Development of optimized methods to Quality Control GNSS Observations

Diogo Miguel Rodrigues Ventura

Relatório de Estágio para obtenção do grau de Mestre em

Engenharia Informática

(2º ciclo de estudos)

Orientador: Prof. Doutor Paul Crocker

Orientador na empresa MIRASpaco: Prof. Doutor Rui Fernandes

outubro de 2024

Declaração de Integridade

Eu, Diogo Ventura, que abaixo assino, estudante com o número de inscrição m12596 de Engenharia Informática da Faculdade de Engenharias, declaro ter desenvolvido o presente trabalho e elaborado o presente texto em total consonância com o Código de Integridades da Universidade da Beira Interior.

Mais concretamente afirmo não ter incorrido em qualquer das variedades de Fraude Académica, e que aqui declaro conhecer, que em particular atendi à exigida referenciação de frases, extratos, imagens e outras formas de trabalho intelectual, e assumindo assim na íntegra as responsabilidades da autoria.

Universidade da Beira Interior, Covilhã 11/10/2024

Diogo Ventura

Agradecimentos

Gostaria de expressar minha profunda gratidão a todos que contribuíram para a realização deste estágio curricular e para a elaboração deste relatório.
Em primeiro lugar, aos meus orientadores, Paul Crocker e Rui Fernandes, pela orientação, paciência e sabedoria compartilhada ao longo deste estágio. Suas orientações foram essenciais para meu aprendizado e desenvolvimento profissional.
Agradeço também aos meus pais e à minha família, cujo apoio incondicional e encorajamento foram fundamentais em toda a minha jornada acadêmica e pessoal. Vocês sempre acreditaram em mim e me motivaram a buscar meus objetivos.
Um agradecimento especial à MIRASpaco, que me proporcionou a oportunidade de realizar meu estágio. A experiência adquirida neste ambiente foi inestimável e contribuiu significativamente para o meu crescimento.
Um agradecimento também ao Fernando Geraldes, pela ajuda dentro da empresa.
Agradeço igualmente aos meus amigos, que sempre estiveram ao meu lado, oferecendo apoio e momentos de descontração que tornaram essa experiência ainda mais especial.
Por último, mas não menos importante, gostaria de dedicar um agradecimento especial à minha namorada, Patrícia Mateus, pela compreensão e apoio contínuo.
A todos vocês, meu sincero muito obrigado!

Resumo

O presente relatório descreve o estágio realizado na empresa MIRASpaco, com foco no desenvolvimento de procedimentos de controle de qualidade para os ficheiros RINEX distribuídos pelo software MIRAnet. Este software, desenvolvido pela MIRASpaco, é utilizado para a distribuição de ficheiros RINEX aos seus utilizadores, e o estágio teve como principal objetivo a implementação de funcionalidades que permitam a monitorização dos dados de estações GNSS por meio desses ficheiros.
O relatório inicia-se com uma apresentação da empresa e uma descrição detalhada dos objetivos do estágio. Em seguida, são abordados conceitos fundamentais do sistema GNSS, essenciais para a compreensão do MIRAnet e do trabalho a ser realizado. O documento também inclui um estudo sobre o estado da arte em aplicações de controle de qualidade para ficheiros RINEX, com uma análise do software G-Nut/Anubis, que foi selecionado para ser utilizado neste projeto. Além disso, o relatório apresenta um capítulo dedicado à engenharia de software, onde são especificados os requisitos do sistema, a arquitetura do software e o esquema da base de dados necessários para o desenvolvimento do projeto. As tecnologias utilizadas são discutidas para contextualizar o desenvolvimento do projeto que é descrito no penultimo capítulo.
Por fim, o relatório conclui que todos os objetivos definidos para o estágio foram atingidos com sucesso, resultando na implementação de um procedimento eficaz de controle de qualidade para os ficheiros RINEX no MIRAnet.

Palavras-chave

MIRASpaco, MIRAnet, GNSS, RINEX, Controlo de Qualidade, G-Nut/Anubis

Resumo alargado

MIRAnet constitui o software desenvolvido pela MIRASpaco para a distribuição de ficheiros RINEX aos seus utilizadores. No seu compromisso contínuo de aprimorar esse software, a MIRASpaco está ativamente envolvida na incorporação de funcionalidades de controlo de qualidade.

O presente relatório descreve o estágio realizado na empresa MIRASpaco, cujo principal objetivo foi o desenvolvimento de procedimentos de controlo de qualidade e a implementação de funcionalidades para monitorização dos dados de estações GNSS, utilizando os ficheiros RINEX distribuídos pelo software MIRAnet.

A criação de ficheiros RINEX são a última etapa de um processo complexo de aquisição de dados GNSS. Começando por constelações de satélites que enviam sinais a estações terrestres, que guardam observações GNSS em ficheiros com formatos proprietários dos fabricantes, que por fim são convertidos no formato RINEX que é usado globalmente como o formato standard para partilhar dados GNSS.

Os ficheiros RINEX podem ser de baixa qualidade devido a vários fatores como avarias na antena, ruído e problemas de rede durante a transmissão dos dados. Estes fatores refletem-se principalmente na qualidade dos dados presentes no ficheiro, mas os ficheiros podem ter também problemas relacionados com a sua estrutura, nomeadamente inconsistências nos nome do ficheiro, que deve seguir as normas da versão RINEX a ser utilizada, e inconsistências nos metadados do ficheiro.

Para realizar operações de controlo de qualidade nestes ficheiros foi utilizado o software G-Nut/Anubis que é considerado o software mais competente para o efeito. O software permite guardar dados de controlo de qualidade que podem alertar para problemas existentes com as estações GNSS. Usando estes dados, foram gerados gráficos que permitem a visualização temporal de determinadas métricas obtidas dos ficheiros RINEX, que permite a monitorização das estações.

O presente relatório descreve detalhadamente os passos tomados durante o desenvolvimento dos objetivos. É feita uma apresentação de diversos componentes de engenharia de software, seguido de um capítulo descritivo do desenvolvimentos dos diferentes scripts criados para atingir os objetivos.

Por fim, o relatório conclui que todos os objetivos definidos para o estágio foram atingidos com sucesso, resultando na implementação de um procedimento eficaz de controle de qualidade para os ficheiros RINEX no MIRAnet.

Abstract

This report describes the internship carried out at MIRASpaco, focusing on the development of quality control procedures for RINEX files distributed through the MIRAnet software. This software, developed by MIRASpaco, distributes RINEX files to its users. The internship's primary objective was to implement functionalities enabling GNSS station monitoring through these files.

The report begins with an introduction to the company and a detailed description of the internship objectives. It then covers fundamental concepts of the GNSS system, essential for understanding MIRAnet and the work to be done. The document also includes a study on the state of the art in quality control applications for RINEX files, with an analysis of the G-Nut/Anubis software, which was selected for use in this project.

Additionally, the report presents a chapter dedicated to software engineering, where system requirements, software architecture, and the database schema necessary for the project's development are specified. The technologies used are discussed to provide context for the project's development, which is detailed in the penultimate chapter.

Finally, the report concludes that all the objectives set for the internship were successfully achieved, resulting in the implementation of an effective quality control procedure for RINEX files in MIRAnet.

Keywords

MIRASpaco, MIRAnet, GNSS, RINEX, Quality Control, G-Nut/Anubis

Contents

List of Figures

List of Tables

Lista de Acrónimos

UBI	Universidade da Beira Interior
GNSS	Global Navigation Satellite System
CORS	Continuously Operating Reference Station
RINEX	Receiver Independent Exchange Format
EPOS	European Plate Observing System
GPS	Global Positioning System
GLONASS	GLObalnaya NAvigatsionnaya Sputnikovaya Sistema
BeiDou	the Big Dipper Constellation
SNR	signal-to-noise ratio
QC	Quality Control
GUI	Graphical user interface
CLI	Command line interface
UNAVCO	University Navstar Consortium
DB	Database
UI	User interface
HTML	Hyper Text Markup Language
CSS	Cascading Style Sheets
PHP	Hypertext Preprocessor
DOY	Day of year

Chapter 1

Introduction

1.1 Company description - MIRASpaco

MIRASpaco [1] is a spin-off of the University of Beira Interior (UBI) [2] founded by Rui Fernandes in 2021. The company is focused on Geomatics and Geoinformatics and is hosted by the Faculty of Engineering.
Although it is a young company, the MIRASpaco team has already carried out projects with international companies such as RxNetworks (Canada) [3].
MIRASpaco has been connecting GNSS's [4] scientific and technical worlds by developing and installing GNSS CORS hardware and software in regions with adverse logistics. The developed systems permit the installation of geodetic systems that are completely autonomous in terms of power and use low-cost but efficient solutions to transmit data and access the systems remotely.
The company provides many services like installation and rehabilitation of GNSS CORS networks and estimation of accurate coordinates with respect to global and national reference frames.
Other service groups are vertical positioning services, geophysical surveys, and geo-IT services. As for geo-IT services, MIRASpaco offers many products like:

- MIRAnet: Management of GNSS networks using web services.
- MIRAcoor: Automatic and dedicated estimation coordinates with respect to global and national referential.
- MIRArouter: Integrated software/hardware solutions for remote devices.
- Automatization of procedures to estimate geo-products.

Lastly MIRASpaco also provides training and formation as these are fundamental components to advance any organization, especially for MIRASpaco given its academic background.

1.2 MIRAnet

MIRAnet is a software package developed by MIRASpaco that aims to monitor the status of CORS stations and store and disseminate the acquired GNSS observations using RINEX files.
It has a front page (see Figure 1.1) that allows users to authenticate as administrators or regular users.

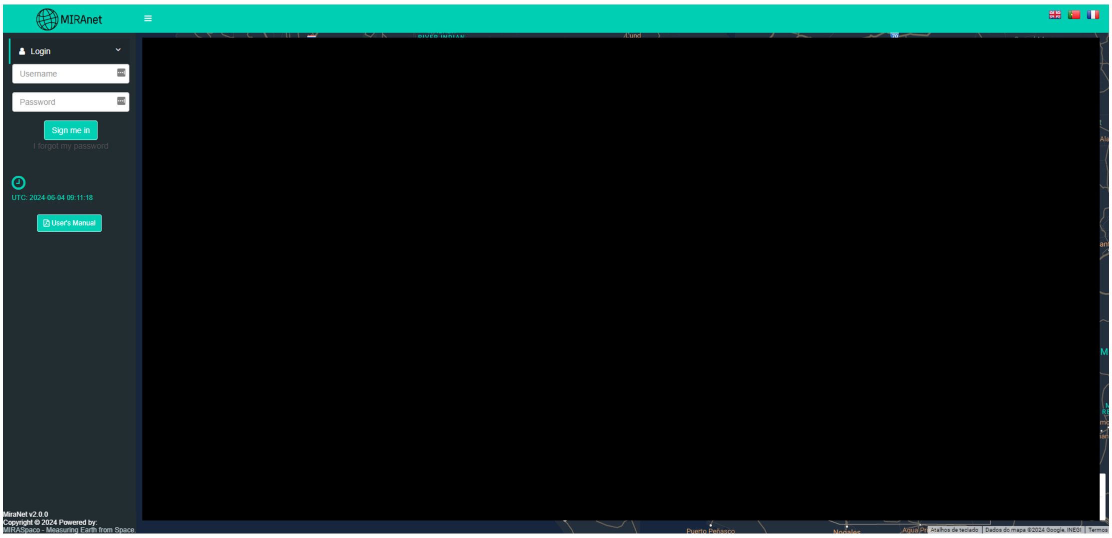


Figure 1.1: MIRAnet home page

Regular users typically have limited access to MIRAnet functionalities. However, administrators possess the authority to assign customizable permissions to each user. The front office (see Figure 1.2) represents the most basic level of user access, featuring a simple interface that allows users to download RINEX files for various network stations and obtain information for each station.

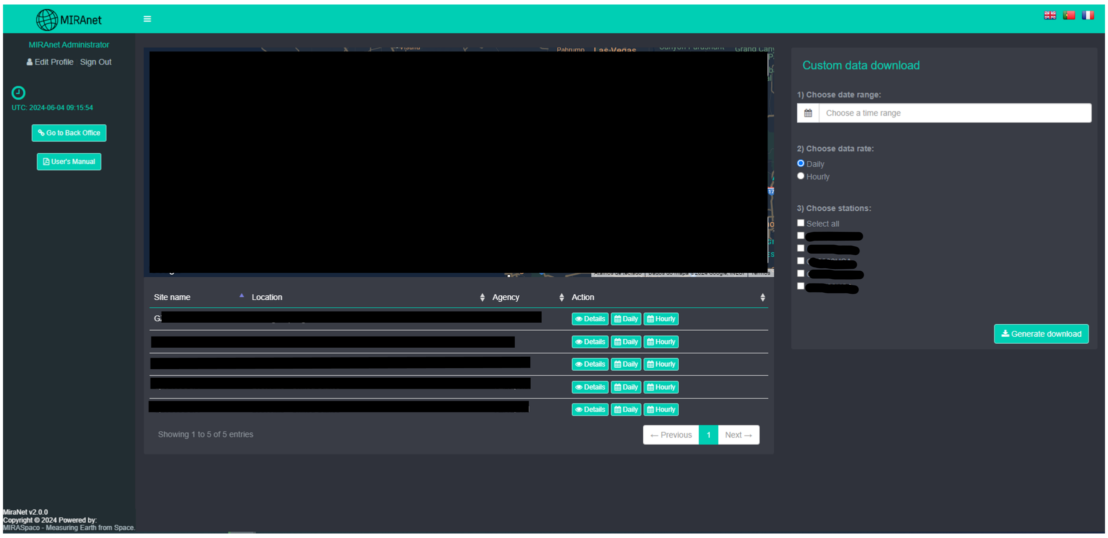


Figure 1.2: MIRAnet front office

Higher permission levels allow access to the back office (see Figure 1.3). In addition to all front office functionalities, this access permits the utilization of administrative features, such as editing station and network metadata, creating and managing user accounts, modifying various MIRAnet settings, and monitoring the network.

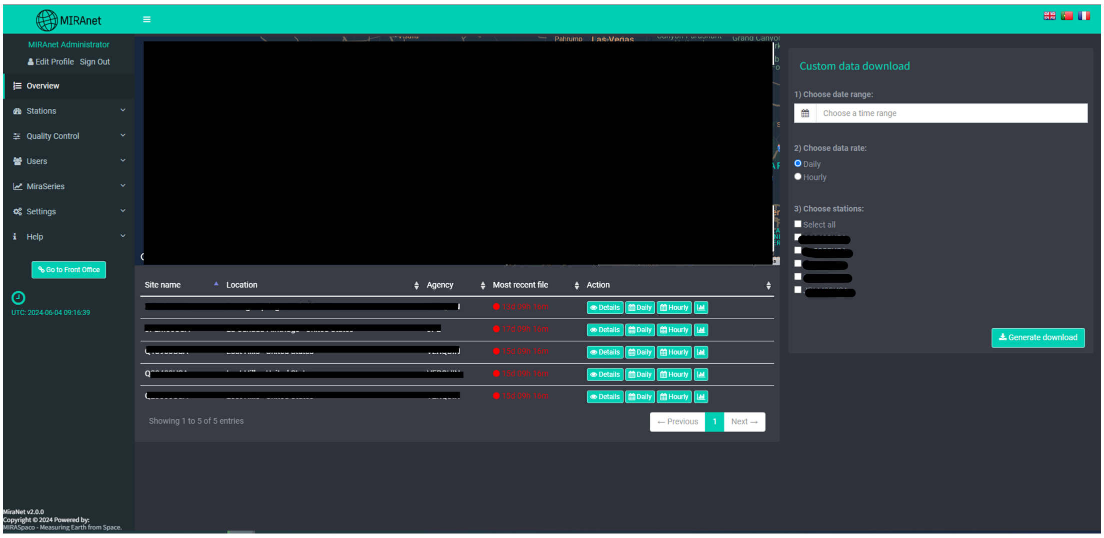


Figure 1.3: MIRAnet back office

1.3 Motivation

CORS (Continuously Operating Reference Stations) acquire GNSS observations that are stored in proprietary format files, commonly referred to as raw files, that are converted in RINEX (Receiver Independent Exchange Format) [5] to be disseminated for scientific and technical applications. Assessing the quality of these files is a key component of the distribution software package since it needs to provide the end-user with the guarantee that the data are correct and consistent with the metadata of the station.

MIRAnet 1.2 constitutes the software package developed by MIRASpaco for the distribution of RINEX files to its users, as well as the management and monitoring of GNSS networks. In its ongoing commitment to enhance this software, MIRASpaco was actively engaged in incorporating a quality control feature. This strategic initiative aims to attract more clients and enhance satisfaction levels among existing users.

1.4 Objectives

The implementation at MIRAnet of optimized quality control procedures is essential to guarantee proper station monitoring and data quality. Currently, the most advanced application for Quality Control (QC) GNSS observations is G-Nut/Anubis[6]. The optimal integration of this software in the MIRAnet package will be the ultimate goal of this project.

The first step of this work was to modify the existing MIRAnet database to be able to store the information generated by Anubis about each created file.

Another component was the creation of efficient procedures to interact with the MIRAnet administrators (i.e., the person(s) responsible for managing the CORS stations). Simple but efficient procedures must be implemented to allow any abnormal situation to be quickly reported. The generation of clear graphics showing the estimated QC parameters will be fundamental to identifying any potential problem.

The developed procedures can also be later applied in scientific projects where UBI is involved, particularly in EPOS (European Plate Observing System) [7].

Chapter 2

General Concepts and Background

2.1 Introduction

In this chapter, general concepts related to GNSS are discussed and explained. These concepts will be crucial for the overall document and the implementation of the solution.

2.2 GNSS Satellites and Constellations

The Global Navigation Satellite System (GNSS) comprises satellites orbiting the Earth, transmitting encoded messages on various radio-frequency carriers. This enables GNSS receivers on the ground to determine their locations with different levels of accuracy by detecting, decoding, and processing signals from the satellites.

These satellites are organized into constellations managed by different entities. The main GNSS constellations are GPS (USA) [8], GLONASS (Russia) [9], GALILEO (EU) [10], and BeiDou/Compass (China) [11]. Each GNSS constellation consists of 24 to 35 satellites, ensuring that at least four satellites are visible at all times to be able to calculate a precise position for a GNSS station.

2.3 GNSS and CORS stations

A CORS [12] (Continuously Operating Reference Station) (see Figures 2.1 and 2.2) is a station that continuously provides positioning information by collecting GNSS data and transmitting it to users via the Internet. These stations can also offer additional data, such as meteorological parameters, obtained from co-located instruments. CORS stations are typically organized into networks, allowing for combining data from multiple stations to enhance GNSS coverage and improve accuracy.

In addition to CORS, GNSS stations not part of the CORS network are called campaign stations. These stations are set up for specific projects for a limited period and do not operate continuously.

Figure 2.1: Photograph of an antenna from a GNSS station installed at UBI

Figure 2.2: Photograph of an integrated GNSS System for precise applications

2.4 Raw GNSS data files

GNSS data collected by receivers from various brands are typically stored in proprietary binary files, which can only be processed using specialized software. These files are commonly known as 'Raw files' and consist of the receiver's observation data and the broadcast orbit information of the tracked satellites, like satellite position, speed, and time [13]. However, the specific file extension often indicates the brand of the receiver that generated the file.

In Table 2.1 are some examples of existing receiver brands and their possible file extensions:

Receiver Brand	File Extensions
Septentrio	.sbf, .xx_
Leica	.mdb
Trimble	.T04, .T02, .dat
ComNav Technology	.cnb
NovAtel	.bin

Table 2.1: Examples of receiver brands and their file extensions

2.5 RINEX files

Receiver Independent Exchange (RINEX) [5] files are the community standard for the distribution of GNSS observation data. Converting Raw files to the RINEX format allows the GNSS scientific and technical community to use the data independently and combine multiple Raw files into one RINEX file.
As the RINEX format evolved, different versions were created tp enhance its capabilities. For example, version 3, which came after version 2.11, was created to provide generic and systematic support for all GNSS constellations, signals, and observations and a more descriptive filename. As of 2024, the most recent version is RINEX 4.1 but RINEX 3 is still the most used format. RINEX 4.1 is a big update to the file format guidelines. It's meant to make the files work well with the new navigation messages from all satellite navigation systems and include new system data messages. Table 2.2 has a summary of the different RINEX versions based on the IGS website [14].

Rinex Version	Description
RINEX v. 2.10	GPS and GLONASS observations, meteorological data, and navigation files
RINEX v. 2.11	GPS, GLONASS, and Galileo observations, meteorological data, and navigation files. Additionally, the C2, L2C/L5, and Galileo codes have also been introduced
RINEX v. 2.11-A	Rinex 2.11 format definition addendum authorizing and encouraging the use of gzip compression as approved by the IGS-RTCM RINEX Working Group and the IGS Infrastructure Committee
RINEX 3 Long Filename Creation	The RINEX 3 format defines a new naming scheme for the files.
RINEX v. 3.01	GPS, GLONASS, Galileo, BeiDou (Compass), QZSS, and SBAS, however, the structure of the data record has changed significantly with the addition to detailed characterization of actual signal generation
RINEX v. 3.02	Enhanced 3.01 to include a new header message to specify the GLONASS code-phase bias
RINEX v. 3.03	Added the Indian Regional Navigation Satellite System (IRNSS) and many other major updates and improvements
RINEX v. 3.04	The RINEX 3.04 release supports all publicly available signals.
RINEX v. 3.05	RINEX 3.05 (2020) is a major restructure and revision of the format document to make it clearer and easier to read
RINEX v. 4.00	RINEX 4.00 (2021) is a major revision of the format document to modernize the Navigation message files to be able to accommodate the new navigation messages from all the GNSS constellations.
RINEX v. 4.01	RINEX 4.01 (2022) is an upgrade of the format document that introduces new observation codes.

Table 2.2: RINEX Versions

RINEX files also have different types that the file extension can easily identify. The RINEX

2.x file extension is normally composed of three characters, with the first two being the last 2 digits of the year it was recorded, and the last character being the type of RINEX file. Some examples of RINEX types are the Observation file (.YYo) and the GPS Navigation file (.YYn). The RINEX 3 version introduced a new name and more descriptive naming convention (nnnnxxccc_s_yyyydddhhmm_ppp_iii_tt.fff.gz), Table 2.3 details the information provided by this name format.

Characters	Meaning
nnnnxxccc	station identifier, where nnnn is a four character id, xx is a number id, and ccc is a country code
yyyydddhhmm	file start time and date
ppp	file data span
iii	logging interval
tt	data type
fff	first two being the last 2 digits of the year it was recorded, and the last character being the type of RINEX file

Table 2.3: RINEX naming convention description

The RINEX format comprises more than one file, each containing different kinds of information (see Table 2.4).

File type	character identifier
The archive of observation data	O
The weather data file	M
The file with the GPS navigation message	N
The GLONASS navigation message file	G
The GLONASS navigation message file	L
Mixed GNSS navigation message archive	P
SBAS Payload Navigation Message File	H
SBAS Transmission Data File	B
Clock File	C
Summary file	S

Table 2.4: RINEX file type character identifier

Below is an example of a RINEX 3.04 Observation file (see Listing 2.1). The file is composed of two parts: first is the file header that contains the metadata: file information, observation details, time and coordinates information, correction parameters, antenna, receiver information, and additional comments.

```
     3.04           OBSERVATION DATA    M                   RINEX VERSION / TYPE
sbf2rin-15.6.1                          20230915 000220 UTC PGM / RUN BY / DATE
MMUB                                                        MARKER NAME
24208M001                                                   MARKER NUMBER
CORS                                                        MARKER TYPE
GGN                 JPL                                     OBSERVER / AGENCY
3055406             SEPT POLARX5        5.5.0               REC # / TYPE / VERS
0020                TWIVC6150       SCIS                    ANT # / TYPE
 -1245565.7486  4098163.8280  4711834.5128                  APPROX POSITION XYZ
        0.0350        0.0000        0.0000                  ANTENNA: DELTA H/E/N
G   17 C1C L1C S1C C1W S1W C2W L2W S2W C2L L2L S2L C5Q L5Q  SYS / # / OBS TYPES
       S5Q C1L L1L S1L                                      SYS / # / OBS TYPES
E   15 C1C L1C S1C C6C L6C S6C C5Q L5Q S5Q C7Q L7Q S7Q C8Q  SYS / # / OBS TYPES
```

```
       L8Q S8Q                                             SYS / # / OBS TYPES
R   15 C1C L1C S1C C1P L1P S1P C2P L2P S2P C2C L2C S2C C3Q  SYS / # / OBS TYPES
       L3Q S3Q                                             SYS / # / OBS TYPES
C   15 C1P L1P S1P C5P L5P S5P C2I L2I S2I C7I L7I S7I C6I  SYS / # / OBS TYPES
       L6I S6I                                             SYS / # / OBS TYPES
J   15 C1C L1C S1C C2L L2L S2L C5Q L5Q S5Q C1L L1L S1L C1Z  SYS / # / OBS TYPES
       L1Z S1Z                                             SYS / # / OBS TYPES
SEPTENTRIO RECEIVERS OUTPUT ALIGNED CARRIER PHASES.         COMMENT
NO FURTHER PHASE SHIFT APPLIED IN THE RINEX ENCODER.        COMMENT
G L1C                                                      SYS / PHASE SHIFT
G L2W                                                      SYS / PHASE SHIFT
G L2L  0.00000                                             SYS / PHASE SHIFT
G L5Q  0.00000                                             SYS / PHASE SHIFT
G L1L  0.00000                                             SYS / PHASE SHIFT
E L1C  0.00000                                             SYS / PHASE SHIFT
E L6C  0.00000                                             SYS / PHASE SHIFT
E L5Q  0.00000                                             SYS / PHASE SHIFT
E L7Q  0.00000                                             SYS / PHASE SHIFT
E L8Q  0.00000                                             SYS / PHASE SHIFT
R L1C                                                      SYS / PHASE SHIFT
R L1P  0.00000                                             SYS / PHASE SHIFT
R L2P  0.00000                                             SYS / PHASE SHIFT
R L2C                                                      SYS / PHASE SHIFT
R L3Q  0.00000                                             SYS / PHASE SHIFT
C L1P  0.00000                                             SYS / PHASE SHIFT
C L5P  0.00000                                             SYS / PHASE SHIFT
C L2I                                                      SYS / PHASE SHIFT
C L7I                                                      SYS / PHASE SHIFT
C L6I                                                      SYS / PHASE SHIFT
J L1C                                                      SYS / PHASE SHIFT
J L2L  0.00000                                             SYS / PHASE SHIFT
J L5Q  0.00000                                             SYS / PHASE SHIFT
J L1L  0.00000                                             SYS / PHASE SHIFT
J L1Z  0.00000                                             SYS / PHASE SHIFT
    30.000                                                 INTERVAL
  2023     9    14     0     0    0.0000000     GPS         TIME OF FIRST OBS
  2023     9    14    23    59   30.0000000     GPS         TIME OF LAST OBS
   126                                                     # OF SATELLITES
 C1C    0.000 C1P    0.000 C2C    0.000 C2P    0.000        GLONASS COD/PHS/BIS
DBHZ                                                       SIGNAL STRENGTH UNIT
 24 R01  1 R02 -4 R03  5 R04  6 R05  1 R06 -4 R07  5 R08  6 GLONASS SLOT / FRQ #
    R09 -2 R10 -7 R11  0 R12 -1 R13 -2 R14 -7 R15  0 R16 -1 GLONASS SLOT / FRQ #
    R17  4 R18 -3 R19  3 R20  2 R21  4 R22 -3 R23  3 R24  2 GLONASS SLOT / FRQ #
This data is provided as a public service by NASA/JPL.      COMMENT
No warranty is expressed or implied regarding suitability   COMMENT
for use.  For further information, contact:                 COMMENT
ggnops at jpl dot nasa dot gov                              COMMENT
                                                            END OF HEADER
```

Listing 2.1: RINEX 3.04 header

Right after the header section, there is the second part of the file, which is the observation data section (see Listing 2.2). In this section, the first line contains the epoch of the observation. Next, there is the data for each satellite at that epoch. For example, for satellite C01, the first number is the pseudorange in meters. The following ”6” indicates the signal strength (SNR or Signal-to-Noise Ratio). The next number is the carrier phase measurement, and the last number is the Doppler frequency observation value. This sequence is repeated for different signals for that satellite.

```
> 2023 09 14 00 00  0.0000000  0 57
C01

    38628128.268 6 201146781.54706        38.781    38628125.668 6 155539426.45306
    37.844    38628128.933 6 163448223.69706        41.312
C02

    38563027.413 6 200807808.22806        37.406    38563021.513 6 155277292.49706
    39.844    38563025.981 7 163172779.96407        42.125
C03

    37858669.993 6 197140000.21506        41.125    37858666.436 6 152441121.94006
    40.406    37858668.623 7 160192377.66307        43.969
C04

    39689112.935 5 206671576.33505        33.438    39689110.786 5 159811543.85205
    35.594    39689112.356 6 167937568.77506        39.281
C05

    39846939.300 5 207493521.68905        34.531    39846935.340 5 160447159.89805
    35.781    39846937.791 6 168605489.18206        39.562
C07

    37727307.332 6 196456077.76806        41.094    37727302.008 6 151912411.14906
    41.875    37727301.662 7 159636752.02007        45.594
C08

    35885938.934 7 186867543.00407        47.750    35885931.774 7 144497817.49007
    47.188    35885932.484 8 151845161.55808        51.156
```

Listing 2.2: Fragment of RINEX 3.04 data

2.6 Bad quality RINEX files

Several factors can contribute to the low quality of a RINEX file. These issues may stem from station-related problems, such as a malfunctioning antenna or obstructions around the station like construction. Additionally, network issues during data transmission, errors during data conversion, and even complex physical phenomena like solar activity can affect the quality of the data.

On the RINEX file itself, the problems can be seen in the following aspects [15]:

- Data gaps;
- Low observation ratio, by comparing possible observations with real observations;
- High number of cycle slips;
- Signal-to-noise ratio lower than expected;
- High multipath.

These aspects affect primarily the quality of the data present in the RINEX file, but the file can have problems that are not related to the data. These problems are generally related to metadata inconsistencies or formatting errors.

- Header inconsistencies, when the file metadata do not correspond to the station metadata;
- File name don't follow RINEX version naming rules.

2.7 BRDC files

GNSS stations provide broadcast navigation data in addition to observation data. Daily broadcast ephemeris files (BRDC files) [16] are created from site-specific files transmitted by GNSS stations. These files contain the unique satellite ephemeris messages (positioning, velocity, and clock information) for each day for all GNSS satellites. A BRDC file is created at the start of the day and updated on an hourly basis from the hourly broadcast navigation files.
These files, which contain a record of the satellite's positioning history, allow, for example, to ascertain the number of satellites a station can observe at any given moment of the day. By comparing the expected number of satellites a station should observe with the actual count in a RINEX file, it becomes possible to derive a reliable metric for assessing the quality of these files. Another example is that by knowing the precise orbits of the satellites, along with clock corrections, the receivers can triangulate signals from multiple satellites calculating an estimated position.

2.8 Conclusion

In this chapter, some important GNSS concepts were introduced. They provided an idea of how a RINEX file is created, starting with the satellites or constellations (groups of satellites) that send signals to ground-level receivers. These receivers process the signals to store various GNSS data. This data is sent to the station owner or administrator via the internet, who generally proceeds to convert RAW files to RINEX files using the appropriate software. The BRDC files are created through a similar process, but contain a compilation of satellite navigation data from the station.

Chapter 3

GNSS Quality and Monitoring: State of Art and Related Work

3.1 Introduction

This chapter provides a study of the current state of quality control for RINEX files and GNSS stations monitoring. As the objective of this internship is not to develop software to perform the QC of RINEX files, but the overall process to monitor the quality of the files and stations, this study will allow us to justify the choice of the RINEX QC software package to use on the desired final solution. This study will also inspire the methods to be used for the final implementation.

3.2 Quality Control

Quality control is the process in which a company tries to assure that the quality of the product it provides is kept or improved. It involves testing units to determine if the product meets the final requirements. Various techniques are used for quality control, depending on the nature of the product and the specific industry in which the company operates.
The necessity of quality control starts from a company's commitment to their clients only to provide quality products. Furthermore, creating a product has its costs and it can be a waste of time and resources to produce products without a control plan in mind.[17]

3.3 RINEX QC Software packages

RINEX QC tools are essential for ensuring the integrity and accuracy of GNSS data before further processing. With the growing demand for high-quality geospatial information, various software solutions have emerged, each offering unique features and capabilities. In this chapter, we will explore several leading RINEX QC software packages

3.3.1 PoiNT-QC

PoiNT-QC [18] (Positioning and Navigation Toolkit – Quality Control) is a Windows-based software that estimates the quality control parameters of GNSS observations recorded in RINEX files. This software is designed to detect and remove outliers, multipath, cycle slips, and ionospheric errors in GNSS observation data using L1 and L2 signals. The GUI also provides visual representations of the quality control parameters as well as a detailed summary report.

3.3.2 GPSQC

GPSQC [19] is a Windows-based program that estimates the quality of GPS data in a RINEX file. The software is designed to control the multipath, cycle slip, and ionospheric errors of L1 and L2 signals in GPS observations.

3.3.3 TEQC

TEQC [20] was the most used software for RINEX operations, but it reached its end-of-life in 2019 with the owner not planning to release any more versions. Its support only extends to version 2 of RINEX files, making it unsuitable for current projects.

3.3.4 BKG Ntrip Client - BNC

The BNC [21] is a program for simultaneously retrieving, decoding, converting, and processing real-time GNSS data streams. It was developed by BKG [22] which is a GNSS Data Center that provides high-accuracy observational and navigational GNSS data and products.
The software was developed for Linux, Windows, and Mac OS systems as a GUI application but with support for CLI.
As for checking the quality of RINEX files, the software do multipath analysis sky plots, Signal-to-noise ratio sky plots, Satellite availability plots, and Satellite elevation plots while supporting the RINEX 3 format, being a suitable tool for RINEX QC.

3.3.5 G-Nut/Anubis

G-Nut/Anubis [6] is a command-line software package used to verify the quality of all GNSS constellations available, giving qualitative and quantitative control of RINEX files.
The software has three versions, Anubis Free, Anubis Pro, and Anubis Real-Time. On all its versions the software provides support for versions 2 and 3 of RINEX, performing header checking, observation statistics, data availability, and other important metrics.
Anubis has a well-documented manual that describes in detail all the software versions, all their capabilities, and operating modes giving concrete examples of how to use them. Another important description made in the manual is the output description. The Anubis output can be in three different formats which are JSON, XML, and XTR. The XTR format is the most detailed one, as the other formats lack some information.

3.4 Similar Services

3.4.1 EPOS

The European Plate Observing System (EPOS) is a multidisciplinary, distributed research infrastructure that integrates information from the solid Earth science community in Europe [23]. EPOS operates GNSS stations and provides the monitoring of the RINEX data availability and visualization of daily RINEX data quality of EPOS GNSS stations using a system called RunQC.

3.4.1.1 RunQC

RunQC [24] is a software used by EPOS and is very similar to the one we intend to implement at MIRASpaco. It makes use of the G-Nut/Anubis software to analyze the quality of RINEX 2 and 3 files and can be used for three main purposes:

1. Generating QC-XML files that contain a summary of the most important QC parameters stemming from a single RINEX file;
2. Populate the EPOS DB with the QC-XML files;
3. Providing both actions in a single sequence.

3.4.2 EUREF

The EUREF [25] Permanent GNSS Network consists of a network of continuously operating GNSS reference stations, data centers providing access to the station data, analysis centres that analyze the GNSS data, product centers or coordinators that generate the EPN products, and a Central Bureau that is responsible for the daily monitoring and management of the network.
ROB [25] is the Royal Observatory of Belgium and is the entity that manages the EUREF Central Bureau. They provide a daily and monthly tracking of station performance with data extracted from Anubis and represented with plots that allow to see what happened on that timeline for specific metrics.

3.4.3 UNAVCO/Earthscope

The EarthScope consortium [26] handles GPS/GNSS data and products from globally distributed permanent stations and globally distributed campaign sites. Their process is based on receiving RAW data from the stations, translating it to RINEX format, and checking the quality of the files using TEQC. With the resulting data, they make quality reports to the users.

3.5 Conclusions

In conclusion, this chapter highlights the critical role of quality control in product development. The inspection and verification processes provided by QC are essential for ensuring the accuracy and reliability of RINEX files. The wide range of software packages available for RINEX quality control underscores the diversity of tools and methods used to maintain data integrity. Additionally, the fact that various organizations distributing GNSS data implement their own quality control measures further reinforces the importance of quality assurance in GNSS data management.

Chapter 4

Software, technologies and engineering

4.1 Work plan

This section is dedicated to the planning of this internship

4.1.1 Company integration

The first task of this internship was integrating the company. Understanding the workflow, the products provided, and the clients was crucial for preparing for the proposed challenges.

4.1.2 Studying of general GNSS and geodesy concepts

The main challenge of this internship was to develop procedures for controlling the quality of RINEX files. This required some understanding of GNSS concepts. To achieve this, the GNSS data creation process was studied, along with the structure of RINEX files and the factors that contribute to poor-quality files.

4.1.3 Familiarization with G-Nut/Anubis

G-Nut/Anubis was the software chosen by MIRASpaco for the quality control of RINEX files. This software takes in a RINEX file along with a BRDC file from the same date and outputs a report file with several metrics useful for quality control. The familiarization with this software was made via the official user manual and by testing the software with experimental scripts.

4.1.4 Database analysis and requirements assessment

The data produced by Anubis will be stored on the MIRAnet database. Therefore it was necessary to analyze the current database and adapt it to the new needs. This involved creating new tables for the quality control data as well as a new settings table for file approval.

4.1.5 Script development for GNSS QC data processing

This task consisted on creating a script that runs the RINEX files from each stations through Anubis, analyzing the Anubis output and inserting the data in the database.

4.1.6 Script development for data visualization and plot generation

After inserting the quality control data into the database, the next step was creating a script that uses that data to generate quality control plots that show the behavior of parameters of each station over time.

4.1.7 Update of Miranet for Plot Display and Settings Management

This task consisted of adding functionality to the MIRAnet UI to display the new quality control functionalities. New pages were created and existing ones were updated.

4.1.8 Writing of the report

The writing of the report was carried out during the internship and describes every step to conclude the tasks proposed.

4.2 Technologies used

This section is dedicated to describing the technologies used on the development of this project.

4.2.1 HTML

Hyper Text Markup Language [27] is the standard markup language for creating web pages, it describes the structure of a web page using several elements that tell the browser how to display the page contents.
In the context of this project, HTML was used to improve the MIRAnet web application by displaying the newly implemented quality control functionalities.

4.2.2 CSS

Cascading Style Sheets [28] is the standard language to style a web page. It describes how HTML elements should be displayed on a web page and allow for a deep personal customization of a website. The style definitions are normally saved in external .css files allowing for design changes of an entire website by modifying only one file.

4.2.3 JavaScript

JavaScript [29] is an interpreted high level programming language. It is the most used scripting language for web development and along with HTML and CSS, they form the three most used technologies of the web.

4.2.3.1 Highcharts

Highcharts is a JavaScript library that makes data visualization easy on the web.

4.2.4 Bootstrap

Bootstrap [30] is a framework for the development of the front-end of web applications that uses HTML, CSS, and Javascript. It simplifies the task of styling a website, by providing pre-styled elements with simple, but effective designs that are easy to integrate. A downside of Bootstrap is that it lacks the deep personal customization provided by raw CSS. On the other hand, it manages to create a easier to read code. Bootstrap is praised for its responsive layouts that allow for a page to adapt to the dimensions of the device it is being displayed on.

4.2.5 PHP

PHP [31] is a scripting language that is especially suited for web development. It is commonly used as a back-end development language, meaning that is executed on the server side.
In the context of this and many other applications, PHP works alongside JavaScript to handle requests between the client and the server. For example, if a client requests a change to the database, that request is sent via JavaScript to PHP, which then validates the request, communicates with the database, and finally returns a response to the front-end.

4.2.6 Python

Python [32] is a very popular multi purpose programming language. It is used for scripting, mathematics, web development, and software development. It is a high level language with an elegant syntax that allows for easy to read code. Python runs on an interpreter system, and works on most operating systems.
For this project, python was used for the development of the scripts that automate the RINEX files quality control procedure, meaning running files through Anubis and gathering the output data, and the script that generates the quality control plots.

4.2.6.1 Matplotlib

Matplotlib [33] is a comprehensive library for creating static, animated, and interactive visualizations in Python. This library was used to generate quality control plots that allowed the users to see a station's behaviour over time.

4.2.7 MySQL

MySQL [34] is a fast and robust SQL database management system. MySQL databases are relational, meaning that the data is stored on separate tables with well defined relations with each other that prevent inconsistencies and unorganized data.
SQL stands for "Structured Query Language", and is is the most common standardized language used to access databases.

4.2.8 DBSchema

DBSchema [35] is a software that helps design and manage databases by creating schema, generating SQL scripts or even connecting directly to a live database and pushing changes to a database structure in real time. It is also very useful to generate comprehensive documentation on databases with tools that generate whole HTML pages and PDF files.

4.2.9 RxTools

RxTools [36] is a set of GUI (Graphical User Interface) applications that work with all Septentrio GNSS receivers. These tools have various purposes like plotting data or converting raw Septentrio GNSS files to RINEX.
On the context of this work, RxTools was used to extract data not related to RINEX from the Septentrio raw GNSS files, which allows for monitoring of the receiver's health status.

4.3 Software engineering

This section outlines key concepts derived from this internship description and discussions with the team at MIRASpaco. The following requirements helped shape the proposal for the MIRAnet architecture and identified the necessary modifications for the database.

4.3.1 Quality Control System Requirements

1. All RINEX files must go through the quality control process;
2. After the quality control process, RINEX files can be approved, approved with warnings, sent to quarantine, or excluded;
3. The MIRAnet administrator must be able to choose what errors make a file be sent to quarantine or excluded, and what errors aren't important;
4. There must be a log system to notify the MIRAnet administrator about the state of RINEX files;
5. There must be a platform for the MIRAnet administrator to revise RINEX files in quarantine;
6. MIRAnet administrator must be able to require the reprocessing of a RINEX file;
7. MIRAnet administrator must be able to mark a quarantined RINEX file as excluded;
8. The system must store quality control information from the G-nut/Aubis output file;
9. The system must generate plots with different information for station monitoring purposes;

4.3.2 System Architecture

In this section the new architecture model for the MIRAnet will be proposed, being based on the current MIRAnet architecture. The current architecture was explained by people at MIRASpaco and the sharing of the following diagrams was allowed by the company.

In the old flow of the MIRAnet (see Figure 4.1), the CORS were sending the raw GNSS files to a gateway file server which then proceeded to forward those files to the MIRAnet repositories. These repositories called a service, also built by MIRASpaco, that converted raw GNSS files into RINEX files, returning them to the user that made the request (repository in this case). The RINEX files were then inserted into the database and were made available to the MIRAnet users.

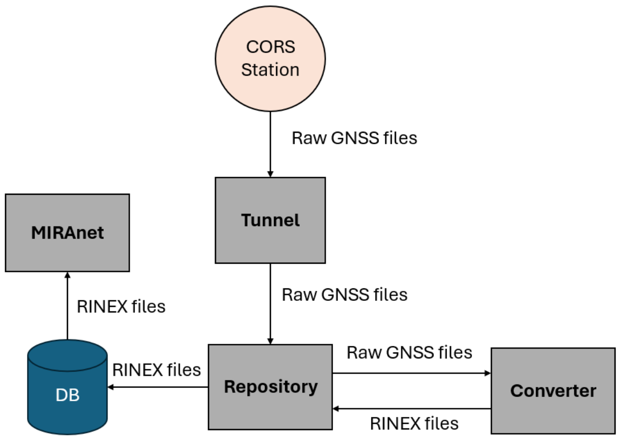


Figure 4.1: Old MIRAnet architecture.

In the new flow (see Figure 4.2), GNSS raw files are sent from the stations to a server that stores the files on a bucket of raw files. These files are sent to an internal service that converts raw GNSS files into RINEX files. After converted, both files are stored in raw and RINEX buckets waiting for the QC procedure. From this point, the files are ran through the QC procedure and are inserted in the database with a status value that indicates their QC state. This state will say if a file hasn't gone through QC (in case of an error), if it passed, or if it failed. It will also determine which action to take with the file (see Table 4.1). After the QC process, only approved files are made available to the user and important data will be stored in the database for the construction of plots for the monitoring of the stations. Files that don't pass the QC go to the error or quarantine repositories, depending on the severity of the problem. Files on the quarantine repository, can then be corrected and reprocessed by sending the corrected GNSS raw file into the unprocessed raw bucket.

state	description	action
0	Hasn't went through QC	Go through QC
2	Passed QC	Make available to the users
1	Passed QC with warnings	Make available to the users and notify the administrator
-1	Failed QC with warnings (Quarantine)	Automatic or manual reprocess of recoverable files
-2	Failed QC	Exclude file

Table 4.1: QC state

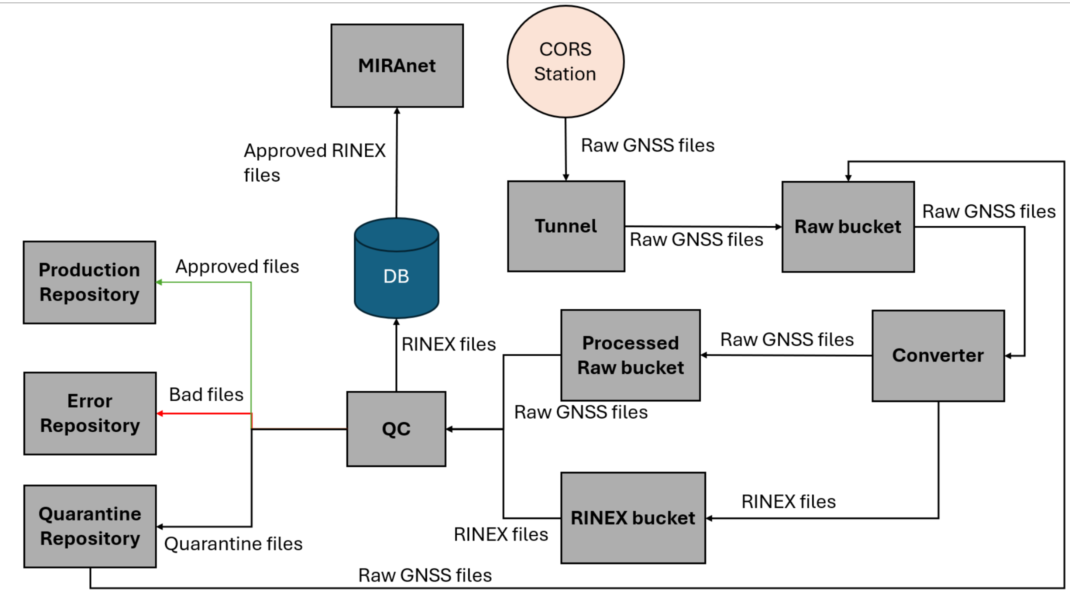


Figure 4.2: Proposed MIRAnet architecture.

4.3.3 Quality Control service Diagram

The diagram in Figure 4.3 illustrates the workflow of the RINEX file processing within the quality control service. Initially, the file is run through the Anubis software, which generates an output that is analyzed to determine the file's status and extract the necessary QC data parameters. These parameters are used to monitor the file and the associated station, with the data being stored in the database. Subsequently, the QC data are processed by software that generates plots, enabling the visualization of the station's behavior over time.

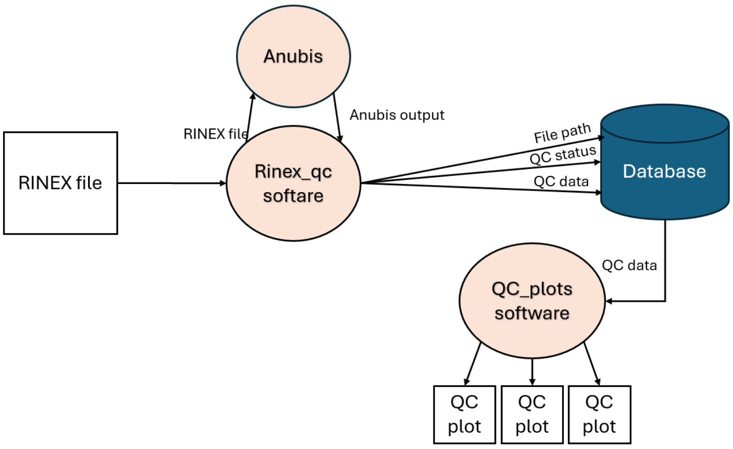


Figure 4.3: QC service Diagram.

4.3.4 Database

After discussing the old MIRAnet database model (see Figure 4.4) with the people at MIRASpaco the conclusion was that the database was not yet ready for the QC features. As so, in this section, the changes that were proposed for the QC implementation are discussed.

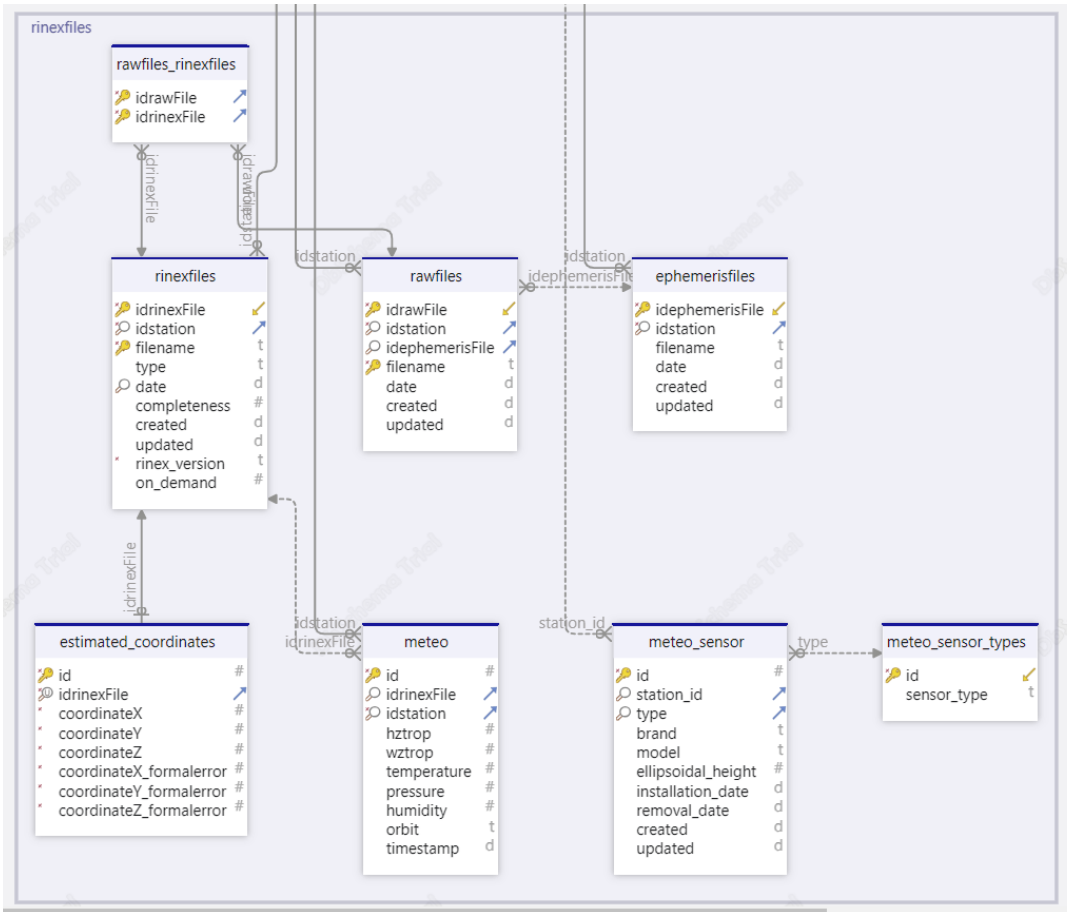


Figure 4.4: Old MIRAnet database (RINEX section).

On the new version of the database (see Figure 4.5) a rinex_qc section was added with seven new tables:

1. **constellations:** This is a pre-loaded table with the different possible constellation id's that can be found in a RINEX file;
2. **signals:** This table stores the different signals id's from each different constellation;
3. **constellations_signal:** This table creates a relation between the different constellations and signals;
4. **summary:** This table stores the main general statistics created during the QC procedure, which is used both for the definition of the file status and the generation of plots;
5. **const_sign_summary:** This table stores general statistics related to specific constellations, this data is used both for the file status definition and the creation of plots;
6. **constellation_summary:** This table stores statistics specific to each signal of each constellation, this data is used for the generation of plots;
7. **qc_logs:** This table registered all the errors and warnings, discovered during the QC procedure, in a well-descriptive manner for later display on the MIRAnet web application.

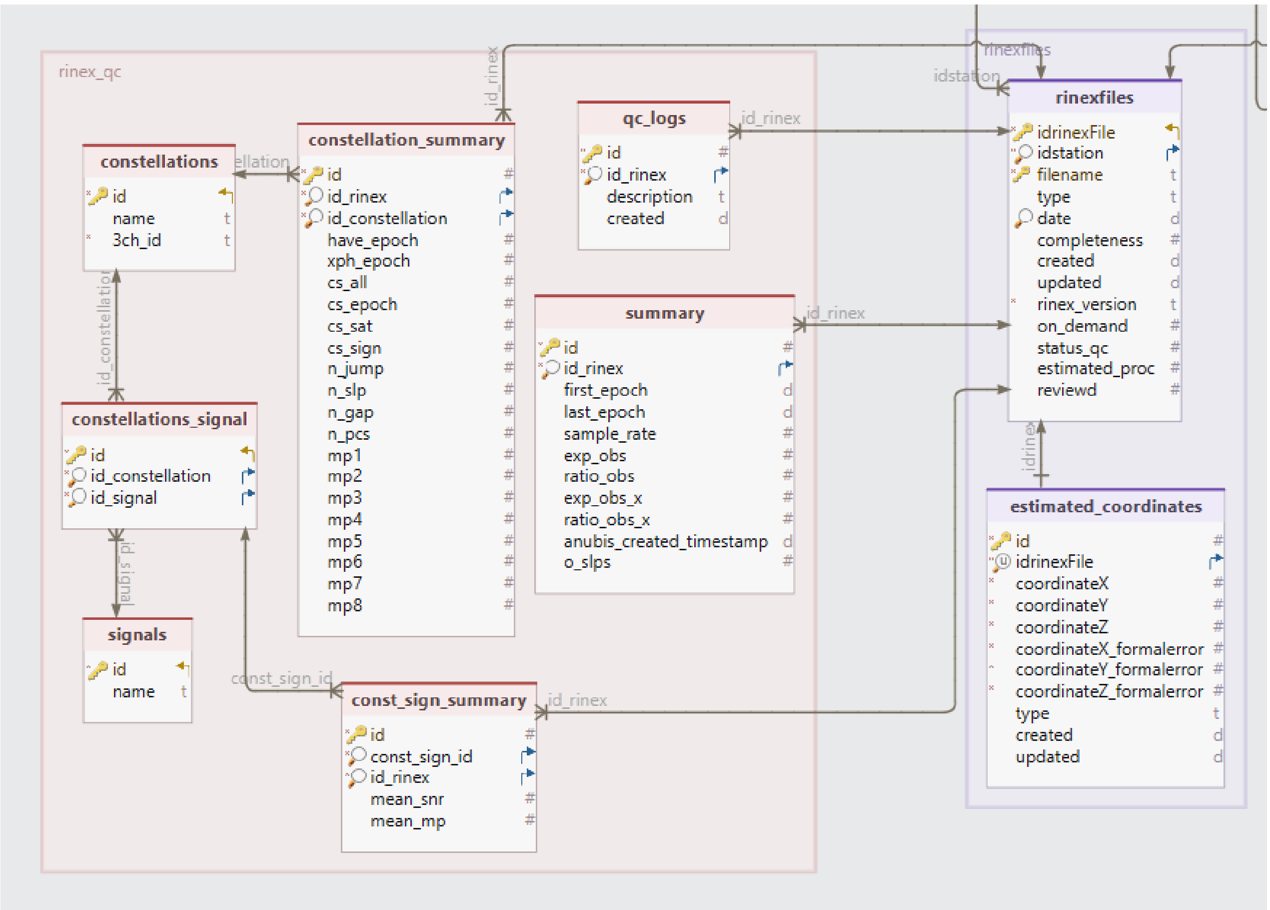


Figure 4.5: New MIRAnet database (RINEX and QC section).

It was also necessary to create tables to define settings (see Figures 4.6 and 4.7) for the QC procedure. For that two new tables were created:

1. **qc_settings:** This table is pre-loaded with the various settings used in the QC procedure, populated with the values recommended by the MIRASpaco team. It will automatically complete the QC-related section of the form when inserting a new station into the MIRAnet;

2. **station_qc_settings:** This table is designed to make the qc_settings customizable for each station in the network. This feature can be particularly useful if the MIRAnet administrator observes that a specific station's behavior significantly differs from others. For example, if the general observation ratio approval is set at seventy percent and a particular station consistently falls below this threshold, the administrator may choose to lower the threshold for that station to ensure its data are approved.

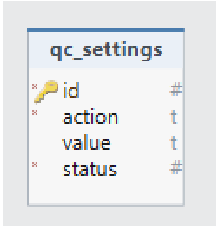


Figure 4.6: New MIRAnet database (qc_settings).

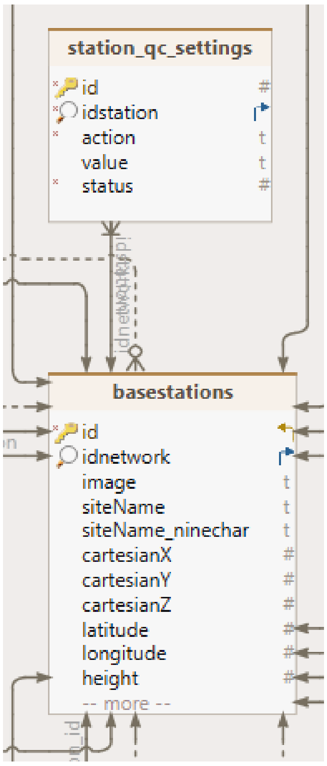


Figure 4.7: New MIRAnet database (basestations and station_qc_settings).

After talking with a possible client that is interested in the new QC features of the MIRAnet, a personalized request was made by the client which was accepted by the MIRASpaco team. The entire client network is composed of Septentrio receivers, and the request was that there was a deeper monitoring on the station's health by adding more information to the raw GNSS file like temperature, voltage, CPU load, and radio interference. This data can be later extracted from the raw file using Septentrio specific software.
To store this information four new tables were created (see Figure 4.8):

1. **sbf_metrics:** This table gathers the different types of data;
2. **sbf_cpu_gain:** This table stores the information of the gain and CPU load;
3. **rfi_data:** This table stores the information of the frequency and with of possible radio interferences;
4. **sbf_tmp_vlt:** This table stores the information on the temperature and voltage.

The three separate tables for different data types were necessary to accommodate varying data generation rates. For example, it might be reasonable to store temperature values every 10 minutes, while RFI data needs to be recorded every 10 seconds.

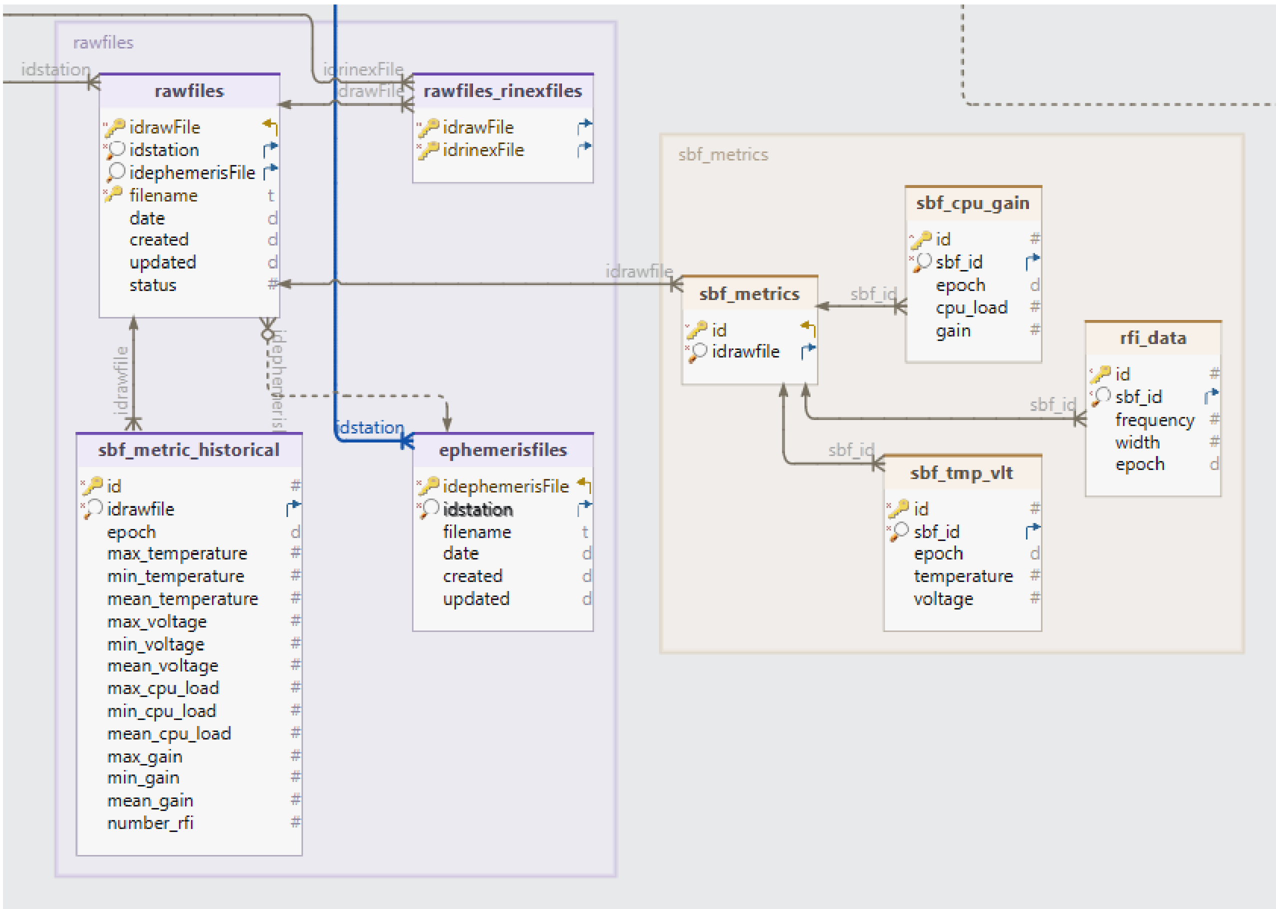


Figure 4.8: New MIRAnet database (Raw and sbf_metrics section).

4.4 Conclusion

This chapter provided an overview of the work plan, the technologies employed, and the software engineering methodologies guiding the project. The work plan outlined a structured approach to ensure efficient progress and timely completion of tasks. The selection of technologies was driven by the specific requirements of the project, focusing on scalability, performance, and ease of integration. Finally, the application of sound software engineering principles ensures that development follows best practices, leading to a robust and maintainable system.

Chapter 5

Development

5.1 Introduction

This chapter is dedicated to the description of the methods used for the development of the different tasks realized during the internship.

5.2 Database

The first task in the development of the entire QC procedure was to prepare the database for the new needs. For that, the MIRASpaco/MIRAnet team gathered together to look at the old database and decide what changes were to be made.

This meeting was a crucial moment for the planning of the development phase as it also served as a brainstorm to decide what features were to be implemented and what problems might surge along the way.

Meetings like this were held regularly during the entire internship duration to keep track of the work done, solve problems, and new ideas. The meetings results are presented in the chapter 4.3.4.

5.3 RINEX Quality Control - Rinex_qc

Rinex_qc is the internal name given to the script responsible for processing RINEX files, gathering QC data and decide a QC status for the file. This section is used to describe the script and how it works.

Firstly the raw GNSS files are sent by the receivers and stored in a specific directory. Using a script, created by the MIRASpaco team, that runs on the crontab, these files are sent to a program, that converts raw GNSS files into RINEX files. If the conversion is successful, the files are stored in a new directory dedicated to RINEX files, organized by station. This directory serves as the starting point for the Rinex_qc script (see Figure 5.1). If there are issues with the conversion, the process is repeated a predetermined number of times or until it succeeds. If the conversion ultimately fails, the raw files are moved to an error bucket.

Starting from this directory called RINEX bucket, which is a directory that holds and organizes unprocessed RINEX files, the script loops through each sub-directory, meaning that it processes the files of each station at a time.

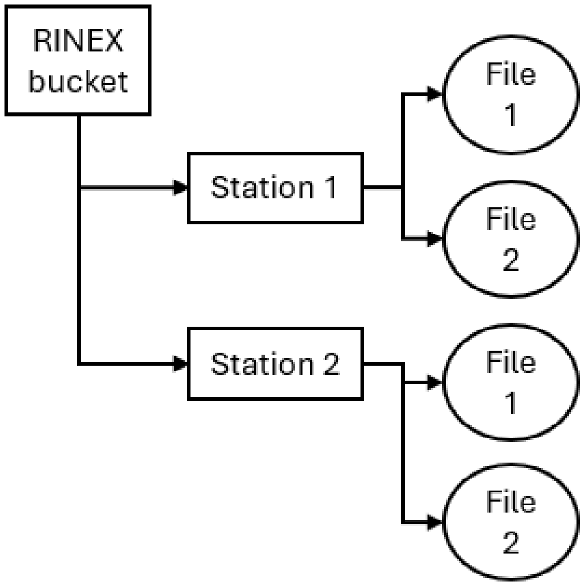


Figure 5.1: RINEX bucket structure.

From the name of each file, the RINEX version, year, DOY (day of year), and station name are extracted. Next, the file year and DOY are used to download the BRDC file corresponding to that day from a public repository. The BRDC file is downloaded only once per day and reused for all RINEX files from the same day. This BRDC file, along with the RINEX file itself is then served as input to the Anubis software that processes it and generates an output file. This output file is a '.xtr' extension and is an ASCII file that contains a report of quality control metrics about the RINEX file (see Figure 5.2).

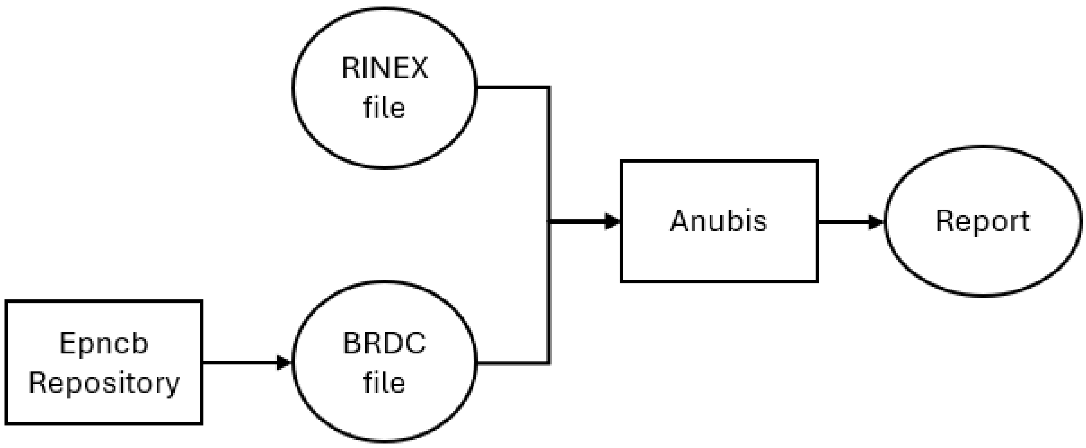


Figure 5.2: Rinex_qc - step 1

Having the .xtr file, it is necessary to analyze it, retrieve the QC data, and define a QC status for the RINEX file. For that, a variable called qc_status is initialized with the value 2. This means that we start by assuming that the file will pass the QC tests. From there it is only possible for a file to go down on this score.
The QC analysis is divided into two parts. First, the header is analyzed. This is the first part of

any RINEX file and has information about the file, the receiver, and other general data. Here we look if information like marker name, domes number, and agency are correct (see Table 5.1). If an issue is detected, the corresponding qc_status value for that specific information is retrieved from the table station_qc_settings. If this retrieved value is lower than the current qc_status, the qc_status is updated to the new value. Additionally, a description of the error is added to the qc_logs table.

Parameter	**Status**	**Value**
Marker Name	Quarantine (-1)	
Domes Number	Approved with warning (1)	
Agency	Approved with warning (1)	
Receiver type	Quarantine (-1)	
Receiver serial number	Quarantine (-1)	
Antenna type	Quarantine (-1)	
Antenna serial number	Approved with warning (1)	
Antenna Delta	Quarantine (-1)	
Approximate Position	Approved with warning (1)	5 meters
Maximum approximate Position	Quarantine (-1)	10 meters

Table 5.1: Analyzed information on header section with example status and value

The second part of the QC analysis is the data part (see Table 5.2). The current value of the qc_status is kept, and the RINEX data is compared to the station_qc_settings thresholds. The process is similar to the first part and repeats the idea of getting the value being compared, going to the database, and getting the threshold value for that metric. If the value is below the threshold and the metric qc_status is inferior to the current qc_status it becomes the new qc_status and an error is logged to the qc_logs table.

Parameter	**Status**	**Value**
Minimum observation ratio	Rejected (-2)	30%
Observation ratio	Approved with warning (1)	80%
Total number of cycle slips for each constellation	Approved with warning (1)	300
Maximum number of cycle slips for each constellation	Rejected (-2)	10000

Table 5.2: Analyzed information on data section with example status and value

Also, other metrics that don't decide the qc_status are stored for the generation of QC plots, such as:

- **First epoch**;
- **Observation ratio**;
- **Expected observations**;
- **Minimum observed elevation**;
- **Observation ratio above x degrees over the horizon**;
- **Expected observations above x degrees over the horizon**;
- **Number of carrier-phase observations per one cycle slip**;

- **Different types of cycle slips per constellation**;
- **Number of jumps, slips, gaps, and pieces per constellation**;
- **Average Signal to noise ratio per signal of each constellation**;
- **Average multipath per signal of each constellation**;

After the the RINEX file is added to the database, and depending on the defined qc_status it is moved to a specific directory.
Taking into consideration the values in Table 4.1 the files are moved to three possible directories. A repository bucket (see Figure 5.3), for approved files. A quarantine bucket for files in quarantine that the administrator can manually approve and an error bucket that contains files with qc_status value equal to -2.
Additionally, hourly files differ from daily files is the duration of the data they contain: hourly files cover one hour of data, while daily files cover a full day. This means that each station generates one daily file and twenty-four hourly files per day.
The error bucket contains all the files within the same level, meaning that there is not much structure to the directory. However, the quarantine and repository buckets were designed to be more organized, separating files by RINEX version, type (meaning daily or hourly), year, and DOY.

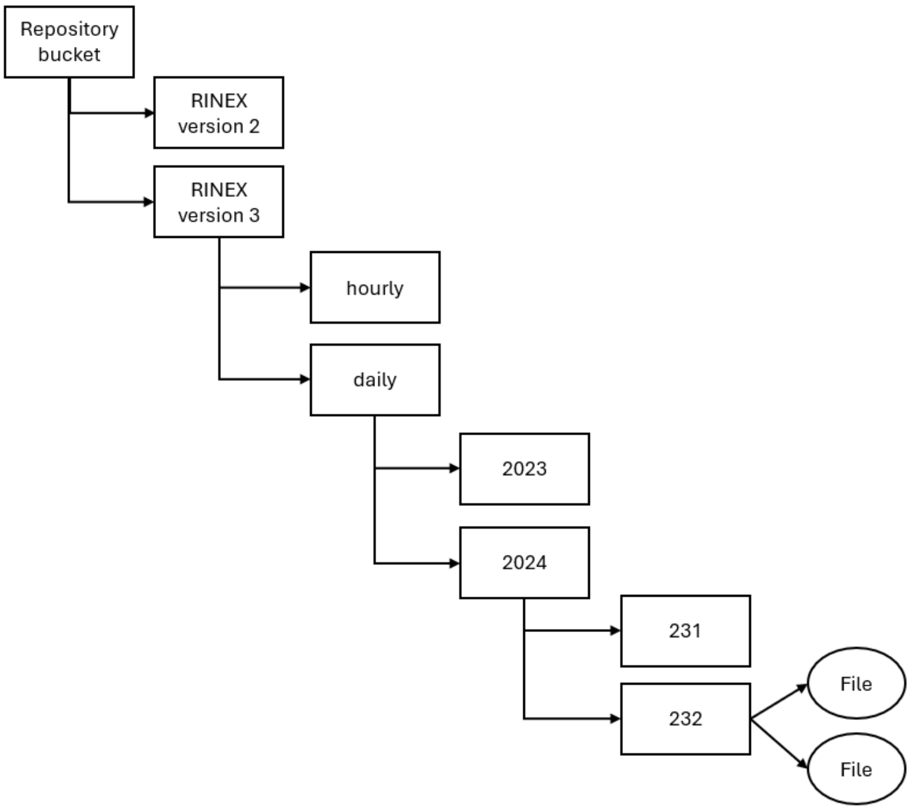


Figure 5.3: Repository bucket structure

At the end of the script, the idea is to have the RINEX bucket empty and the files and QCdata inserted in the database and distributed to the respective repositories.

5.4 Quality Control Plots - Qc_plots

The Qc_plots script was developed to generate quality control plots using the QC data previously stored in the database. These plots are displayed on the MIRAnet web application, providing administrators with essential visualizations of key metrics that help assess the health of receivers and quality of the acquired data.
All the plots are generated for the entire data span and also for a configurable data span for more recent data.

5.4.1 Observation ratio plot

The observation ratio plot 5.4 allows the visualization of the percentage of observations made by the receiver throughout its history. This percentage is obtained by making a ratio between the possible observations that could be made during an interval of time and the actual observations made during that time. Receivers rarely have a ratio of one hundred percent on observations, and this can be due to several factors (e.g., obstructions in a given direction), but generally, the ratio is consistent over time. This plot can help to identify outliers and sudden pattern changes that can initialize the process of identifying the problem.
For the observation ratio plot the script starts by looping through the different stations in the database that contain RINEX files and QC data, specifically observation ratio values. Then for each station, the observation ratio data is retrieved from the database along with the corresponding epochs. The observation ratio values are stored in an array y and the corresponding epoch values are on an array x, from here the python library matplotlib generates a simple plot with a few settings previously defined.

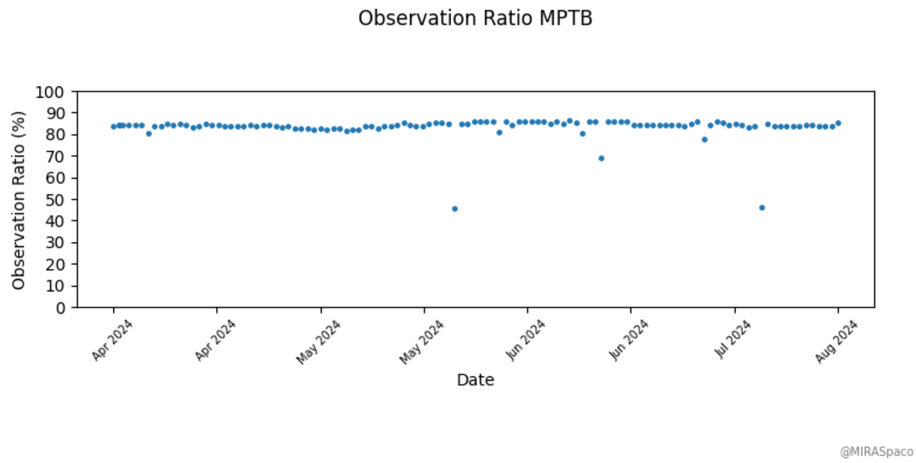


Figure 5.4: Observation ratio plot

5.4.2 Cycle slips and missing epochs plots

A cycle slip plot 5.5 5.6 allows users to visually identify and analyze discontinuities in the carrier-phase measurements of GNSS data. By plotting cycle slips over time for the different constellations, it is possible to identify outliers and patterns that might indicate that there is a problem that needs to be looked into, like signal obstruction due to environmental interference.

A missing epoch plot 5.7 5.8 can help identify data discontinuities on epochs where there should be an observation. Outliers and certain patterns might indicate problems like poor satellite visibility or receiver malfunctioning.

The script loops through each station and each constellation, gets the QC data from the database along with the corresponding epochs, stores them into arrays, and uses the matplotlib library to generate two types of plot, one where all the constellations are present, and the other type being individual plots for each constellation.

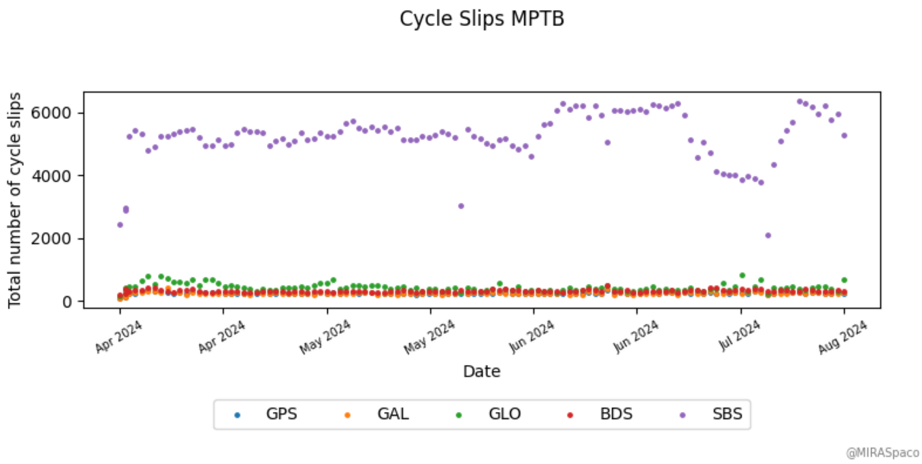


Figure 5.5: Number of cycle slips for all constellations

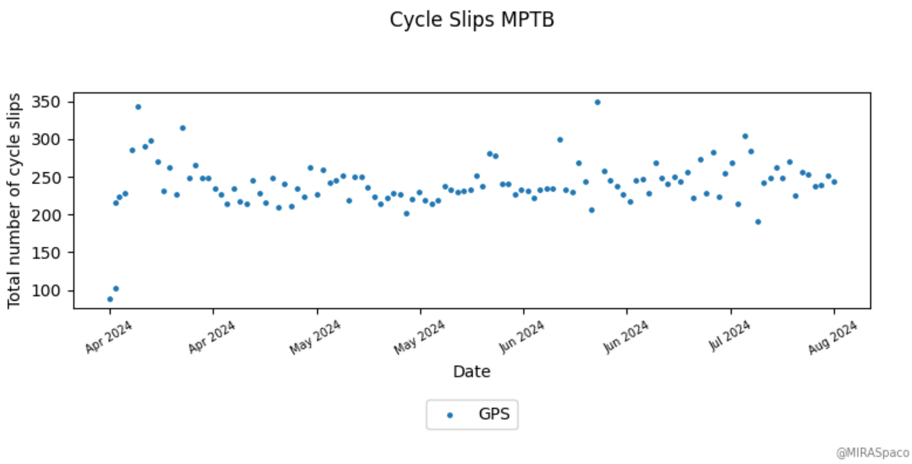


Figure 5.6: Number of cycle slips for GPS

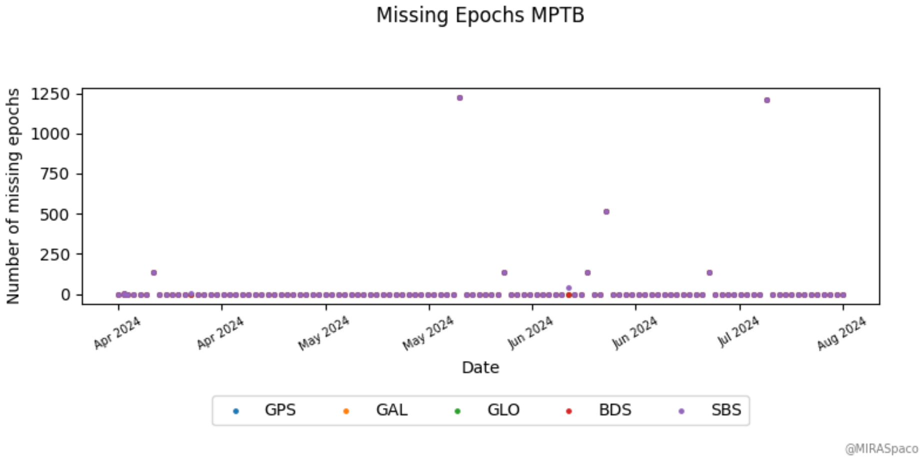


Figure 5.7: Number of missing epochs for all constellations

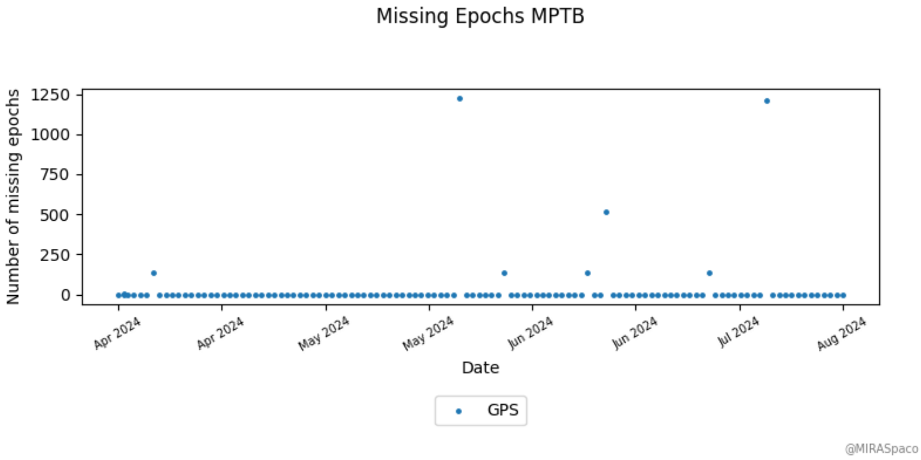


Figure 5.8: Number of missing epochs for GPS

5.4.3 Signal to noise ratio and multipath plots

Signal to noise ratio plots 5.9 5.10 allow the user to identify variations in signal strength and the presence of noise. Variations in these parameters might indicate problems with the the antenna, signal interference, or even adverse atmospheric conditions.

The multipath plots can be useful to identify the interference that the path that the signal takes before reaching the receiver has on the quality of the signal. Before reaching the receiver, the signal can be reflected on buildings and other surfaces increasing the distance that the signal traveled. This is called multipath, and its increase can jeopardize the quality of the position data generated.

For these plots, the script generates plots for each constellation. Each plot has QC data corresponding to the different signals of each constellation.

The script loops through each station and each constellation retrieving the QC data and corresponding epochs, and feeds it to matplotlib generating the desired plots.

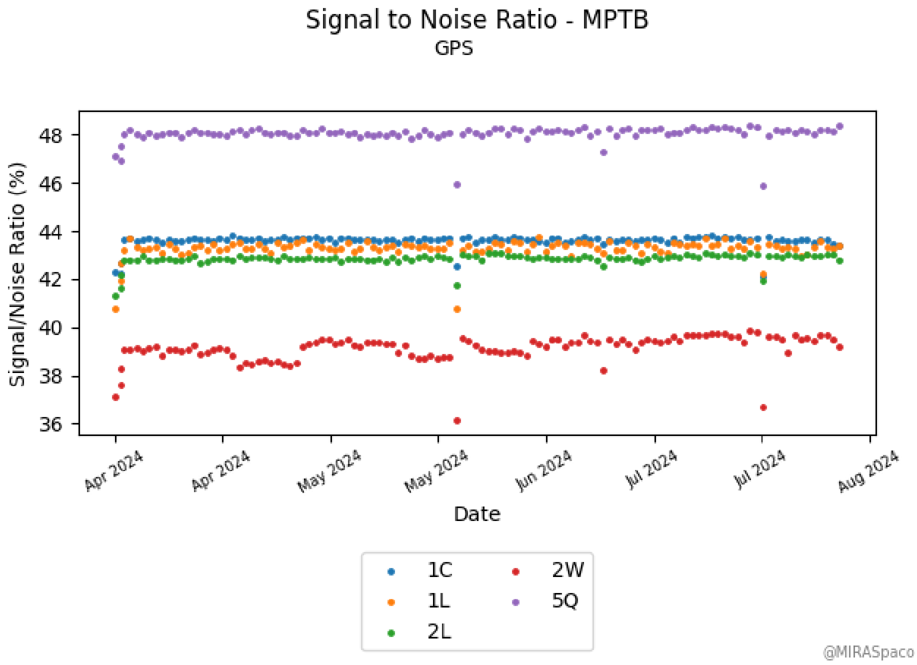


Figure 5.9: Signal to noise ratio for GPS

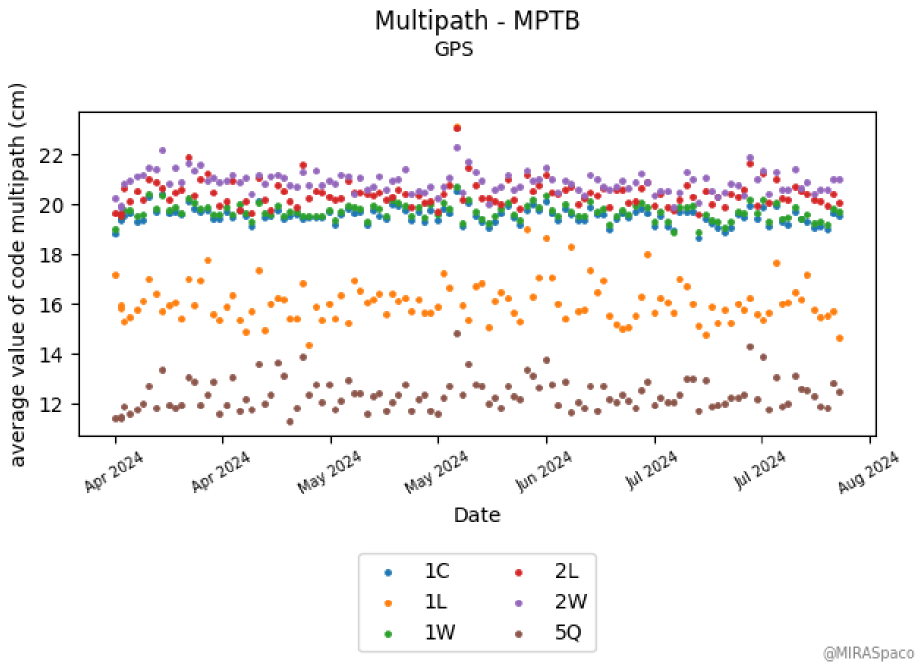


Figure 5.10: Multipath for GPS

5.5 MIRAnet Integration

Having the QC system working and the plots being generated the final step was to prepare the MIRAnet to interact with the system.

5.5.1 Quality Control (QC) settings

The first step was to be able to change the QC settings from the MIRAnet UI and add them to the form to add new stations. The MIRAnet already had a page to add new stations and another page to edit an existing station, so the definition of QC settings was added to these pages (see Figures 5.11 and 5.12). A new page was also created to edit the default QC settings for the new stations.

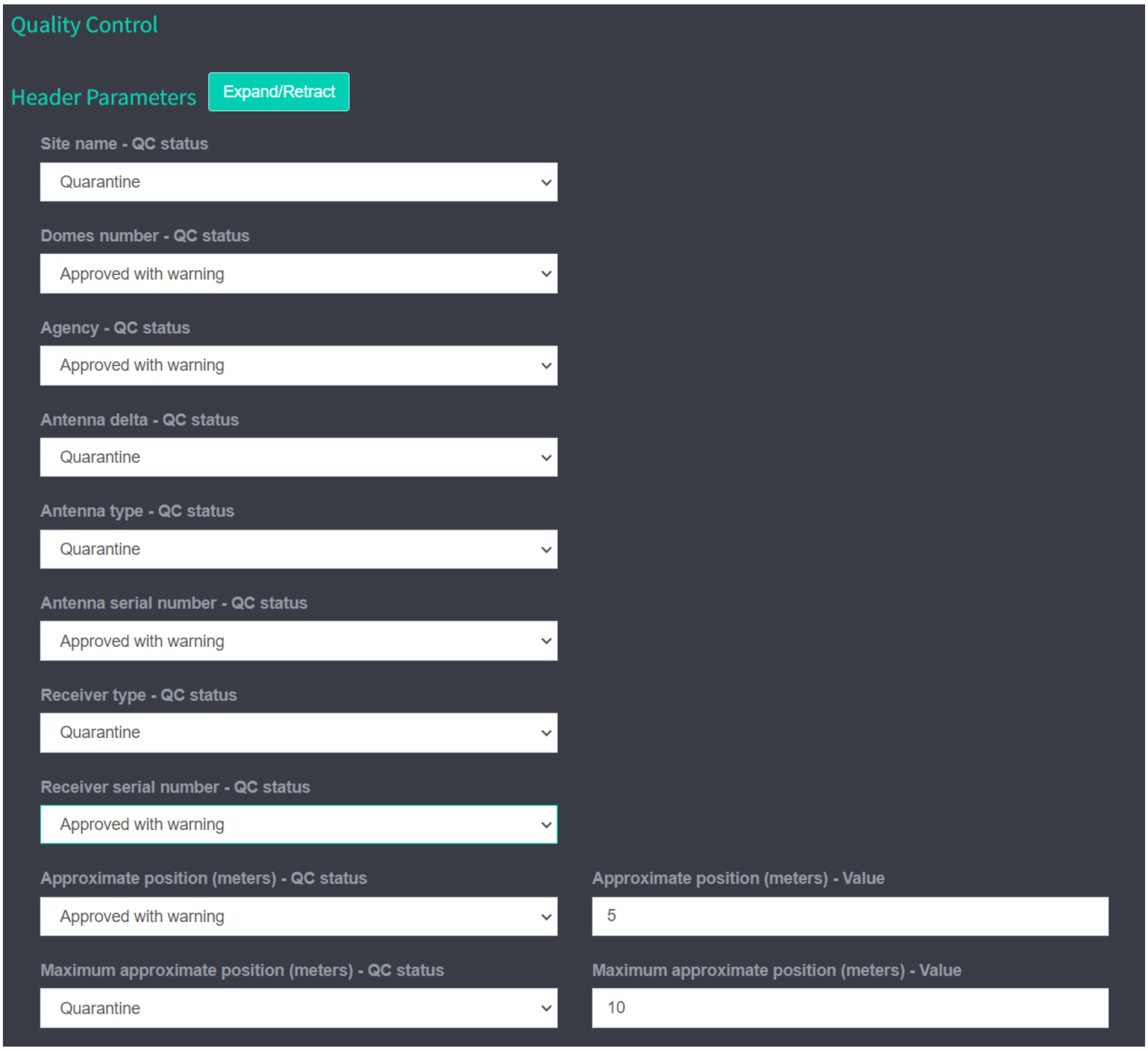


Figure 5.11: Station QC settings - header parameters

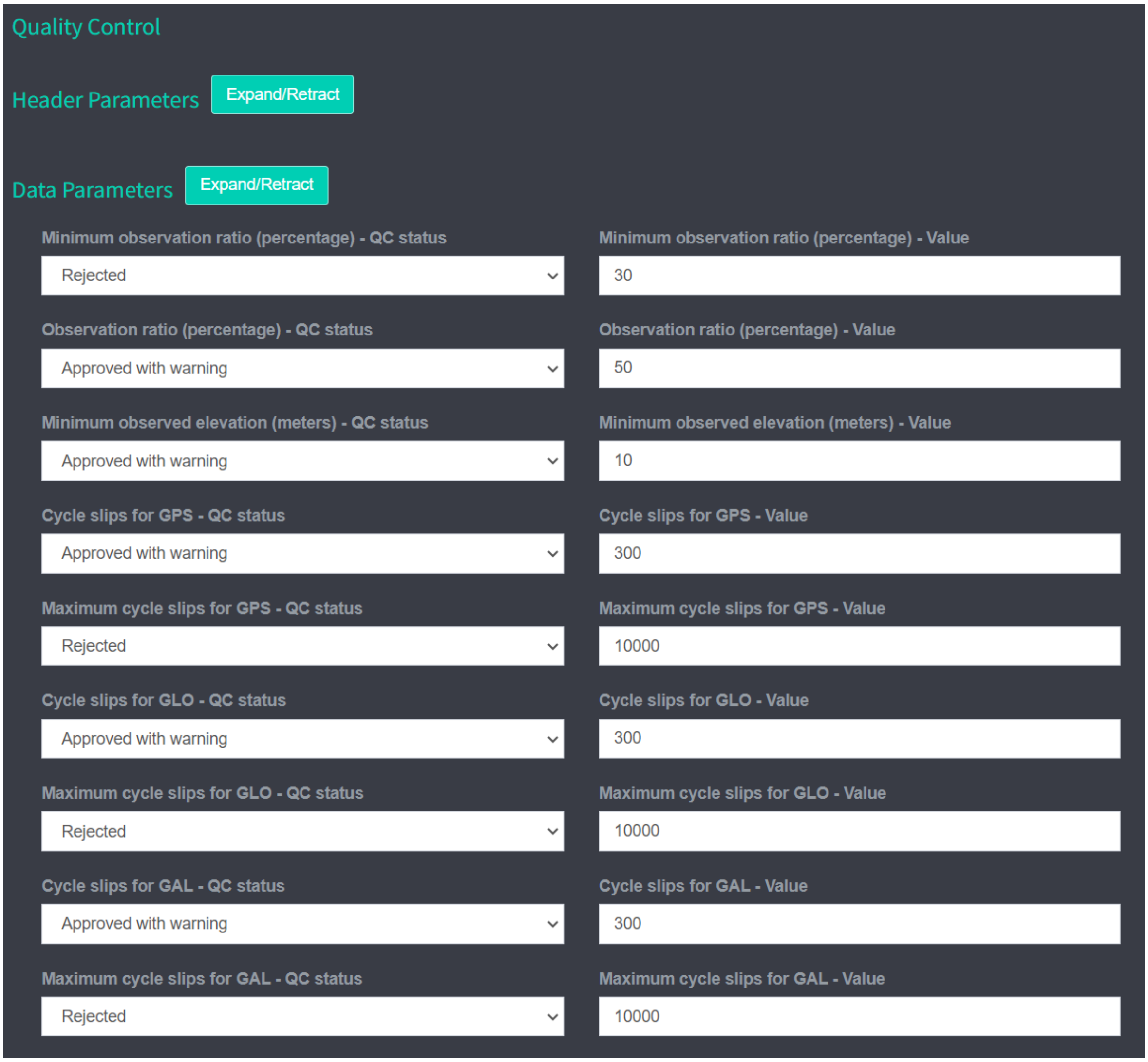


Figure 5.12: Excerpt of station QC settings - data parameters

5.5.2 Warning and error pages

Next, it was necessary to create a panel where the MIRAnet administrator could access the errors that emerged from the QC procedure. For this, two different pages were created. One page to display the files with warnings, the approved ones with a status equal to 1, and the ones that go to quarantine, with a status equal to -1. Another page was created for the error files that have a status equal to -2 and are not recoverable.

The page has a form that allows filtering the displayed contents. The desired stations and the interval of time of the displayed files can be selected. For the warnings page, it is also necessary to choose if the warning or quarantine files are displayed. (See Figures 5.13 5.14).

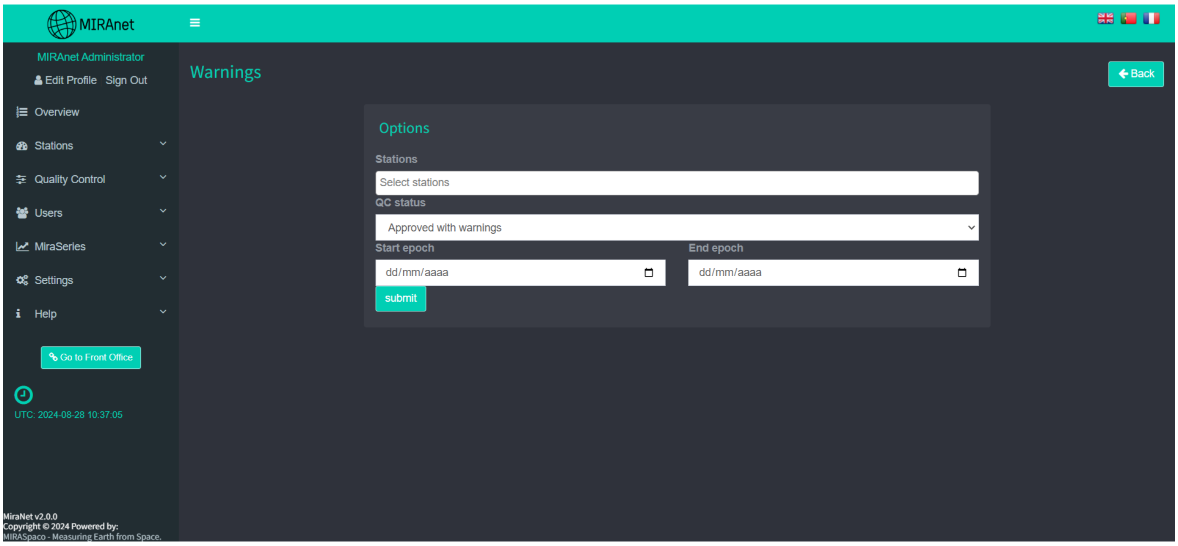


Figure 5.13: Warnings form

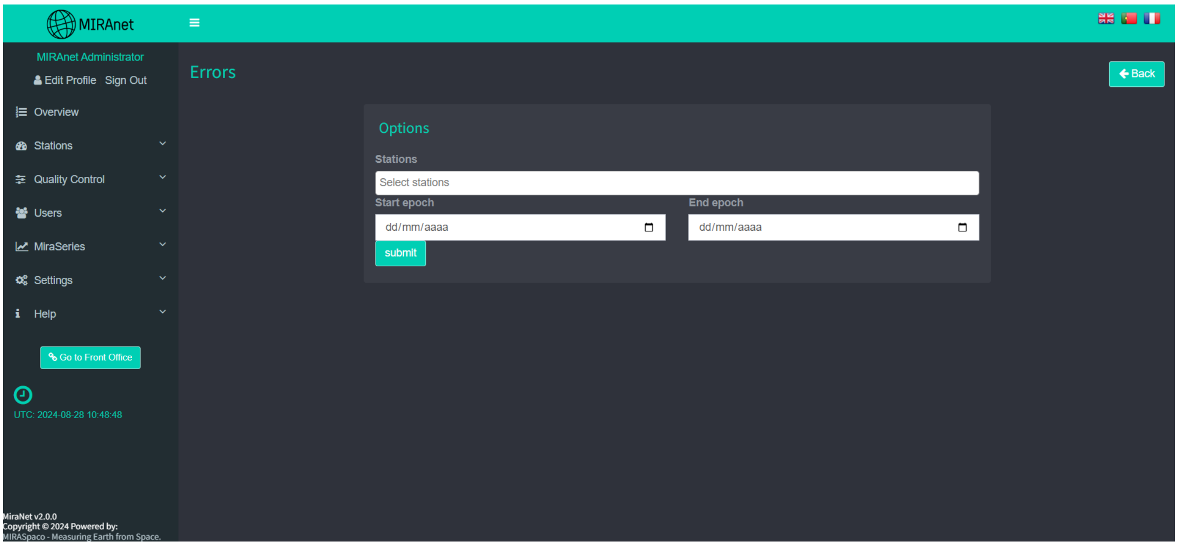


Figure 5.14: Error form

The files, error descriptions, and recommended actions are displayed in a table. Each station has its table containing its respective files. The warnings table allows the administrator to review errors either dismissing them, which removes the file from the table, or requesting reprocessing if the errors are solvable. Solvable errors typically involve metadata issues, while errors related to the data are generally unrecoverable.

The errors table is similar, but only allows the administrator to mark files as reviewed, removing them from the table. All files with status equal to -2 are assumed to be unrecoverable (See Figure 5.15).

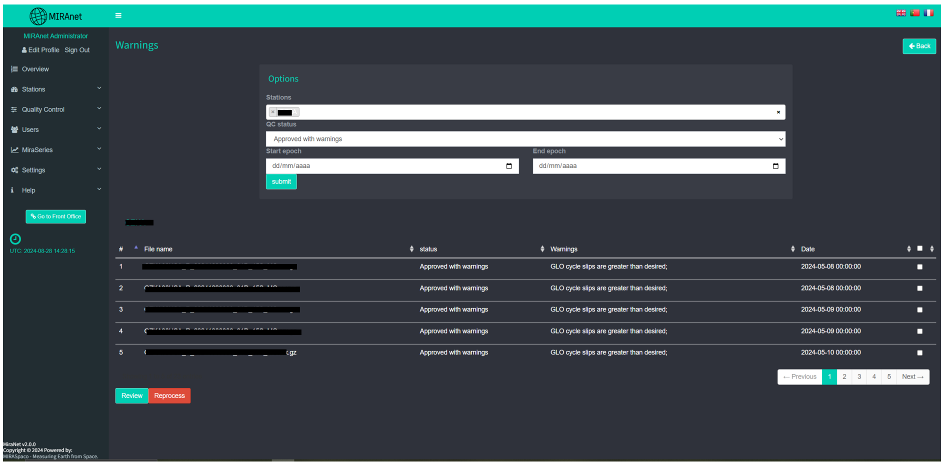


Figure 5.15: Warnings table

5.5.3 QC reports

A page was created to generate a customizable table to display all the stored QC data from each file.

The page has a similar design to the warnings and errors pages, starting with an options form that customizes the table, and a table for each station that helps with data visualization. (See Figure 5.16).

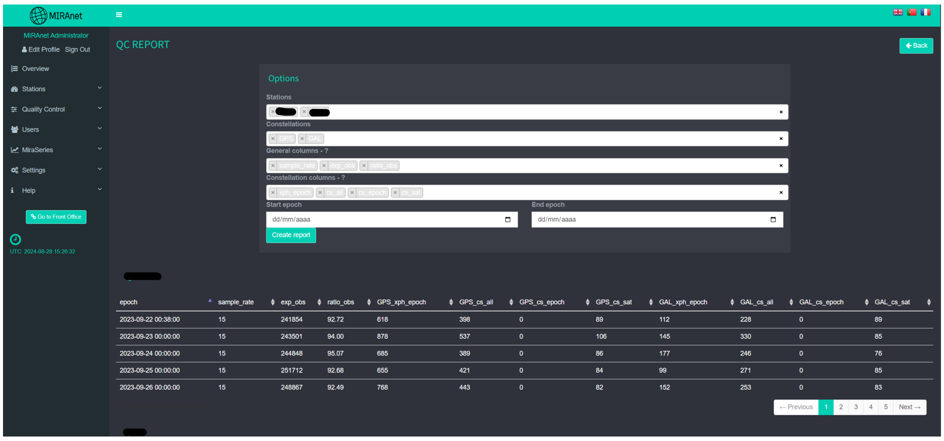


Figure 5.16: Qc reports page

5.5.4 Plots

The last step to integrate the new QC functionalities into MIRAnet was the creation of a page to display the QC plots (see Figures 5.17, 5.18, and 5.19). This page displays QC plots for individual stations, so it is accessed from the station list. The QC plot page has sections for each type of plot and the different plots from each type are in a carousel that allows the visualization of the different images.

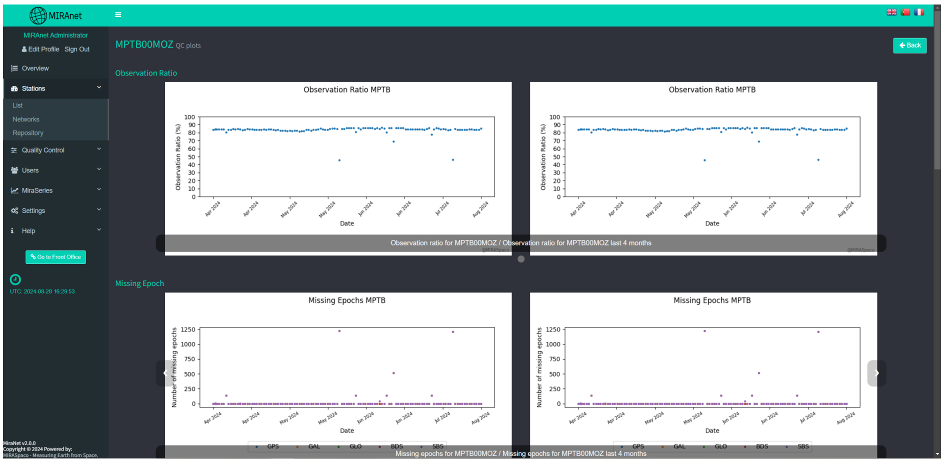


Figure 5.17: QC plots page - 1

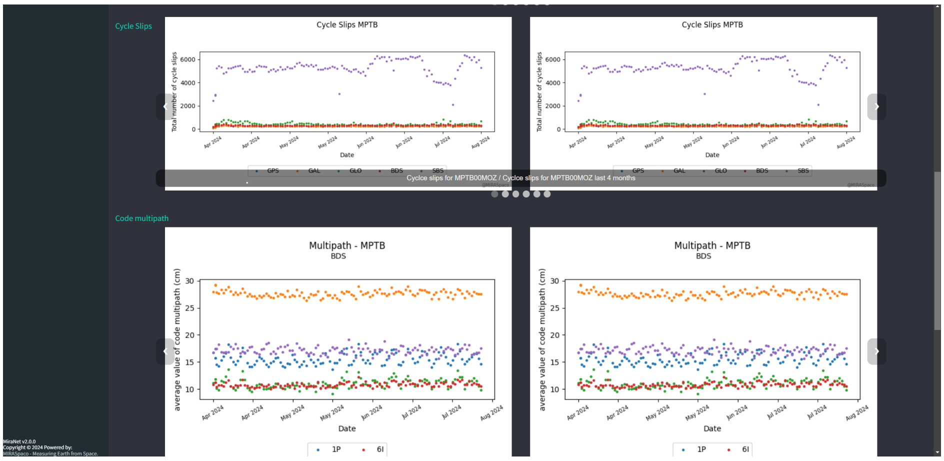


Figure 5.18: QC plots page - 2

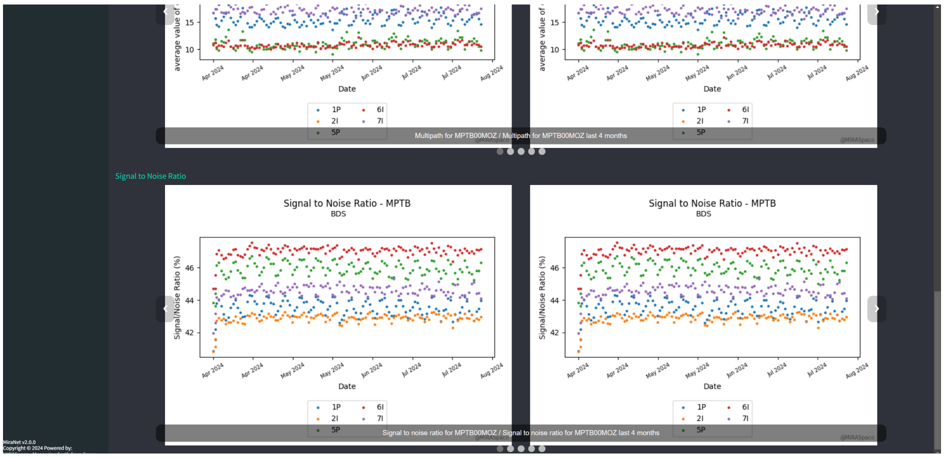


Figure 5.19: QC plots page - 3

5.6 Client Requirements

One of the factors that led MIRASpaco to include QC procedures on MIRAnet was a request made by a potential client. This client had a clear idea of their needs and offered to help describing the requirements and provide all the feedback necessary to develop these features.

At the end of the development phase, a meeting was scheduled with the client to review the newly implemented features. During this meeting, all the new functionalities were presented and discussed. The feedback received was positive. However, the client also expressed a desire for an additional set of functionalities.
The client's entire network is composed of Septentrio receivers, which can log important health and status data within their GNSS raw files (some other brands also have similar capabilities that can also be included in the future). The client requested the ability to log metrics such as voltage, temperature, and radio interference, among others, and to generate plots similar to the existing QC plots.
MIRASpaco, a small-sized company that prides itself on delivering reliable and personalized products, gladly accepted this request and promptly began the development process.

5.7 Septentrio specific functionalities

The first step was to prepare the database to receive the new parameters. This process was already described in chapter 4, section 4.3.4.
Next, the data was extracted from the raw GNSS files using Septentrio's proprietary software, RxTools 4.2.9. While this tool is freely available for download, it only comes with a GUI interface. However, to automate the data extraction process, command-line interaction is necessary. Additionally, since MIRASpaco primarily uses Ubuntu Server as their operating system, which lacks a GUI, the reliance on a graphical interface posed a significant challenge. To address this challenge, RxTools was installed on a local virtual machine running Ubuntu Desktop. This configuration enabled the examination of files extracted from a standard RxTools installation. During the process, it was determined that the scripts required for extracting the logged data from RxTools could be executed through the command line, eliminating the need for a graphical interface. The required files were then located and copied to a virtual machine with Ubuntu Server. This approach proved successful, enabling the extraction of logged data from the raw GNSS files using a script called bin2asc, which was extracted from the RxTools software.

5.7.1 Septentrio Receiver Metrics - Sbf_metrics

Sbf_metrics is the name of the script created to extract the provided Septentrio receiver metrics. The sbf metric files are divided by station in well defined directories. These directories are fed with new files every day, and the script is responsible for analysing the files and moving them out of the directories. For that, the script runs the sbf metric files through the bin2asc script. From here three different files are generated.

- A power status file, that contains information about the receiver voltage input throughout the time interval of the logged data;
- A receiver status file, that contains information about the temperature, CPU load and gain which is the total pre-amplification of the RF distribution network from the antenna to the receiver's input [37];

- An RFStatus file, which contains information about radio frequency interference's that were detected during observations.

These files are created from blocks that are stored inside of the sbf raw files, and the receiver has to be configured to log this data. This means, that a raw file might not have the data to generate some of these files, leaving some epochs without the metrics data.

Having the metrics files generated, the script extracts the data from each file, storing data into the database at a pre-determined sample interval. Per example, if a file has the receiver temperature logged every fifteen seconds, it is possible to define that this variable should only be stored at ten minute intervals. But storing data daily, for each station on low sample intervals, leads to a great amount of data to be stored, and for that it was decided that the data would only be stored at this rate for thirty days, keeping the maximum, minimum and mean values for each day, on a separate table for historical purposes.

5.7.2 MIRAnet Integration

After the data are stored in the database, it must be made available to the MIRAnet administrator through the MIRAnet portal. With this in mind, a new page was created on the MIRAnet to display plots that help the visualization of the data, and tables to access and download the data with more detail.

Figures 5.20 and 5.21 show the options form used to generate plots and tables for the selected station. The 'Metric' dropdown allows users to select the metric to be displayed, while the 'Historical' button toggles between viewing historical or current data. When historical data is selected, users can specify a custom time interval. In contrast, selecting current data always displays information from the last thirty days

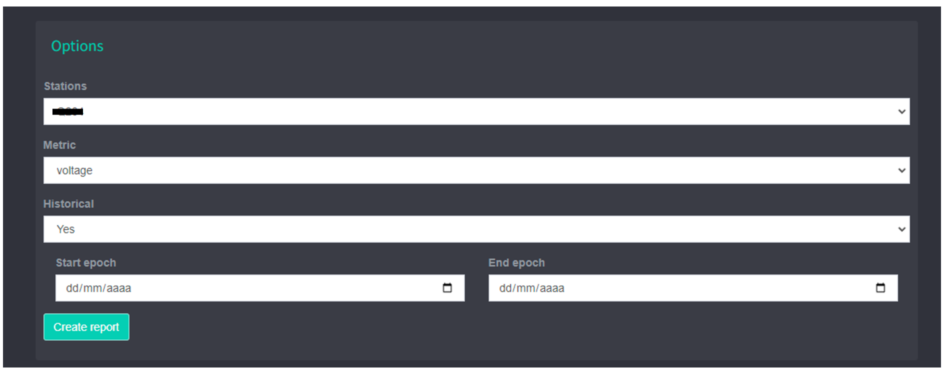


Figure 5.20: Options for Septentrio metrics (historical data)

Figure 5.21: Options for Septentrio metrics (current data)

The plots on Figures 5.22 and 5.23, are created using the Highcharts.js library with the data stored in the database. The Highcharts library was chosen over static plots due to its interactive capabilities, which offer greater flexibility and support the implementation of future features. The images show voltage, for a given station with historical and current data.

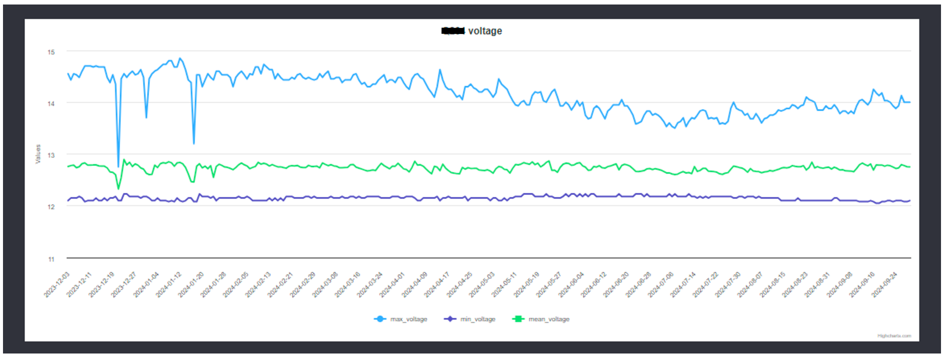


Figure 5.22: Historical plot for chosen station and metric

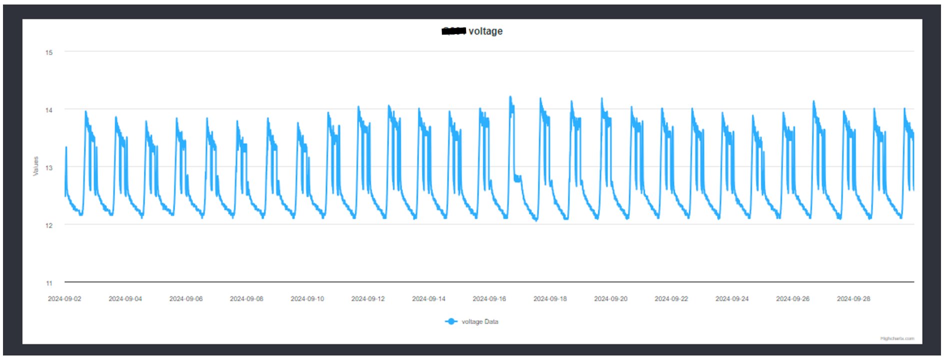
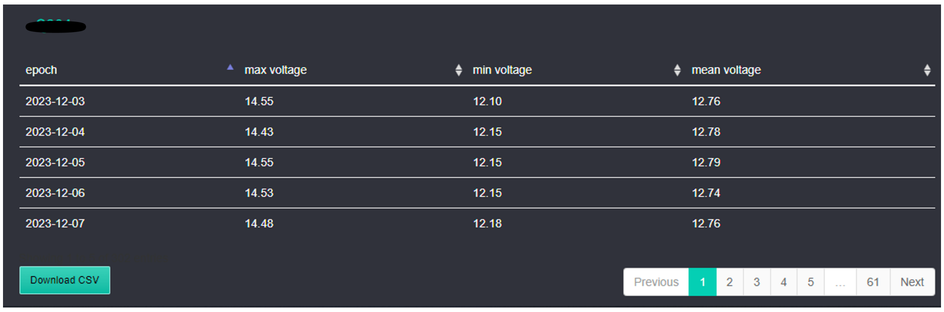


Figure 5.23: Plot for chosen station and metric for last 30 days

Additionally, the values are also present in table format (see Figures 5.24 and 5.25) which can be downloaded as a csv file.



[redacted]

epoch	max voltage	min voltage	mean voltage
2023-12-03	14.55	12.10	12.76
2023-12-04	14.43	12.15	12.78
2023-12-05	14.55	12.15	12.79
2023-12-06	14.53	12.15	12.74
2023-12-07	14.48	12.18	12.76

Download CSV

Previous 1 2 3 4 5 ... 61 Next

Figure 5.24: Historical table for chosen station and metric (voltage)

[redacted]

epoch	voltage
2024-09-02 00:00:00	12.53
2024-09-02 00:01:00	12.5
2024-09-02 00:02:00	12.5
2024-09-02 00:03:00	12.53
2024-09-02 00:04:00	12.5

Download CSV

Previous 1 2 3 4 5 ... 8064 Next

Figure 5.25: Table for chosen station and metric (voltage) for last 30 days

5.8 Conclusion

In this chapter, a description of the development of the different tasks was provided. The development stage took a logical sequence that enabled a smooth transition between tasks. It began with database modifications, proceeded to script development, and concluded with the integration of new functionalities into the MIRAnet web application.
In summary, the success of the development was due to the previous careful planning that went on the early stages of the internship.

Chapter 6

Conclusion and future work

6.1 Conclusions

The primary objective of this internship was to develop and implement procedures to collect, store, visualize, and assess the quality control of GNSS observations, a critical component in ensuring data integrity and accuracy in geospatial and GNSS-related applications.
During my time ate MIRASpaco, I had the opportunity to collaborate closely with a small but highly skilled team, which significantly deepened my appreciation for the importance of teamwork in a professional setting. This experience not only reinforced my understanding of software development processes but also enhanced my problem-solving skills as I tackled practical challenges during the implementation phase.
Integrating into a new team was initially challenging, as it required quickly adapting to the established dynamics and communication styles. Through active listening, asking questions, and engaging in discussions, I gradually found my place within the group.
Another significant challenge was quickly learning the basics of geodesy, a field that was relatively new to me. Given that my project involved working with RINEX files and ensuring their quality, understanding the fundamentals of geodesy became essential.
By the end of the internship, all the goals were successfully achieved. MIRAnet now has a fully operational quality control procedure in place for monitoring GNSS observations.

6.2 Future work

Even though the goals were achieved, there is still room for improvement. Currently, the QC plots are generated as static images, but it could be worthwhile to explore the possibility of creating real-time dynamic plots directly integrated into the MIRAnet UI using libraries like Highcharts. Additionally, enhancing MIRAnet's capability to support more brands in the procedure that extracts receiver health and status-related data would strengthen its functionality as a comprehensive receiver health monitoring service. This ongoing process of refinement not only underscores the importance of innovation but also sets the stage for future advancements in the system.

Bibliography

[1] MIRASpaco, "Miraspaco company profile," PDF, 2022. 1

[2] Universidade da beira interior. [Online]. Available: https://www.ubi.pt/ 1

[3] Home - rx networks. [Online]. Available: https://rxnetworks.com/ 1

[4] R. E. N. John M. Dow and C. Rizos, "The international gnss service in a changing landscape of global navigation satellite systems," *Journal of Geodesy*, vol. 83, no. 3, pp. 191–98, 2009. 1

[5] W. Gurtner and L. Estey, "Rinex-the receiver independent exchange format-version 3.00," *Astronomical Institute, University of Bern and UNAVCO, Bolulder, Colorado.*, 2007. [Online]. Available: https://epic.awi.de/id/eprint/29985/1/Gur2007a.pdf 3, 7

[6] (2024) A tool for checking the quality of multi-gnss observations and content of data files/streams disseminated in standard formats (rinex 2/3, rtcm 3). [Online]. Available: https://gnutsoftware.com/software/anubis 3, 14

[7] R. Fernandes, C. Bruyninx, P. Crocker, J.-L. Menut, A. Socquet, M. Vergnolle, A. Avallone, M. Bos, S. Bruni, R. Cardoso *et al.*, "A new european service to share gnss data and products," *Annals of Geophysics*, vol. 65, no. 3, pp. DM317–DM317, 2022. 4

[8] P. L. P. S. P. K. S. G. P. P. Amit Kumar, Shubham Kumar, "Gps and gnss technology in geosciences," 2021. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/B9780128186176000019 5

[9] P. Daly, "An introduction to the USSR GLONASS navigation satellites," *Journal of the British Interplanetary Society*, vol. 46, no. 10, pp. 385–390, Oct. 1993. 5

[10] J. P. Bartolomé, X. Maufroid, I. F. Hernández, J. A. López Salcedo, and G. S. Granados, "Overview of galileo system," in *GALILEO Positioning Technology*. Springer, 2014, pp. 9–33. 5

[11] Y. Lu and Y. Lu, "Brief introduction to the gps and beidou satellite navigation systems," *BDS/GPS Dual-Mode Software Receiver: Principles and Implementation Technology*, pp. 37–72, 2021. 5

[12] J. Kouba, "A guide to using international gnss service (igs) products," 2009. [Online]. Available: http://acc.igs.org/UsingIGSProductsVer21.pdf 5

[13] (2020) Data. [Online]. Available: https://igs.org/data/ 6

[14] Formats. [Online]. Available: https://igs.org/wg/rinex/#documents-formats 7

[15] S. S. Kumar and A. K. Chauhan, “Automated quality control of global navigation satellite system (gnss) data,” in *9 th Biennial International Conference & exposition on Petroleum Geophysics*. [Online]. Available: https://spgindia.org/spg_2012/spgp315.pdf 10

[16] (2023) Broadcast ephemeris data. [Online]. Available: https://cddis.nasa.gov/Data_and_Derived_Products/GNSS/broadcast_ephemeris_data.html 11

[17] (2024) Quality control: What it is, how it works, and qc careers. [Online]. Available: https://www.investopedia.com/terms/q/quality-control.asp 13

[18] N. u. S. Arif Iqbal, “Point–qc – a tool for gnss data quality control parameters,” 2016. [Online]. Available: https://www.researchgate.net/publication/308467287_PoiNT_-_QC_-_A_Tool_for_GNSS_Data_Quality_Control_Parameters 13

[19] Y. S. J. H. Dongha Lee, Jaemyoung Cho and H. Yun, “A new window-based program for quality control of gps sensing data,” 2012. [Online]. Available: https://www.mdpi.com/2072-4292/4/10/3168 14

[20] F. B. D. P. Christine M. Puskas, Charles M. Meertens, “Evaluating anubis software for gnss data quality control in the gage facility,” Poster presented at AGU Fall Meeting 2019, 2019. [Online]. Available: https://www.unavco.org/data/gps-gnss/derived-products/docs/Poster_2019_SAGE-GAGE_Workshop_Puskas_Introduction-to-Anubis-software-for-GNSS-quality-control-in-the-GAGE-Facility-and-NOTA.pdf 14

[21] F. A. for Cartography and Geodesy, “Bkg ntrip client (bnc),” PDF, 2013. [Online]. Available: http://acc.igs.org/misc/bnchelp.pdf 14

[22] The bkg gnss data center. [Online]. Available: https://igs.bkg.bund.de/ 14

[23] Epos. [Online]. Available: https://www.epos-eu.org/ 14

[24] Runqc. [Online]. Available: https://gitlab.com/gpseurope/RunQC 15

[25] Euref permanent gnss network. [Online]. Available: https://epncb.oma.be/ 15

[26] Measuring our changing earth. [Online]. Available: https://www.unavco.org/ 15

[27] Html introduction. [Online]. Available: https://www.w3schools.com/html/html_intro.asp 18

[28] Css introduction. [Online]. Available: https://www.w3schools.com/css/css_intro.asp 18

[29] Javascript. [Online]. Available: https://developer.mozilla.org/pt-BR/docs/Web/JavaScript 18

[30] Introduction. [Online]. Available: https://getbootstrap.com/docs/5.0/getting-started/introduction/ 19

[31] What is php? [Online]. Available: https://www.php.net/manual/en/intro-whatis.php 19

[32] Python introduction. [Online]. Available: https://www.w3schools.com/python/python_intro.asp 19

[33] Matplotlib: Visualization with python. [Online]. Available: https://matplotlib.org/ 19

[34] What is mysql? [Online]. Available: https://dev.mysql.com/doc/refman/8.4/en/what-is-mysql.html 19

[35] The visual database design management tool. [Online]. Available: https://dbschema.com/ 19

[36] Rxtools: Gnss receiver control and analysis software. [Online]. Available: https://www.septentrio.com/en/products/gps-gnss-receiver-software/rxtools 20

[37] Sbf analyzer - front end gain. [Online]. Available: https://customersupport.septentrio.com/s/article/SBF-Analyzer-Front-End-Gain 40