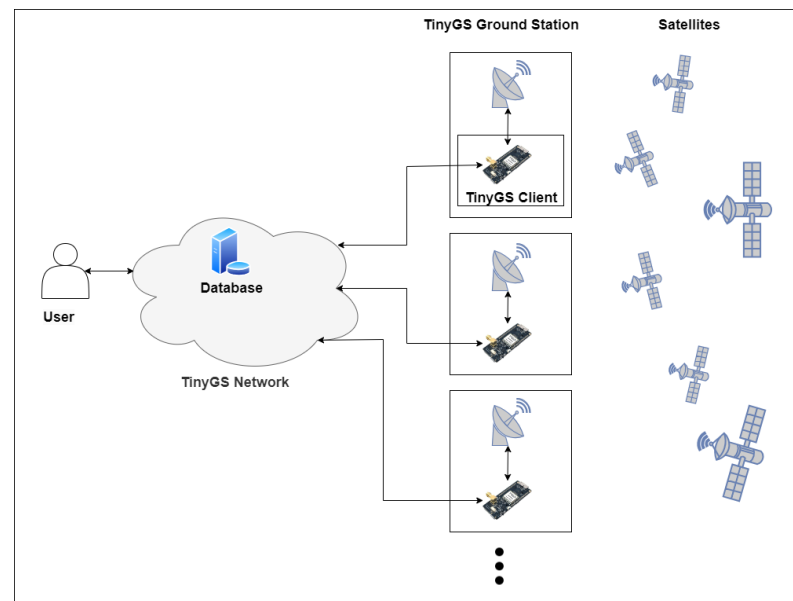
- TinyGS ground station;
- TinyGS network;
- TinyGS "client".

The TinyGS ground station refers to all of the hardware used in the communication from the antenna, LNA, and filter (if needed) to the LoRa module [9].

The TinyGS network is responsible for scheduling observations automatically, sending observations jobs, receiving, saving, and making the data available to everyone [9].

Although the term "TinyGS client" is not used in the TinyGS project, it is used here to create symmetry between TinyGS and SatNOGS and represents the software that runs on the LoRa module. It is responsible for receiving observation jobs and changing the GS configuration according to the satellite with which the network wants to communicate [9].

Figure 4 shows the high-level relationships between users, the TinyGS network, GS, and satellites. Contrary to SatNOGS, in TinyGS, the information about satellites, GS, and observations is contained in the same database. If one wants to collect or access any information, one can do so through the web application [23], Telegram, or the programmatic API (if one owns a GS), depending on the information one wants.

Figure 4. High-level relationships between TinyGS parts. The bidirectional arrows represent the sending or receiving of data. Satellites represent supported satellites found in the TinyGS web application [23]. The ground station symbol next to the LoRa module symbol represents all the hardware, from the antenna and LNA to the filter.

The operation of TinyGS it is slightly different from that of SatNOGS. While in SatNOGS the users are the ones carrying out the observations, in TinyGS, it is the TinyGS network that is responsible for this task. Therefore, it is the TinyGS network itself that decides, by proximity, which satellite (within supported satellites) the ground station will communicate with and also indicates the configuration that the GS must have to be able to communicate with that satellite. However, if one wants to communicate with a particular satellite, it is possible to do so, but only with its GS. To do that, one has to configure the GS according to that satellite, and it is advisable to disable automatic tuning [9]. After the GS

is configured to communicate with a certain satellite, when that satellite passes over the GS, communication takes place, and the LoRa board receives the data. Then, the LoRa board forwards the data to the network where it is stored on a database and available to any user through the web application and Telegram. Furthermore, if the user wants to get the data (or change the GS parameters), it is possible to do so using the programmatic API [9].

4.2. TinyGS Ground Station

The TinyGS ground station is one of the most important parts of the TinyGS project, since it is what allows us to communicate with the satellite. Contrary to SatNOGS, the TinyGS software, at the time of writing, is not made to control rotators for antenna positioning. Consequently, its block diagram does not include a rotator or rotator controller. As an initial step, if one intends to construct a TinyGS ground station with the minimum number of components, the following suffices:

- Lora Module;
- Antenna.

However, in this scenario, and contingent upon the antenna's quality primarily, there may be limited or no signal reception due to the inherently weak signals. Therefore, the addition of a low-noise amplifier (LNA) can prove to be pivotal in enhancing signal reception. An example of a block diagram with it is presented in Figure 5.
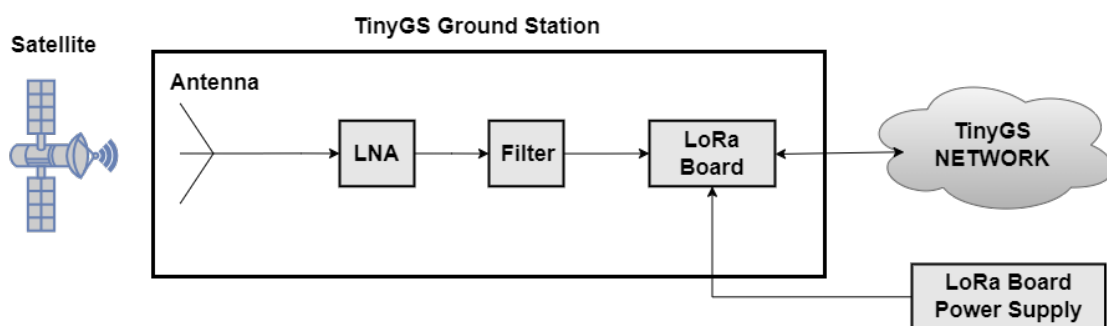
Figure 5. Block diagram of a TinyGS ground station with an LNA.

The TinyGS ground station is made up of an antenna that is responsible for receiving all signals sent by the satellite. As mentioned earlier, since the radio signal received is very weak, it is recommended to use an LNA to amplify the signal and add the minimal amount of noise possible. The LNA can be powered by an external source or a bias tee. Although the bias tee may be a better solution, it can bring certain issues to the LoRa board, as it can route DC back to the LoRa module, which one can read about on the TinyGS github [9]. To address this issue, it is advisable to incorporate a DC block between the LNA and the LoRa board. This serves as a barrier to prevent the current from reaching the LoRa module [9]. Regarding recommended LNAs, examples include the LNA4ALL and the Cubesat Filtered Preamp [9]. After the LNA comes the filter, and as was mentioned earlier, it serves to block unwanted frequencies.

Then there is a LoRa board that has the function of receiving the data sent by the satellite, processing them, and sending them to the network. Conversely, it receives jobs originally coming from the network that indicate the configuration that the GS must have to be able to communicate with the satellite. This LoRa module, in order to send and receive data from the network, will need to be connected to Wi-Fi.

Finally, addressing a common concern for those new to this field, LoRa operates in different frequency bands depending on the region. For example, in Europe, LoRa mainly operates in the 863–870 MHz frequency band, while in the United States, it operates in the 902–928 MHz frequency band [24]. It is important to understand that these frequency restrictions are applicable only if one intends to transmit signals within these regions. Consequently, if someone in Europe is interested in building a 433 MHz ground station to

receive signals, there is no issue. However, it is crucial to emphasize that building a ground station that transmits on 433 MHz requires appropriate licensing. A list of supported TinyGS LoRa boards can be referenced on the TinyGS GitHub page [9]. If, for any reason, one does not have access to any of the LoRa modules listed, there is an alternative option available. It is possible to use any ESP32 board with an sx126x or sx127x module. However, to ensure compatibility with the firmware, the module must be configured appropriately. This configuration process can be facilitated using templates, as detailed in the "Board Templates" section on the project's GitHub page [9].

5. Discussion

In this section, all of the topics mentioned in the previous section will be discussed and presented in greater detail.

5.1. Comparison of the Block Diagrams

In the first instance, consider the block diagram of a fixed SatNOGS GS with only one omnidirectional antenna. Compared to a rotator SatNOGS GS, Figure 3, the fixed SatNOGS GS has no rotator, diplexer, or controller and its power supply. Consequently, we would have the LNA connected directly to the antenna. In this way, the main difference between a fixed SatNOGS GS and a TinyGS GS is the presence of an SDR since the LoRa module "replaces" the Raspberry. In terms of construction, the difficulty remains practically the same, but in terms of costs, it becomes extra. The cost of the SDR will vary depending on which option one chooses, but, for example, if one opts for a commonly used SDR, the RTL-SDR, then the cost is around EUR 30.

Now, comparing the block diagram of a rotator GS, Figure 3, with that of a TinyGS, Figure 5, in addition to the SDR, there is the controller, the rotator, and the diplexer. Unlike the previous case, now both difficulty and cost increase considerably. Concerning the level of difficulty, it varies based on the choice between building a custom rotator or opting for a commercial one. Constructing a custom rotator presents a more complex challenge. For example, assuming that all necessary materials are already available, assembly of a TinyGS ground station can be completed in a single day. On the other hand, building a rotator SatNOGS GS (with SatNOGS rotator v3 and SatNOGS rotator controller v2.4) might take a few days. As for costs, they increase significantly, particularly with the mentioned rotator and controller alone, potentially adding at least EUR 400 to the expenses, depending on the reader's location and component accessibility.

5.2. Comparison of the Difficulty and Time in Building and Configuring the GS and Uploading and Updating the Software

During this comparison, certain considerations will be taken, such as already having the LNA, filter, Raspberry or LoRa module, diplexer, SDR and/or the materials to build the remaining components (antenna, rotator, and controller). Obviously, when talking about the SDR, diplexer, rotator and controller, these only refer to SatNOGS.

Comparing fixed SatNOGS GSs [17] with TinyGS GSs exclusively in terms of GS construction, without antenna construction, both have low difficulty. Constructing a custom antenna produces a slight increase in difficulty, though the level will depend on the specific type of antenna chosen. For instance, building a turnstile is relatively simpler than building a QHF antenna.

However, a rotator SatNOGS GS with a SatNOGS rotator v3 [17,19] is highly difficult since the construction of the rotator and controller system is more challenging, and this will considerably increase the time needed to build it. Again, this difficulty can vary if one intends to build the antennas and which antenna one chooses.

Regarding GS configuration and software upload in SatNOGS, although it is relatively easy compared to TinyGS, it becomes more complex and more susceptible to problems. In the case of SatNOGS, it is necessary to download the latest Raspbian SatNOGS Image (artifacts.zip) from the wiki [17] or GitLab, flash the SatNOGS image (.img file) to a micro

SD card (using, e.g., Balena) and then into the Raspberry Pi. Then it is time to do the SatNOGS Client setup. To do so, one needs to go to the Raspberry's console and run "sudo raspi-config" and then "sudo satnogs-setup". Next, one needs to set up, at least, its basic configuration option by setting its API Token (go to SatNOGS Network -> Dashboard -> API key), its RX device (set rtlsdr if one is using an RTL-SDR), elevation (in meters), and so on. Furthermore, depending on its GS components and the performance of its GS, some parameters may need to be changed or added in the advanced configuration option. Lastly, remember to click "Apply", otherwise the configuration will not be saved. Following this step, the ground station will be ready, and its presence can be observed on the SatNOGS network, as indicated by "Last seen, 0 min ago" [17].

With TinyGS, the process is relatively faster, simpler, and less susceptible to problems. The first step is to download the TinyGS Uploader according to the utilized operating system, connect its board to the computer, and then upload the TinyGS firmware to the LoRa board. After the board boot, one needs to configure its station by first connecting to a Wi-Fi network called "My TinyGS" and second setting some parameters like its latitude, longitude, MQTT credentials, Wi-Fi SSID and password to have access to the network, etc. These MQTT credentials are provided by the TinyGS Personal Bot on Telegram. After that, its GS will be ready and added to the system [9]. One can see if its GS is connected to the network on its station dashboard. To do that, one needs to place the IP address that appears on the LoRa board in the browser and check if the MQTT server status is equal to "CONNECTED".

Regarding software updates, TinyGS has the advantage of automatically updating itself by enabling automatic firmware updates [9]. However, in SatNOGS, updating SatNOGS Client has to be conducted manually by running "sudo satnogs-setup" and selecting "Update" and "Apply".

Finally, considering the significance of the difficulty/time relationship and assuming the inclusion of the previously mentioned tasks (building, configuring, updating, and uploading), it was decided to create Table 1, wherein different scenarios and their corresponding difficulty/time relationships can be observed. Within the scope of various scenarios, it was decided to specify the inclusion or exclusion of the antenna's construction for both TinyGS and SatNOGS and the inclusion or exclusion of the rotator and controller's construction solely for SatNOGS. This decision stems from the fact that these components represent some of the most costly elements, as will be elaborated upon in the cost section. Naturally, readers may also opt to construct their own LNA and filter for cost reduction purposes, albeit at the foreseeable expense of increased difficulty and time investment. However, for this comparison, the construction of these components was not factored in. To make Table 1, it was assumed that the reader had acquired all of the necessary materials. Regarding antenna construction, it is presumed that the omnidirectional antenna used is a double turnstile antenna and the directional antenna is a Yagi antenna. Additionally, for SatNOGS, in instances where the construction of the rotator and controller is mentioned, this pertains to the construction of a SatNOGS rotator v3 and a SatNOGS controller.

Table 1. In this figure, the difficulty/time relationship for various scenarios of both projects is depicted. Difficulty is categorized into five levels in increasing order: Low = 1, Medium-Low = 2, Medium = 3, Medium-High = 4, and High = 5. As for time, there are four levels, also in increasing order: Low ($\leq$1 day) = 1, Medium (>1 day and $\leq$5 days) = 2, High (>5 days and $\leq$10 days) = 3, and Very High (>10 days) = 4.

Scenario	Difficulty	Time
TinyGS GS (Without building the omnidirectional antenna)	1	1
TinyGS GS (Building the omnidirectional antenna)	3	2

Table 1. *Cont.*

Scenario	Difficulty	Time
Fixed SatNOGS GS (Without building the omnidirectional antenna)	2	1
Fixed SatNOGS GS (Building the omnidirectional antenna)	4	2
Rotator SatNOGS GS (Without building the directional antenna, rotator and controller)	3	1
Rotator SatNOGS GS (Building the directional antenna without building the rotator and controller)	4	2
Rotator SatNOGS GS (Building the rotator and controller without building the directional antenna)	5	3
Rotator SatNOGS GS (Building the rotator, controller, and the directional antenna)	5	4

In summary, TinyGS without antenna construction represents the project with the lowest difficulty/time ratio, which can be achieved in a single day with relative ease. Conversely, the most complex and time-consuming project corresponds to SatNOGS, involving the construction of the antenna, rotator, and controller components.

5.3. Comparison of Costs

GS construction costs vary according to the chosen components. Before proceeding to outline the costs for each scenario, it is crucial to emphasize that the list of components presented is based on items recommended by these projects and their corresponding online prices at the time of writing this article, specifically in August 2023 and for Portugal. This implies that the reader has the flexibility to purchase alternative components or build their own components, ultimately reducing the overall expenditure. Additionally, during the cost calculation, all components (LNA, Filter, etc.) will be included. However, the reader is not obligated to purchase all of them. For instance, in cases where there is a minimal presence of FM signals, the acquisition of a filter may not be necessary.

If one opts for a TinyGS, the costs are shown in Table 2 and will cost at least EUR 119. In addition, one may need to buy coaxial cables, SMA adapters, outdoor protective housing, etc. However, these expenses are not accounted for because they depend on the objectives and where one is going to build the GS. Still, with respect to Table 2, an LNA with a built-in Upotronics bandpass filter was chosen. Its price is relatively high, but, within the recommendations on GitHub [9], it was the only one obtainable in Portugal.

If one chooses to build a fixed SatNOGS GS in accordance with the recommended materials, but with the same TinyGS antenna, the costs are around EUR 164, as can be seen in Table 3. This increase is mainly due to the presence of the SDR. The LNA present in Table 3 can be powered by a bias tee; however, it is also possible to power it with an external source according to RTL-SDR.com. If one wants to work with a bias tee, one will need to enable the built-in bias tee in RTL-SDR because, by default, it is turned off. Furthermore, if the LNA is bias tee powered, the filter may prevent the bias tee from getting to the LNA. In that case, one should put the filter between the antenna and the LNA; however, one may

obtain worse results. If one does not have a strong presence of FM signals, one can always remove the filter and see if this produces better results.

Table 2. Costs of the different components in the construction of a reference TinyGS GS.

Component	Type	Approx. Price (EUR)
LoRa module	TTGO LoRa32 V2 (433 MHz)	27
LNA and filter	LNA and SAW Bandpass filter Uputronics (435 MHz)	52
Antenna	HYS Antenna Omni (433 MHz)	40

Table 3. Costs of the different components in the construction of a reference fixed SatNOGS GS.

Component	Type	Approx. Price (EUR)
Raspberry	Raspberry Pi 3B	40
SD card	Sandisk 32GB, C10	12
SDR	RTL-SDR v3	35
Filter	Broadcast FM Block Filter	16
LNA	SPF5189Z LNA (by RTL-SDR.com)	21
Antenna	HYS Antenna Omni (433 MHz)	40

Concerning the construction of a rotator SatNOGS using a SatNOGS rotator v3, the associated costs amount to approximately EUR 884, as detailed in Table 4. In this particular GS, given the inclusion of a rotator, the omnidirectional antenna is replaced with a directional one, specifically the Wimo X-Quad [17]. Although this antenna comes with a higher price tag, it serves as a reference and is recommended if one opts for a commercial alternative. Nevertheless, significant cost savings can be achieved by constructing one's own antenna. Examples include the QFH and the SatNOGS helical antenna, both of which yield satisfactory results [20]. The cost of the rotator and the controller includes expenses for 3D printing and online purchases. Opting for commercial rotators might escalate the overall cost—for instance, choosing the Yaesu G-5500 could increase the cost to over EUR 700. However, the use of the g5500-ardushield (compatible with Arduino Uno) developed by SatNOGS [25] provides a cost-saving option to control the Yaesu G-550, avoiding the purchase of relatively expensive controllers like the GS-232B.

Table 4. Costs of the different components in the construction of a reference rotator SatNOGS GS.

Component	Type	Approx. Price (EUR)
Raspberry	Raspberry Pi 3B	40
SD card	Sandisk 32GB, C10	12
SDR	RTL-SDR	35
Filter	Broadcast FM block filter	16
LNA	SPF5189Z LNA (by RTL-SDR.com)	21
Rotator and Controller	SatNOGS rotator v3 and SatNOGS controller	500
Diplexer	SatNOGS diplexer	25
Antenna	Wimo X-Quad (432 MHz)	235

Finally, the costs associated with constructing a TinyGS ground station and a fixed SatNOGS ground station are quite affordable for the average enthusiast and/or other entities. However, these costs can be further mitigated through methods such as building the antenna oneself and opting for more economical components, among other considerations. Nevertheless, the establishment of a rotator SatNOGS ground station, particularly for the average enthusiast, entails a relatively higher expense.

5.4. Comparison of Support

TinyGS and SatNOGS use two different supports. While SatNOGS uses Libre Space Community [26], TinyGS uses Telegram. This is where one can have questions answered, either by posting questions or by reading the answers to other similar questions.

Regarding Telegram, there are three channels:

- tinyGS Telemetry: Channel where the telemetry is placed, the satellite that sent the telemetry data, the stations that received it, as well as the demodulated and decoded data;
- TinyGS Personal Bot: Channel responsible for providing the MQTT credentials, observing the status of its stations, deleting a GS, and generating a link to access the web;
- tinyGS Community: This channel is divided into eleven topics. For example, there is "General" where one can type a general concern, another called "Where to Buy" where one can find links to components used or suggested by other people in the community, and "Technical Problems" which addresses technical problems.

In the Libre Space Community, if one wants to clear up any doubts related to the SatNOGS project, select the "SatNOGS" category, and within it there are two subcategories: "Hardware" and "Software". The choice will depend on the type of issue one has.

When comparing the two support options, on a personal level, SatNOGS has a more organized system, as each question is a topic. This facilitates the search and does not force one to read other questions that do not interest them. Regarding the speed of response, in both projects, the responses are given relatively quickly (most on the same day) by the organizers or by other members of the community. Finally, and regarding which support is the most practical one, the author would say TinyGS, since Telegram is a more practical tool.

5.5. Comparison of the Number of Members

The number of SatNOGS members on 29 July 2023 was based on the number of sign-ups on the Libre Space Community, since SatNOGS is a project of the Libre Space Foundation. Initially, determination of the number of SatNOGS members was intended to be undertaken through the number of accounts created on the SatNOGS network, but taking into account that this information is not publicly available, it was decided to use the previous reasoning. As a result, the number of SatNOGS members will be approximate. In turn, the number of TinyGS members is based on the number of members available on the web application [23].

The number of sign-ups in the Libre Space Community is approximately 2500, so the number of members in SatNOGS is less than or equal to this. Taking into account that the number of SatNOGS created up to 29 July 2023 was 2841 (considering the number of GSs created so far from the SatNOGS Network API), the number of SatNOGS members should be close to the number of sign-ups.

As for TinyGS, the number of members is approximately 4400.

5.6. Comparison of the Number of GSs that Were Online in the Last 24 h on 29 July 2023

In the TinyGS web application, at the time of writing of this article, only the GSs that have been online in the last 24 h are available. Some may be offline, but that means they have only went offline within the last 24 h. In this way, in order to determine the number of TinyGS that had been online in the last 24 h, the TinyGS web application was consulted, and the result was obtained [23].

In SatNOGS, we utilized the SatNOGS Network API [10] to access information about all SatNOGS GSs. Only the GSs that were online or went offline within the last 24 h were selected. To make this selection, we considered only the "status" parameter for online GSs and both the "status" and "last_seen" parameters for offline ones. The "last_seen" parameter allowed us to determine the date and time when a GS was last seen by the network, indicating when it had gone offline. It is important to highlight that within the offline GSs observed by the network in the last 24 h, this category includes GSs marked with a "Testing" status. GSs designated as "Testing" are not available for other observers to schedule observations; only the GS owner has this privilege. This distinction introduces a level of inaccuracy in determining the number of GSs online in the last 24 h, as some may have a "last_seen" timestamp within this timeframe but with a status of "Testing" rather than "Online".

On TinyGS, we found 1307 GSs, while on SatNOGS, we found 255 GSs. This means that TinyGS had approximately five times more antennas online.

5.7. Comparison of the Geographical Distribution of GSs that Were Online in the Last 24 h on 29 July 2023

In Figure 6, the distribution of the TinyGS GSs that were online in the last 24 h on 29 July 2023 are shown.

Figure 6. TinyGS GS distribution on the 29 July 2023.

In Figure 7, one can find the distribution of the SatNOGS GSs that were online in the last 24 h on 29 July 2023.

The first observation is that, in both projects, most of the GSs are found in Europe and North America. Then, in both projects, a small number of GSs can be observed in South America and on the African and Asian continents. As a result, if the reader intends to build a GS and has the possibility of building it in areas with lower numbers of GSs, this construction would be very helpful to these networks. Comparing both projects, one can observe and conclude that TinyGS had a better distribution than SatNOGS on 29 July 2023.

Figure 7. SatNOGS GS distribution on the 29 July 2023.

5.8. Comparison of the Frequency Bands Used by GSs that Were Online in the Last 24 h on 29 July 2023 and the Number of GSs that Use Them

In Figure 8, one can find the bands used by the TinyGS GSs. The "None" column represents the GSs that had no information in the "Band" field. This field can be found in all of the stations of the web application [23]. The columns that have the bands in square brackets represent GSs that work in multiple bands. For example, a GS that works in a frequency range between 90 and 2500 MHz is in the column [VHF, S] because it works in the VHF bands up to S, that is, VHF, UHF, L, and S.
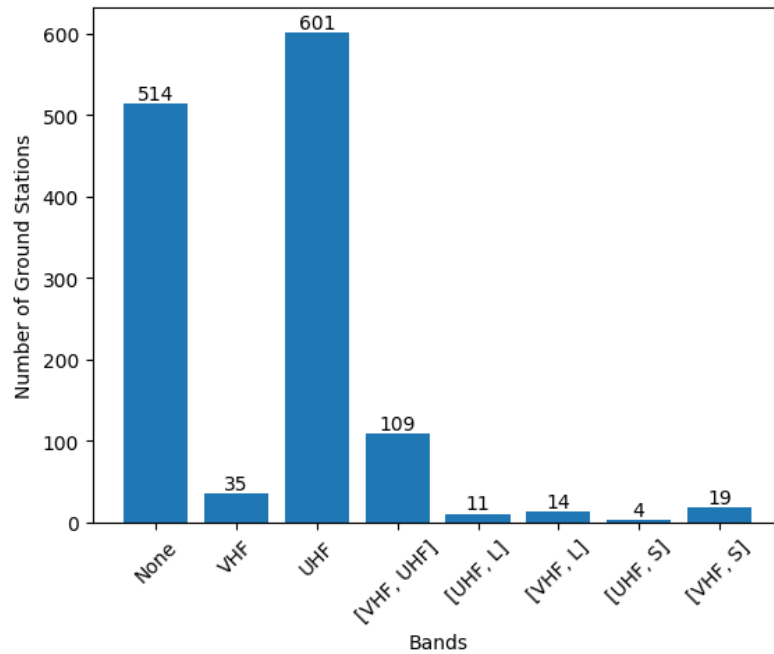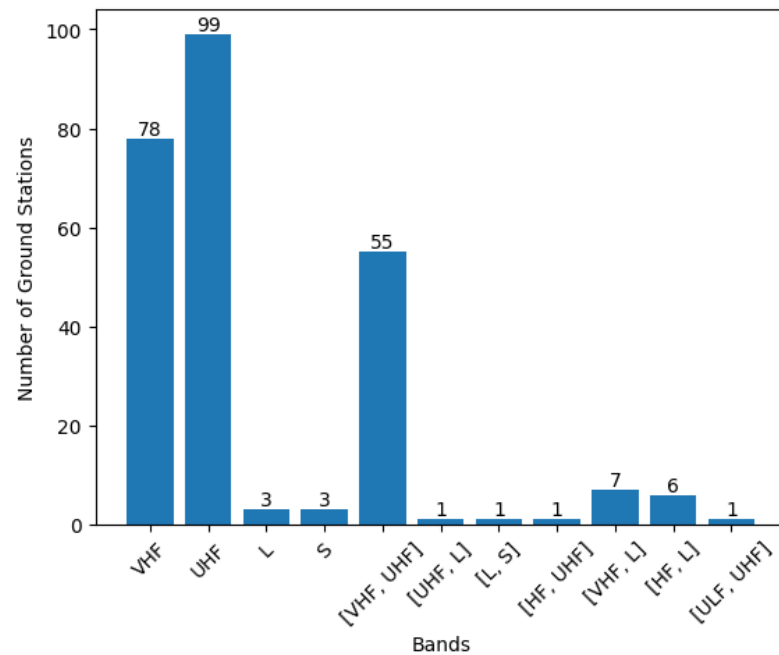

Figure 8. Bands used by TinyGS GSs in the last 24 h of 29 July 2023.

Ignoring the GSs classified as "None", it can be concluded that about 76% of the GSs work only in the UHF band. The reason for this is due to the fact that most of the satellites supported by TinyGS work in that band.

A SatNOGS GSs histogram was created and is shown in Figure 9. For SatNOGS, the process was relatively easier because it was enough to observe the "band" parameter of each GS [10]. For GSs with more than one band, one unit was added to the respective

bands. For example, for a GS that works in the UHF and VHF bands, one unit was added in the UHF column and another in the VHF. However, if a GS has a band like UHF, UHF, VHF, VHF, L, L, it was considered equal to VHF, UHF, L or, as in the histogram, [VHF, L].

Figure 9. Bands used by SatNOGS GS in the last 24 h of 29 July 2023.

As for SatNOGS, approximately 40% work only in the UHF band, 31% in the VHF, and 25% in the VHF and UHF bands ([VHF, UHF]).

As can be seen in both figures, for both projects, the highest percentage of GSs work in the VHF, UHF, and [VHF, UHF] bands. The reasons for this are due to the fact that most satellites work in these bands.

5.9. Comparison of the Number of Satellites that Were Operational and Managed to Send Data to the GSs even if that Data Has Not Been Decoded as of 29 July 2023

In this section, and in the following three sections, satellites will be discussed. For SatNOGS, we took the satellites that were operational according to the SatNOGS Database on 29 July 2023, while for TinyGS, we only took the satellites classified as "supported" in the TinyGS web application [23] and that were operational according to the SatNOGS DataBase. Why are the "supported" satellites chosen for TinyGS? As previously mentioned, one of the functions of the TinyGS network is to decide with which satellite the GS will communicate. The decision is made only taking into account the satellites classified as "supported". "Inactive" satellites, as the name implies, are inactive (even if they are still operational and in orbit according to the SatNOGS database), so when the TinyGS network has to decide which satellite the GS will communicate with, it does not count the "inactive" satellites, but only the supported ones. That is why, for TinyGS, we only select the "supported" satellites. In addition, only currently operational satellites were selected because those that re-entered the atmosphere or are inoperative are no longer of interest. To verify that the satellites were operational, the NORAD of satellites and the SatNOGS DB API [11] were used, and it was verified whether the status of each satellite was equal to "alive". Finally, and regarding this topic, it was decided to study the number of satellites that saw their data being received by the GSs, even if those data had not been decoded. Why were these satellites without decoded data chosen? This was because the data were being received correctly, it was just not possible to decode the data because the decoder had not yet been made available or implemented. The numbers are as follows:

- SatNOGS: 611;
- TinyGS: 25.

As we can see, within the operational satellites, the number of satellites with which SatNOGS was able to communicate, even if the data were not decoded, was almost 24 times greater. However, it is also necessary to bear in mind that TinyGS only has the main objective of communicating with LoRa satellites, and therefore, a much lower number was expected.

5.10. Observation of the Frequencies of the Satellites that Were Operational and Managed to Send Data to the GSs even if this Data Has Not Been Decoded as of 29 July 2023

In Figure 10, it is possible to see the frequencies of the satellites that were operational (according to SatNOGS DataBase) and classified as supported (according to the TinyGS web application) on 29 July 2023 and that had at least one data packet received by a TinyGS GS. To see if the satellites were operational, the SatNOGS DB API [11] was used. To see if they were supported, their frequencies, and to check if the TinyGS GSs had received packets from that satellite, the TinyGS web application was used [23]. Additionally, if one counts the number of satellites in Figure 10, one will see that it is not equal to the 25 mentioned above. This is fundamental, as there are satellites, such as SATLLA-2B and ThingSat, capable of communicating with more than one frequency. Although, for example, TinyGS supports the 437.25 and 2401 MHz frequencies in SATLLA-2B, most of the packets received, if not all, are received with a frequency of 437.25 MHz. However, the satellite transmits on that frequency (2401 MHz), and since the TinyGS allows it to receive packets with that frequency, it was decided to place it in the histogram. For ThingSat, the reasoning is the same; however, most packets are received with a frequency of 868.1 MHz. According to Figure 10, it is possible to observe that most of the satellites that obey the previous conditions have frequencies around 400, 402, 436 and 437 MHz.

It is also noted that the satellite frequencies in Figure 10 were rounded in order to facilitate the visualization of the histogram.
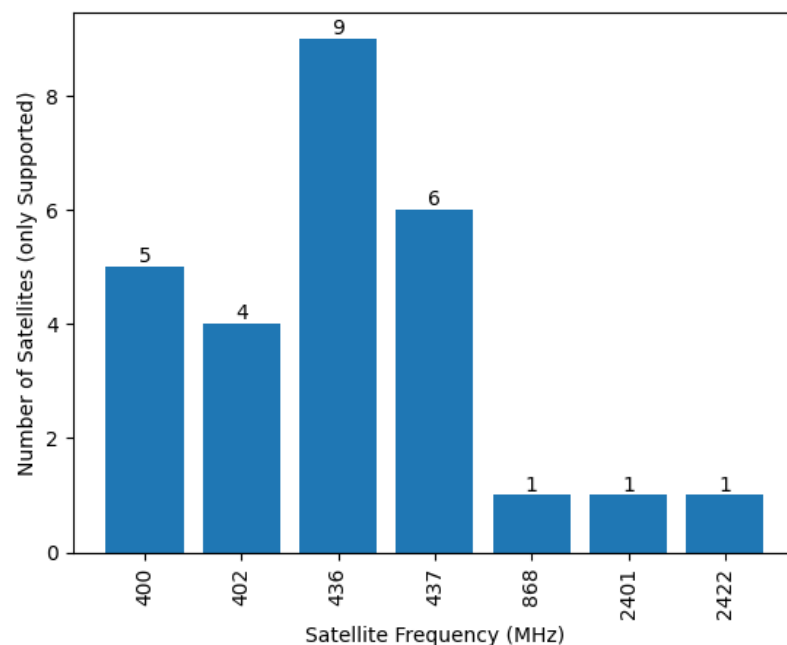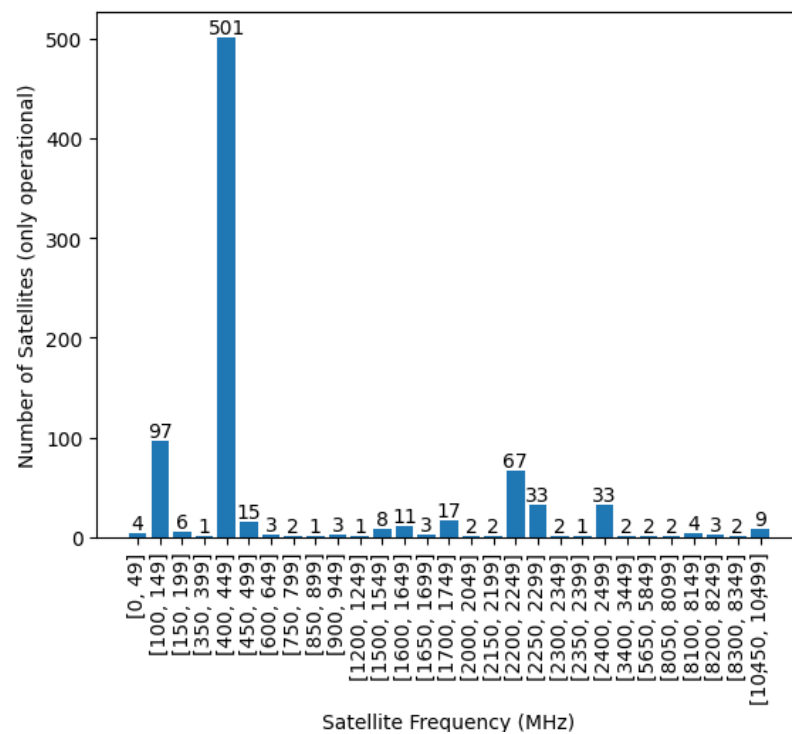

Figure 10. Frequency of satellites supported (by the TinyGS) and operational (according to SatNOGS DataBase) that had at least one data packet received by a TinyGS GS as of 29 July 2023.

In Figure 11 it is possible to see the frequencies of the satellites that were operational (according to SatNOGS DataBase) on 29 July 2023 and that had at least one data packet received by a SatNOGS GS. Note that on the horizontal axis of Figure 11 are frequency

ranges, for example, [100, 149], which covers all frequencies from 100 MHz to 149 MHz, inclusive. Therefore, a satellite communicating at 134 MHz is included in [100, 149].

To create Figure 11, the first step involved accessing all satellites and examining their transmitters. Each transmitter has various parameters, including the downlink frequency, which characterizes the frequency at which the satellite transmits the signal and the saved frequency. However, some satellites have multiple transmitters, and in such cases, downlink frequencies were kept without repetitions. For instance, if a satellite had three transmitters with the same downlink frequency (146 MHz), even if they used different downlink modes (BPSK, USB), the 146 MHz frequency was saved only once and not three times. Furthermore, only the downlink frequencies of transmitters with "alive" set to "True" [11] were saved, meaning malfunctioning transmitters with "alive" set to "False" were not considered. It is important to note that among the operational satellites, 37 either lacked transmitters in the database or possessed only inactive transmitters (e.g., were malfunctioning), and therefore, these satellites do not appear in this histogram.

Figure 11. Frequency of operational satellites (according to SatNOGS DataBase) that had at least one data packet received by a SatNOGS GS as of 29 July 2023.

Regarding Figure 11, it is possible to observe that most of the satellites, about 60%, work with frequencies between 400 and 450 MHz, as in TinyGS. Furthermore, about 80% of the satellites work with frequencies between 100 and 149 MHz, 400–449 MHz and 2200–2249 MHz.

5.11. Comparison of the Number of Satellites that Were Operational and Were Able to Send Data to the GSs and that Data was Decoded as of 29 July 2023

The next step was to observe the operational satellites that had their data decoded, because the vast majority of people who intend to join and/or contribute to the projects also want to see the content of the packages sent by the satellites.

To carry out this aim, the reasoning used in the previous sections was applied, but additionally, it was necessary to go through the satellites and see if at least one piece of data was decoded. In TinyGS, to see which satellites have decoded data, the various satellites were examined individually in the web application, and it was determined whether at least one packet was decoded. In SatNOGS, the statistics present in the SatNOGS DB [15] were

consulted, and satellites with at least one decoded data packet were saved. The numbers obtained for each category were:

- SatNOGS: 73;
- TinyGS: 22.

Taking into account these numbers and those presented above, it is possible to observe that most of the satellites supported by TinyGS and operational have decoded data. In SatNOGS, this is not the case, and the count went from 611 to 73, thus showing that, for a large majority of satellites, the decoder does not exist or has not yet been implemented. However, the SatNOGS number of continues to be higher than that of TinyGS, about three times greater. This is due to the fact that TinyGS works only with LoRa satellites.

5.12. Observation of the Frequencies of the Satellites that Were Operational and Managed to Send Data to the GSs and that Data Was Decoded as of 29 July 2023

In Figure 12, one can find the frequencies of the TinyGS-supported satellites operational according to SatNOGS DB and which had their data decoded. As one can see, most satellites work with frequencies around 400, 402, 436, and 437 MHz.
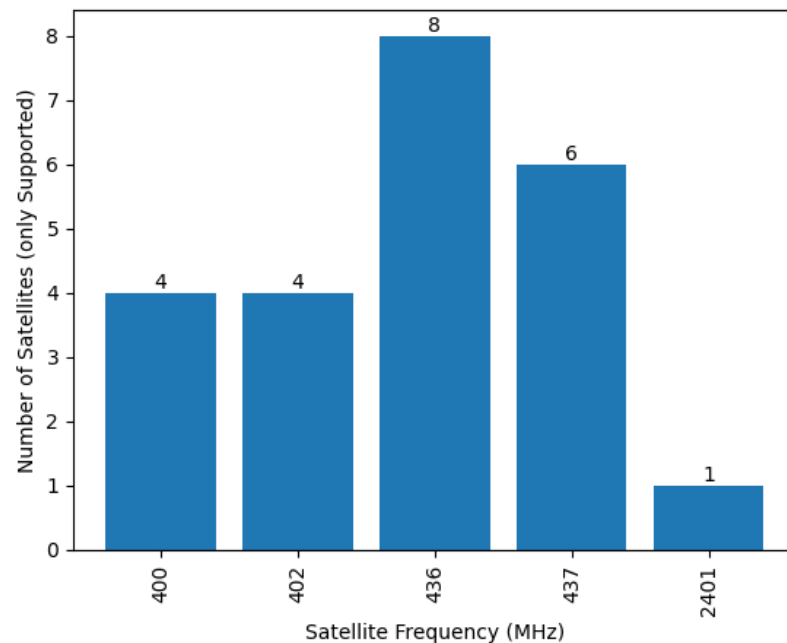
Figure 12. Frequency of satellites supported by TinyGS and operational (according to SatNOGS DataBase) that had at least one data packet received and decoded as of 29 July 2023.

With regard to SatNOGS, the same reasoning was used as in the section that analyzed the various transmitters without decoded data in that only the frequencies (downlink frequencies) of the "alive" transmitters were saved, etc. The only difference is that now, only the satellites with decoded data were studied. In Figure 13, one can find the frequencies of the satellites with at least one decoded data packet, and, as in TinyGS, the frequencies with the highest number of satellites are 436 and 437 MHz.
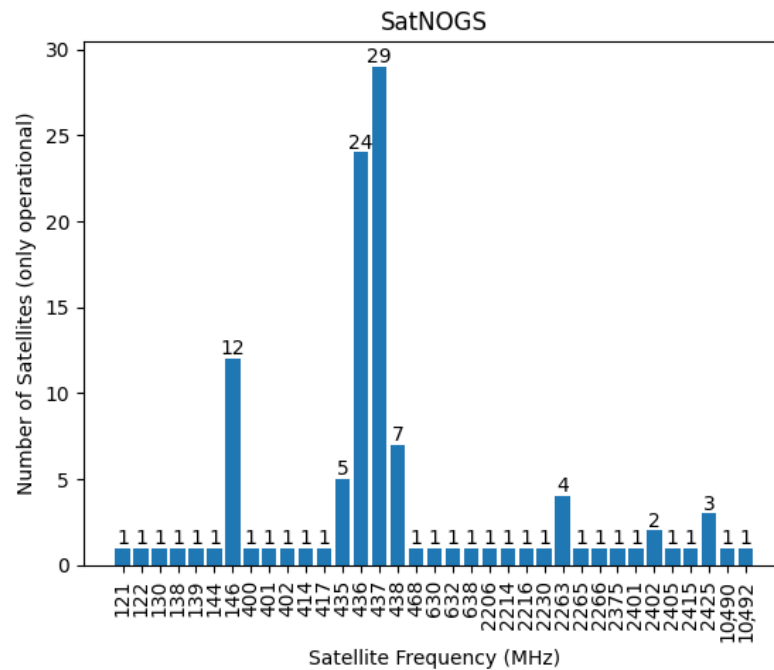
Figure 13. Frequency of SatNOGS operational satellites (according to SatNOGS DataBase) that had at least one data packet received and decoded as of 29 July 2023.

5.13. Comparison of the Number of TinyGS Satellites (Operational and Supported by TinyGS) that SatNOGS was also Able to Communicate with and Decode as of 29 July 2023

The purpose of this section is to determine what percentage of TinyGS satellites SatNOGS was also able to communicate with and decode. For this analysis, the operational satellites supported by TinyGS were used as a basis. To find out if SatNOGS was able to communicate with or decode data from a certain satellite, the SatNOGS DB [15] was consulted again.

Initially, of the operational satellites and supported by TinyGS, the number of satellites with data received by SatNOGS GSs was observed. It is important to note that the data includes decoded and non-decoded data. As we saw earlier, TinyGS was able to receive data from 25 satellites, but SatNOGS only received data from 12 of those, that is, of the 25 satellites, only 48% of them also had data received by SatNOGS GSs.

Then, from the operational satellites supported by TinyGS, the number of satellites with data received and decoded by SatNOGS was observed. As was seen earlier, TinyGS was able to decode data from 22 satellites, but SatNOGS, from those 22, could not decode any. This is because SatNOGS did not have a decoder for any of them.

5.14. Comparison of the Growth in Number of GSs Over Time

Since, in TinyGS, it is not possible to access the database, there is no API, and only the GSs that were online in the last 24 h are available on the web application, it was not possible to observe the growth in the number of TinyGS GSs over time.

However, for SatNOGS, this was possible due to the existence of the API. To carry out this study, the SatNOGS network API [10] was used and the "created" parameter of each GS was studied, which indicates the date and the time when the GS was created.

In Figure 14, a quarterly study is presented where it is possible to observe the number of SatNOGS GSs created from 30 September 2015 to 30 June 2023. As one can see, from the last quarter of 2018, the graph has an approximately linear behavior.
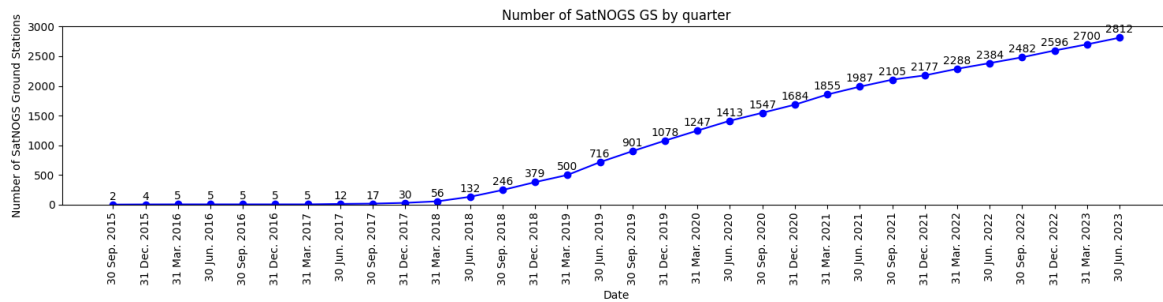
Figure 14. Number of SatNOGS GSs by quarter from 30 September 2015 to 30 June 2023.

It is important to clarify that although this graph shows the SatNOGS created up to 30 June 2023, not all of them are currently in operation. Some of them have been offline for more than a year. However, the purpose of this figure is to show that, over time, this project has had more and more members and more people building GSs.

5.15. Limitations and Implications

In this section, some limitations of the prospective projects will be discussed.

Regarding the limitations of TinyGS, the first is related to the fact that it can only work with one frequency, that is, the frequency of the LoRa module. On the other hand, with SatNOGS, there is an SDR allowing one to receive a wide range of frequencies. The second limitation is the fact that it only works with satellites that are in its database, and although this is its main purpose, it is a relatively restrictive approach. Other limitations are the lack of an API to collect the data and the tie to the Telegram user. So, if you lose your Telegram account, you will not be able to connect a new account to the old one. Instead, you will need to request new MQTT credentials, and you will have to exchange the old MQTT credentials for the new ones. This means that you will lose your GS history. In relation to the web application, one's access can be blocked if, for example, one refreshes the pages too many times to visualize the reception of a package. Consequently, one will not be able to access the data of the web application, and if one tries to access it while it is blocked, that blocking time resets. Additionally, the TinyGS software can not control ground stations with rotators. The incorporation of this capability would enable the use of directional antennas, known for their higher gain, and consequently, their capability to receive weaker signals. Finally, another limitation relates to the project documentation. Although the documentation for building and operating a GS is sufficient, mainly due to its ease, it does not exist for other aspects. For example, there is no documentation explaining how the TinyGS network works, how to add a decoder, or how to add a satellite. On the other hand, with SatNOGS [17], there is a lot of documentation from the previous examples and many others.

With respect to SatNOGS, not many limitations were found. The first is related to the fact that one can only manually update the client. In certain circumstances, such as the GS being in a location far from one's home, if, for some reason, one needed to update the client, one would need to go to that location. Additionally, if new satellites were eventually supported, it would be necessary to update the SatNOGS client to be able to communicate with them. Therefore, it would be useful for the system to conduct these updates automatically, as TinyGS does. The second limitation is related to the number of decoders. As was previously observed, SatNOGS managed to communicate with several satellites; however, it only managed to decode a small portion of them. As such, it would be good in the future to increase the number of decoders to thereby increase the quantity of data available and, consequently, attract a greater number of people to the project. However, it is worth mentioning that many decoders are not made available by the satellite teams on purpose because the data are confidential. An additional constraint arises from SatNOGS relying on the libraries it utilizes, like SoapySDR (for SDRs) and Hamlib (for rotors). As a result, the components integrated into its ground station will be subject to the limitations

of the devices supported by these libraries. For instance, if one possesses a rotor that Hamlib does not support, it is a definite or nearly certain fact that its ground station will not operate as intended. In addition, it can also communicate only with satellites that are in its database.

Finally, and related to the security implications of both projects, there are at least three limitations that merit careful consideration:

1. Public exposure of ground stations: One significant security concern arises from the public exposure of latitude and longitude coordinates associated with ground stations (GSs). This exposure essentially discloses the approximate locations of GSs, potentially including the residences of individuals operating these stations;

2. Data availability: For organizations developing satellites with confidential data and utilizing these projects due to limited financial resources for establishing a network of ground stations, it is imperative to recognize that the data will be made public. Even if the decoder is not readily accessible, it is crucial to understand that encrypted data becomes more susceptible to individuals attempting decryption efforts;

3. Open-Source Code and Increased Vulnerability: The open-source nature of the code powering these projects introduces a unique set of security considerations. Although open-source projects often benefit from community scrutiny, making it easier to detect and fix vulnerabilities, they can also be more susceptible to attacks. The transparency of the code may expose potential weaknesses, requiring constant vigilance and rapid response to address identified vulnerabilities.

6. Conclusions

Involvement in one of these projects can be ideal for certain entities, whether enthusiasts or satellite developers. There are various reasons to join one of these projects, primarily due to the numerous implications associated with implementing a network of ground stations. For instance, the costs associated with purchasing, installing, and maintaining multiple ground stations as well as identifying and acquiring suitable locations for ground stations can be significant. Regular maintenance, such as hardware repairs and software updates for these ground stations located at different sites can be challenging, involving lengthy trips. Additionally, establishing a network infrastructure for data processing, storage, and analysis may require the procurement and creation of servers and software applications, respectively.

Concerning the decision of which project to engage in, the reader should consider their objectives and circumstances carefully, scrutinize the analyses presented in this article, and subsequently make a decision. For example, if one is new in this field, does not have much knowledge, and wants to join a project, the best choice is probably TinyGS, since building and configuring the GS is relatively easier from the author's perspective. If the objective is to communicate with satellites that are not supported by TinyGS, for example, NOAA 15, then the choice should fall on SatNOGS. If the cost is the most important parameter and one does not have a preference on the type of satellites one wants to communicate with, then TinyGS is probably a better option. If one only intends to capture a large number of satellites, they can consult the frequencies most commonly used by satellites and design their ground station with the most-utilized frequency band.

The final choice between the two projects will always depend on one's goals and intentions. In light of this consideration, Table 5 has been compiled to delineate the main strengths and weaknesses of each project, thus aiding the decision-making process. However, it is important to note that both projects offer ample opportunities for learning and enjoyment.

Table 5. Main strengths and weaknesses of TinyGS and SatNOGS.

Project	Strength	Weakness
TinyGS	Low Difficulty/Time *	Lack of Documentation
	Low Cost	Low n° of satellites capable of communication
	Good geographical distribution of GSs **	Only communicates with LoRa satellites
	Open-source	Limited by the satellites in the database
	Support and n° of members	Unable to control motors/rotators for antennas ***
	High n° of GSs online **	Low range of frequencies ****
SatNOGS	Good Documentation	Manually updated
	Low Cost *****	Limited by the satellites in the database
	High n° of satellites capable of communication	Low percentage of satellites with decoded data compared to those successfully communicated with
	Open-Source	-
	Support and n° of members	-
	Able to control motors/rotators for antennas ***	-
	The user can schedule observations on other users' GSs	-
	Wider range of frequencies ******	-

* It is presumed that the reader chooses to exclusively utilize commercial components. ** Based on data from 29 July 2023. *** Useful to use directional antennas to receive weaker signals. **** The GS is limited by the LoRa module frequency. ****** Since it uses an SDR.

In terms of future endeavors, the following initiatives are suggested: updating the provided data is imperative, encompassing ground station information such as geographical distribution and frequency bands as well as satellite data due to the emergence of new satellites and the deactivation of others; conducting a comprehensive study of the online presence of ground stations over time, aiming to discern whether these projects are experiencing an increase, decrease, or maintaining a consistent level of activity; investigating the geographical distribution based on the most-utilized frequencies of the ground stations within each project with the goal of assisting satellite developers in making informed decisions regarding the selection of downlink frequencies for their satellite communication; conducting a comprehensive and more detailed examination of both software platforms, scrutinizing their features, strengths, and areas for improvement with the aim of providing a nuanced understanding of the capabilities and limitations of each software; based on the recommended components mentioned earlier, undertaking a more detailed investigation into the hardware employed in ground stations to identify opportunities for cost-effective improvements in performance with the aim of optimizing the capabilities of the ground station through improvements to the hardware infrastructure.

Author Contributions: Conceptualization, A.F.d.S.; Data curation, J.S.G.; Formal analysis, J.S.G.; Funding acquisition, A.F.d.S.; Investigation, J.S.G. and A.F.d.S.; Methodology, A.F.d.S. and J.S.G.; Supervision, A.F.d.S.; Writing - original draft, J.S.G.; Writing - review and editing, J.S.G., A.F.d.S. All authors have read and agreed to the published version of the manuscript.

Data Availability Statement: During the article's development, all TinyGS data including GS details and satellite information were collected from TinyGS's web application [23]. However, for SatNOGS, the GSs' details were obtained from the SatNOGS network API [10], and the satellite details were

retrieved from the SatNOGS DB API [11]. GS data encompassed their coordinates ([latitude, longitude]), frequency ranges, creation date, and the last time that GS was online (only for SatNOGS), while satellite data covered operational status, downlink frequency, and the number of data received and decoded by GSs from that specific satellite. The data concerning the last time the GS (ground station) was online (e.g., 5 October 2022 12:49:26) were crucial to determining whether the SatNOGS GSs were online within the last 24 h on 29 July 2023, and consequently, were used to carry out Topics 6, 7, and 8 mentioned in the Methodology section. In TinyGS, these data were not needed because, at the time of writing (August 2023), all the GSs listed in the TinyGS web application were online within the last 24 h. Coordinates (e.g., [38.0169, 23.731] in degrees) and the frequency range of the GSs (e.g., [400,460] MHz) were used exclusively in Topics 7 and 8, respectively, to observe the geographical distribution and frequency bands. The last piece of data regarding GSs, the creation date (e.g., 22 July 2015), was used in Topic 14 to observe the number of GSs from corresponding projects over time. Regarding satellites, the operational status aims to indicate whether a satellite is on the network satellite list so that the network can configure GSs to observe and receive data from that satellite. This variable takes only two values, "True" or "False", and was used in Topics 9, 10, 11, and 12. In turn, the numbers of data packets received and decoded by GSs from particular satellites are integers (e.g., 925,348 and 827,571, respectively) and were used in Topics 9, 10, 11, 12, and 13. Finally, the downlink frequency of satellites, also referred to as the satellite frequency (the frequency at which the satellite is transmitting the signal and the ground station is receiving it, for example, 436.703 MHz), was used and rounded to whole units in Topics 10 and 12 for histogram creation. It is important to note that some satellites had more than one downlink frequency, for example, they could transmit signals at frequencies of 868.100 MHz and 2422.000 MHz, and in that case, both frequencies were considered.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ADC	Analog-to-Digital converter
API	Application Programming Interface
DC	Direct Current
FSK	Frequency-Shift Keying
GEO	Geostationary Orbit
GFSK	Gaussian Frequency Shift Keying
GS	Ground Station
LEO	Low Earth Orbit
LNA	Low Noise Amplifier
MQTT	Message Queuing Telemetry Transport
NOAA	National Oceanic and Atmospheric Administration
QFH	Quadrifilar Helix
SatNOGS	Satellite Networked Open Ground Station
SDR	Software-Defined Radio
SSID	Service Set Identifier
UHF	Ultra High Frequency
VHF	Very High frequency

References

1. UCS Satellite Database. Available online: https://www.ucsusa.org/resources/satellite-database?_ga=2.176708386.663254410.1627310469-1065517005.1626146107&_gl=1*1in2mrm*_ga*ODI0MTM4Nzk5LjE2ODA3OTQxNDI.*_ga_VB9DKE4V36*MTY4MDc5NDE0Mi4xLjAuMTY4MDc5NDE0Ny4wLjAuMA (accessed on 26 June 2023).
2. Satellite Built by Brown Students as Low-Cost Way to Reduce Space Junk Reenters Atmosphere Years Early. Available online: https://www.brown.edu/news/2023-08-18/sbudnic-reentry (accessed on 29 January 2024).
3. Downey, F.; Buchan, S.; Hartig, B.; Busan, D.; Cook, J.; Howie, R.; Bland, P.; Paxman, J. Binar Space Program: Binar-1 Results and Lessons Learned. In Proceedings of the 36th Annual Small Satellite Conference, Logan, UT, USA, 6–11 August 2022.

4. Bogomolov, A.V.; Bogomolov, V.V.; Iyudin, A.F.; Eremeev, V.E.; Kalegaev, V.V.; Myagkova, I.N.; Osedlo, V.I.; Petrov, V.L.; Peretjat'ko, O.Y.; Prokhorov, M.I.; et al. Space Weather Effects from Observations by Moscow University Cubesat Constellation. *Universe* **2022**, *8*, 282.

5. Choi, T.; Stevenson, T.; Lightsey, G. Reference Ground Station Design for University Satellite Missions with Varying Communication Requirements. In Proceedings of the 55th AIAA Aerospace Sciences Meeting, Grapevine, TX, USA, 9–13 January 2017.

6. Baeza, V.M; Ortiz, F.; Lagunas, E.; Abdu, T.S.; Chatzinotas, S. Multi-Criteria Ground Segment Dimensioning for Non-Geostationary Satellite Constellations. In Proceedings of the 2023 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit), Gothenburg, Sweden, 6–9 June 2023; pp. 252–257.

7. Baeza, V.M; Ortiz, F.; Lagunas, E.; Abdu, T.S.; Chatzinotas, S. Gateway Station Geographical Planning for Emerging Non-Geostationary Satellites Constellations. *IEEE Netw.* **2023**, 1–9. https://doi.org/10.1109/MNET.2023.3321531.

8. SatNOGS – Open Source Global Network of Satellite Ground-Stations. Available online: https://satnogs.org/ (accessed on 20 April 2023).

9. TinyGS Github. Available online: https://github.com/G4lile0/tinyGS (accessed on 20 April 2023).

10. SatNOGS Network API. Available online: https://network.satnogs.org/api/ (accessed on 29 July 2023).

11. SatNOGS DB API. Available online: https://db.satnogs.org/api/ (accessed on 29 July 2023).

12. Croissant, K.; White, D.; Álvarez, X.C.; Adamopoulos, V.; Bassa, C.; Boumghar, R.; Brown, H.; Damkalis, F.; Daradimos, I.; Dohmen, P., et al. (Eds.) An Updated Overview of the Satellite Networked Open Ground Stations (SatNOGS) Project. In Proceedings of the 36th Annual Small Satellite Conference, Logan, UT, USA, 6–11 August 2022.

13. White, D.J.; Giannelos, I.; Zissimatos, A.; Kosmas, E.; Papadeas, D. (Eds.) *SatNOGS: Satellite Networked Open Ground Station*; Valparaiso University: Valparaiso, IN, USA, 2015.

14. SatNOGS Network. Available online: https://network.satnogs.org/ (accessed on 22 April 2023).

15. SatNOGS DB. Available online: https://db.satnogs.org/ (accessed on 21 April 2023).

16. SatNOGS Auto Scheduler. Available online: https://gitlab.com/librespacefoundation/satnogs/satnogs-auto-scheduler (accessed on 2 August 2023).

17. SatNOGS Wiki. Available online: https://wiki.satnogs.org/Main_Page (accessed on 25 June 2023).

18. SatNOGS Rotator Controller. Available online: https://gitlab.com/librespacefoundation/satnogs/satnogs-rotator-controller (accessed on 25 July 2023).

19. SatNOGS Rotator v3. Available online: https://gitlab.com/librespacefoundation/satnogs/satnogs-rotator/-/tree/master (accessed on 10 July 2023).

20. SatNOGS Hardware Assembly Instructions. Available online: https://ohai.libre.space/group/satnogs/ (accessed on 28 July 2023).

21. SatNOGS Homebuilt Antennas GitLab. Available online: https://gitlab.com/librespacefoundation/satnogs/satnogs-antennas/-/tree/master (accessed on 28 July 2023).

22. SatNOGS Diplexer. Available online: https://gitlab.com/librespacefoundation/satnogs/satnogs-diplexer (accessed on 26 June 2023).

23. TinyGS Web App. Available online: https://tinygs.com (accessed on 3 July 2023).

24. LoRaWAN® Regional Parameters RP002-1.0.4. Available online: https://resources.LoRa-alliance.org/document/rp002-1-0-4-regional-parameters (accessed on 28 June 2023).

25. SatNOGS g5500-Ardushield. Available online: https://gitlab.com/librespacefoundation/satnogs/g5500-ardushield (accessed on 3 August 2023).

26. Libre Space Community. Available online: https://community.libre.space/ (accessed on 16 July 2023).