GRANADA UNIVERSITY (UGR)

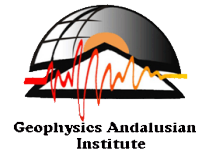**Theoretical and Cosmos Physics Department**

Geophysics Andalusian Institute

THEORETICAL PHYSICS AND THE COSMOS DEPARTMENT

ANDALUSIAN INSTITUTE OF GEOPHYSICS AND PREVENTION OF SEISMIC DISASTERS

"Individual Seismic Data Download from the FDSN network"

(Sistema para descargar datos sísmicos individuales de la red FDSN)

USER MANUAL: VERSION 1.0

Author:

Ligdamis A. Gutiérrez E. PhD.

Masaya Volcano, Nicaragua
Photo by: Ligdamis A. Gutiérrez E.

Granada, Spain 2021- 2023

General Index

1.- Introducción

The "***Individual Seismic Data Download from the FDSN network*** *(Sistema para descargar datos sísmicos individuales de la red FDSN)*" module, it features a simple and user-friendly interface that allows for easy and efficient management of individual searches for seismic records on the FDSN global network. The program, using a series of parameters such as time and network client entered by the user, searches for the specific record from a given network at a station, displaying either one or all three components (North-South, East-West, and Vertical). The main interface is available in English. Information on the folder structure and contents can be found in the appendices. The program offers easy access to two options for obtaining the most commonly used seismic record formats (MSEED and SAC) to store earthquakes from the FDSN (International Digital Seismograph Network) global network in the user's chosen folder or directory. It allows for a reliable, easy-to-use, and fast automatic tool that can help operators at research institutes and observatories store seismic records that are otherwise difficult to obtain and analyze, either because instruments or equipment are not available, or because resources do not allow campaigns to the sites designated for this purpose.

The first version of this system consists of a single interface that includes tools, libraries, and built-in functions to allow the download of a signal in either of two formats: MSEED and SAC. These formats can be used to save the signal from a specific FDSN client, network, station, or one or all three channels that make up the signal (for this, the exact parameters to be searched must be known). The system also allows the user to specify the storage duration of the recording, for example, one or 24 hours. This ensures the information is most useful for performing the respective signal analyses. Various filtering techniques, spectral analysis, and other methods can then be applied using the system's various modules. This automatically adds value to the operator's knowledge, enabling faster and more accurate determination of the characteristics of continuous seismic waves than manual methods.

The module and the entire system have been developed in Python, version 3.8.6. (The set of libraries is compatible with version 3.10.10). Additionally, a series of open-access libraries are included, which, in conjunction with Python, enable the use of graphical and analytical tools, providing ease of use and enhancing computational power for the user. Some of the main elements and libraries used are listed below:

- o **Matplotlib**: Used for creating static, animated, and interactive visualizations in Python. (https://matplotlib.org/stable/users/index.html).
- o **NumPy**: A library for numerical operations in Python. (https://numpy.org/doc/stable/user/quickstart.html).
- o **PyQt5**: A tool that links with the graphical library Qt5 in C++ (https://pypi.org/project/PyQt5/).
- o **Obspy**: A Python toolbox for seismology. (https://docs.obspy.org/).
- o **Tkinter**: Graphical User Interface (GUI) (https://docs.python.org/3/library/tkinter.html)

Another key feature of the system is its definition as a cross-platform application, meaning it can operate on various platforms or operating systems, such as Windows (7, 8, 10, 11) in both 32-bit and 64-bit versions. It also supports Linux systems, such as Ubuntu and other similar systems (Debian, Red Hat, Fedora, SUSE, etc.), macOS, and Android for tablets and mobile devices (with Python appropriately adapted for these devices).

NOTE: In the appendices of this document (*as well as in the Readme.txt and Initial_requirements.txt files*), you can find general information on installation for Windows and Linux systems, as well as guidelines for installing the main programs and additional libraries required by Python to properly execute the developed programs in its environment.

2.- Initial Screen of the System

In the appendices of this document and in the "**README.txt**" file included in the "***Documents***" folder, you will find instructions for installing the system on Windows (*the process on Linux systems is similar*). Essentially, you need to perform two actions:

a) Copy the "Set_tools_System_1_1" folder to "My Documents" on Windows.
b) Copy the "Set_tools_System_1_1.bat" file to the Windows desktop.

Additionally, there are instructions for installing the necessary Python libraries on the system. Once "Set_tools_System_1_1.bat" has been copied to the desktop, you need to right-click on it and select "Run as administrator."

Fig. 1 Popup Window when Right-Clicking on the "Set_tools_System_1_1.bat" File

In the window that opens, click the "Yes" button when prompted with "*Do you want to allow this app to make changes to your computer*?" This is a warning message. However, the application does not make any changes, so you should trust its execution.

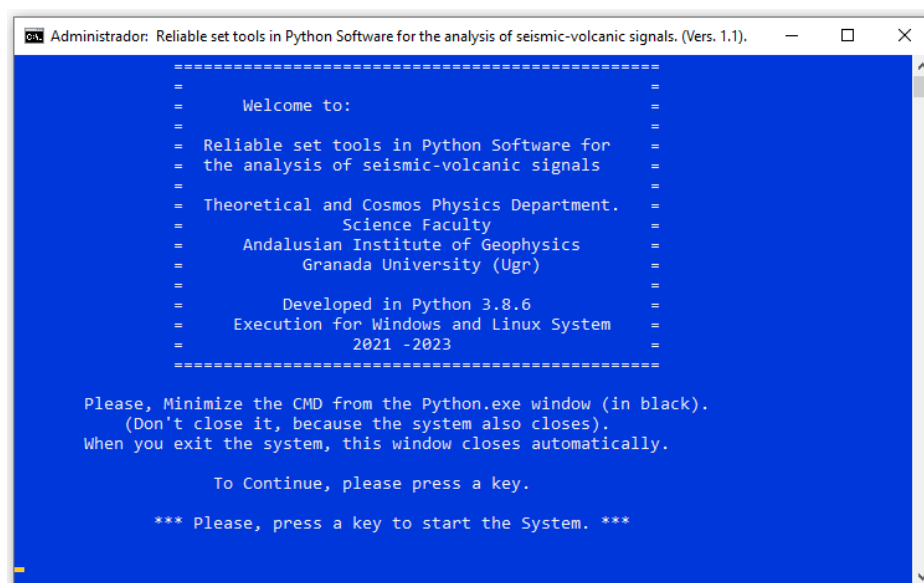Upon clicking "*Yes*," the following command window opens, welcoming you to the system.

Fig. 2 Welcome Screen and Instructions for Loading the System.

After reading the information in the window, you simply need to press any key to access the system's initial screen. The folder should already be copied to "***My Documents***," and the "**Set_tools_System_1_1.bat**" file contains all the loading instructions.

The system's initial screen is "**Menu.py**". It appears when any key is pressed on the Welcome screen. Additionally, the Python command window or console is displayed, similar to the following
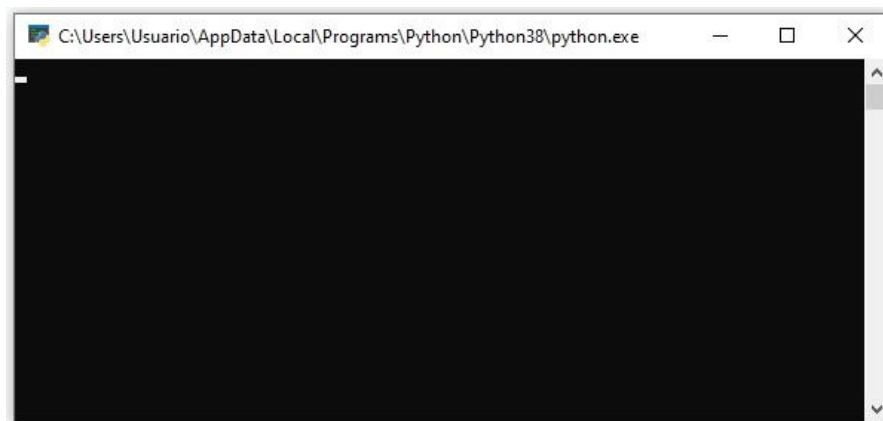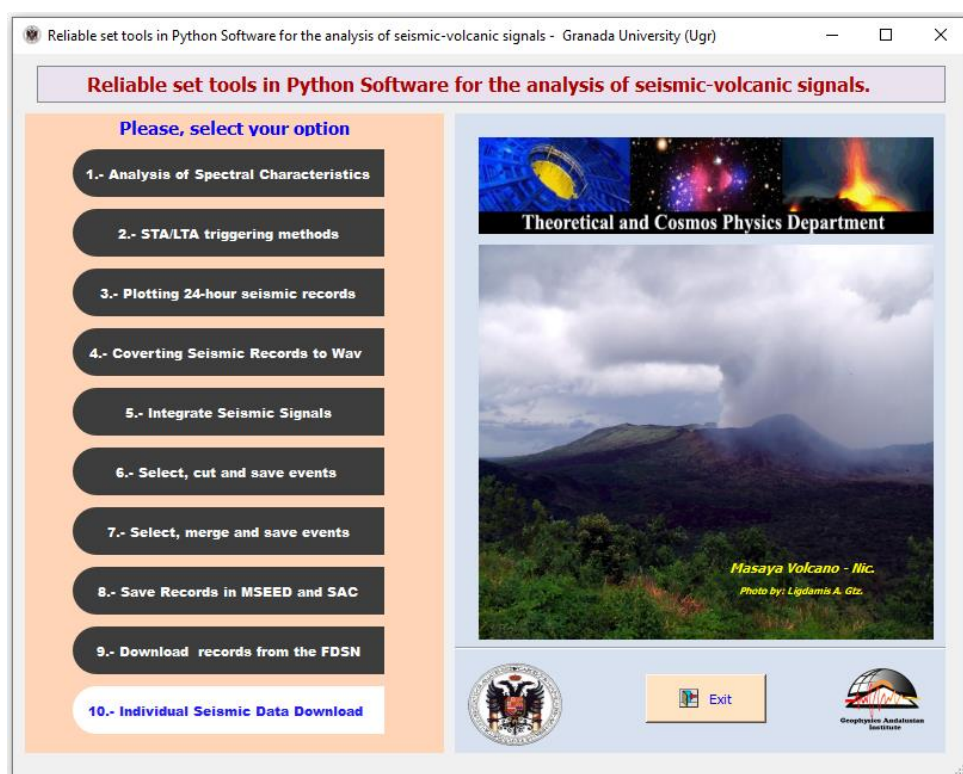

Fig. 3 Python Console (CMD) Window (*Should be minimized*)

To avoid obstructing the view, you can and should "minimize" this screen. Do **not** close it, as this would also close the system's startup window. Once you have finished working with the system, this window will close automatically. The initial presentation screen of the system (the module menu) "**Menu.py**" is as follows:.
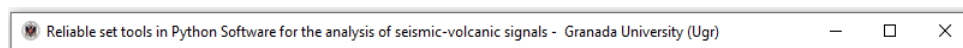

NOTE: When pressing or clicking on a module button, the startup window will close and the module window will open (*this may take a little time depending on the PC's memory. It is recommended to have at least 8 GB of memory in the system, with 16 GB being ideal*).

Fig. 4 **Main Menu Screen.** The module to be worked on is highlighted. Module 10 (*Individual Seismic Data Download*).
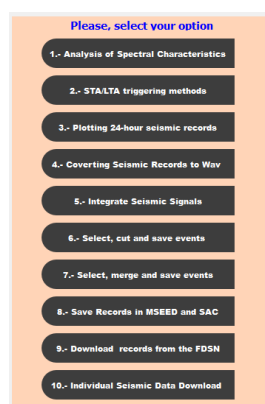
2.1.- Elements of the Initial Screen.

As observed in the previous figure, the initial or presentation screen is a simple window composed of:

a) A top toolbar with basic information about the module.
b) On the left side, there are 10 execution buttons or command buttons for each module of the system.
c) At the bottom, there is a command button that allows for exiting the system.
d) Additionally, it features a background image representing a volcano (Masaya in Nicaragua), and three images with the logos of the University of Granada, the Andalusian Institute of Geophysics, and the Department of Theoretical and Cosmic Physics.

a) At the top, the icon of the University is visible, along with the module title and a reference to the University of Granada (UGR).

b) On the left side, there are 10 execution buttons or command buttons for each module of the system. When the mouse pointer is placed over each button, it is highlighted in white to indicate that it is being selected. Clicking on a button closes the startup menu window and opens the window for the indicated module (this may take a little time depending on the PC's memory).

c) At the bottom, there is a command button: **Exit**. When the mouse pointer is placed over each button, a text appears indicating the action of that button (Exit System, Start System).

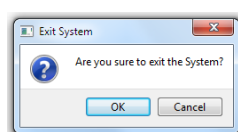If you click the "Exit" button, a window will appear asking the user if they are sure they want to leave the system.

Fig. 5 Text Box Indicating Whether You Want to Exit the System.

If you click "**OK**," the screen will close and the system will exit. If you click "Cancel," you will remain on the initial screen.

3.- Main interface for individual record downloads.

The "search screen" is the system's main interface, where the activities that comprise the data reading tools, or input parameters for searching and downloading seismic records from the FDSN network, are performed. This screen consists of the following parts:
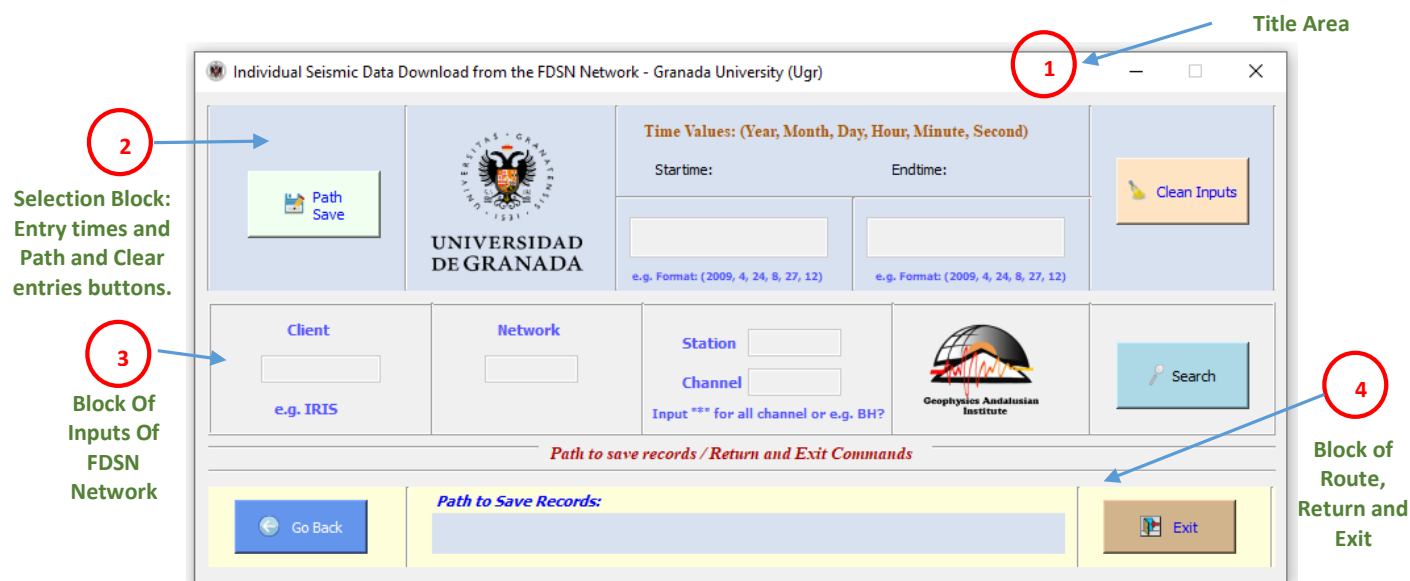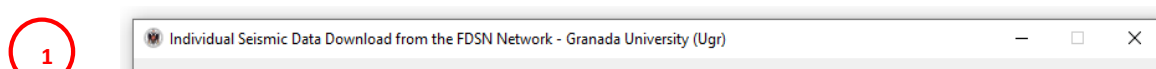

Fig. 6 Elements of the System Analysis Screen.

1) Title Area: Name of the Program and the University.
2) Selection Block: Path or directory where the records will be saved, Start and End Time data entry, Clearing Inputs: a) Command buttons: Path/Save, Clear Input, b) Start Time and End Time data entry, whose input formats are defined by a simple input format with values of: "Year, Month, Day, Hour, Minute, Second". [The comma separator between each value must also be respected].
3) FDSN Network Input Block: This includes inputs for the following values: Client, Network, Station, and Channel. For the Channel input, you can use wildcards like "*" for all channels or "?" to specify that it can be any letter, for example (E, N, Z). Thus, for the BH? Input, the "?" can mean "BHE", "BHN", or "BHZ". And the Search button.
4) Route, Return, and Exit Block. a) Physical location path of the record to be analyzed, b) Command buttons (Go Back, Exit).

The main security interface initially displays the entries and the **Search** command button disabled. They are activated by clicking the **Path/Save** button, which selects where the log will be stored.

The screen is made up of various elements for its use. At the top you will see: Program name, icon, University name, and author as the title (1).

The elements that make up the main screen are detailed below.

3.1.- Elements of the search screen
In addition to number (1), the 3 elements of the search screen have been distributed into three main blocks, which are listed from (2-4) in the red circles.

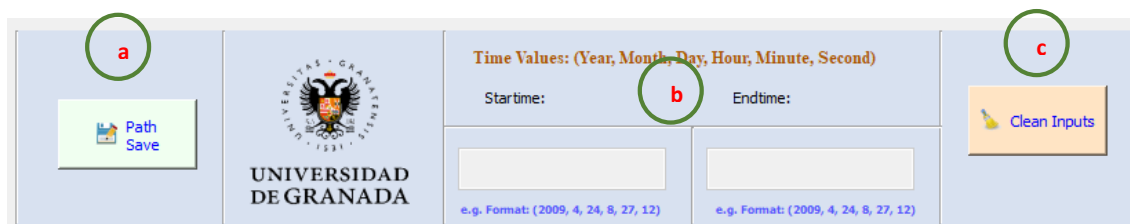**3.1.1.- Selection Block**: Folder, times (Start/End) and Entry Cleanup.

Fig. 7 Selection elements block. In the green circles: a) Command button: Path Save, b) Data input: Starttime and Endtime, c) Command button: Clear Inputs.

This block contains the path or directory where the records will be saved, the Start and End Time data inputs, and the Clear Inputs button: a) Command button: Path/Save, b) Start Time and End Time data inputs. c) Clear Inputs button. NOTE: The input format is defined with the values: Year, Month, Day, Hour, Minute, Second. [The comma separating each value must also be respected]. For example, a valid input would be: (2006, 4, 24, 8, 27, and 12)

3.2.- Selection block elements.

This block is configured (***green circles in the previous figure***) firstly, by the section that groups the action buttons to select the directory or folder where the logs and data cleaning will be saved: a) Command buttons: Path Save (a), Clean Input. (c) Also, the entry of the start time (Startime) and the end time (Endtime) (b)
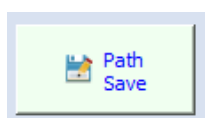
When you place the mouse pointer over the two action buttons, a message displays the actions they perform.

Fig. 8 Select Directory and Clean Data buttons. Action messages are displayed on each button.

As shown, the "**Path Save** (Select Directory to save record)" button searches for and loads seismic records in various formats. The "**Clean Inputs**" button clears or deletes input elements, and also disables data entry and the "**Search**" button.

3.2.1.- Select Directory button.

Clicking the "Path Save (Select Directory to save log)" button opens a file explorer window (by default, the path is located in the root directory "C" of the PC), presenting the options to select the directory or folder on the computer where the logs will be saved. This is shown in the following screenshot. The data entry fields, which are initially inactive, become active when this "Path/Save" button is clicked, as does the "Search" button.

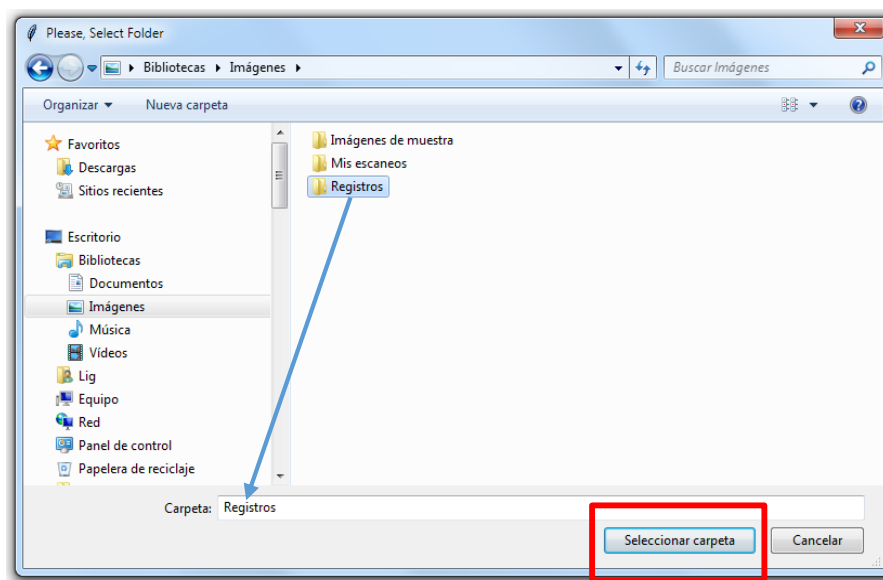Clicking the Select Directory (Path/Save) button displays the following:

Fig. 9 Screen for selecting the folder or directory where the downloaded records will be saved.

On this screen (the language is determined by the operating system), select the desired folder. Using a search similar to navigating directories in Windows, the folder name is placed in the "Folder" area. Clicking the "Select Folder" button (**red box**) moves this name to the "**Path to Save Record**" area, in block number (4) of the main interface, detailed later in this document.

NOTE: Even after selecting a folder and clicking "Cancel," the selected folder will be assigned to "/" by default, indicating the root directory or "C". The stored records should be searched for there. However, it is always recommended to select a predefined folder.

At this point, clicking the "Path/Save" button loads the path to the physical location where the downloaded records will be saved. This path will be displayed in the "Path to Save Record" box, located at the bottom of the analysis screen.

Fig. 10 File path box, which displays the location of the record.

3.2.2. – **Entry time boxes**: Startime and Endtime.   **b**

Fig. 11 Startime and Endtime Data Input Selection Screen

Initially, these data entry fields are disabled, as shown in the figure. Data entry in this field is activated by clicking the folder selection button (Path/Save). The image shows the simple, valid input format that must be entered, separated by commas. The order is: **Year, Month, Day, Hour, Minute, Second.** By using this accessible format, the user does not need to worry about whether the time format will be valid for the search. The system automatically converts these entries to UTC (Coordinated Universal Time), which is the main global time standard used to regulate clocks and time, and which is used by the FDSN network. For example, the value entered in the image (*2009, 4, 24, 8, 27, 12*) will have its equivalent in UTC format: 2009-04-24T08:27:12.000000Z. These calculations are performed internally by the program.

However, you must ensure you provide a valid date so that the UTC format is also valid. This means that the events you want to download must exist on the specific date and at the location you are searching for. Otherwise, the system will display an error message prompting you to enter valid values.

There are three data input validations for this object.

a) If no value is entered in the Start time text box, leaving it empty, a window will appear indicating this error when performing a search, allowing the user to enter a valid value.
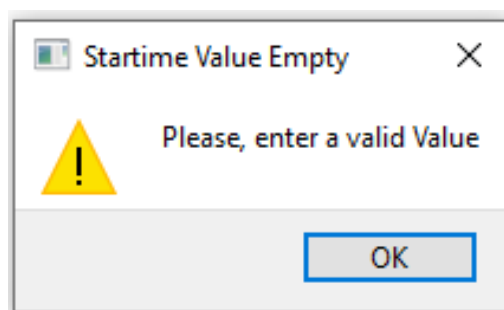

Fig. 12 Screen for validating empty entries in the "*Startime*" data box

b) If no value is entered in the End time text box, leaving it empty, a window will appear indicating this error when performing a search, allowing the user to enter a valid value
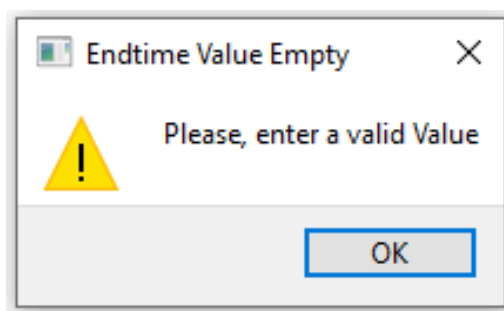

Fig. 13 Screen for validating empty entries in the "*End time*" data box

c) The third validation checks for invalid entries, incorrect formatting, or invalid dates in either of the two time boxes.

In this case, when you click the Registry Search button, the following appears:
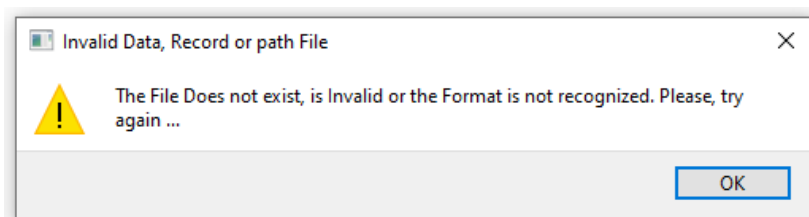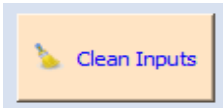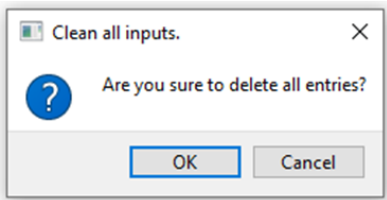

Fig. 14 Screen for validating erroneous or invalid entries in the "**Start time**" and "**End time**" data boxes.

3.2.3.- "Clean Inputs" button.

 Clicking this button clears all entries on the screen. It will clear the data entry text boxes and the path or folder where the downloaded record is stored, deleting all currently active entries containing data. The text boxes and the Search button will be deactivated, restoring the main interface to its initial state (see Fig. 6). Hovering the mouse pointer over the button displays a message indicating its function. Clicking the button opens a window asking the user if they are sure they want to delete the data entries. If yes, all entries are deleted, and the interface returns to its initial state. Otherwise, the current entries remain on the interface.

4.- FDSN Network Entry Block

Fig. 15 FDSN Network Input Block: This includes inputs for the following values: Client, Network, Station, and Channel. For the Channel input, you can use wildcards like "*" for all channels or "?" to specify that it can be any letter, for example (E, N, Z). Thus, for the BH? Input, the "?" can mean "BHE", "BHN", or "BHZ". And the Search button.

In the figure above, you can see the block that determines the configuration parameter inputs that establish the download of a record from the FDSN network.

The entries marked in green circles are as follows: It is recommended that you enter them in the following order:

a) **Client**: Establish the FDSN client. In the example case, the client being called is "IRIS (Incorporated Research Institutions for Seismology)".

b) **Network**: Here you must enter the IRIS network you want to use.

There is a wide variety of networks, both public and those linked to research projects associated with IRIS. A list of these possible networks can be found on the following website:
https://www.fdsn.org/networks/

c) (1) **Station**: Determine the seismic station within the network from which you wish to obtain data. A list of seismic stations worldwide can be found on the following website:
http://www.iris.washington.edu/gmap/#network=_GSN&planet=earth

(2) **Channel**: This establishes the channel, which refers to the three components or directions of seismic movement in the earth: North-South (N), East-West (E), and Up-Down or Vertical component (Z). Depending on the equipment or how the sensors are configured, it may have several names, but generally the letters (E, N, and Z) are assigned, respectively. The starting nomenclature can also change; for example, BH (continuous recording 20 mps), LH (continuous recording 1 mps), VH (continuous recording 0.1 mps), among others. The wildcard character "*****" is used to designate all channels, or "?" for any channel.

There are data input validations for each of the elements of this object.

i) If no value is entered in the Client text box, leaving it empty, a window will appear indicating this error when performing a search, allowing the user to enter a valid value.
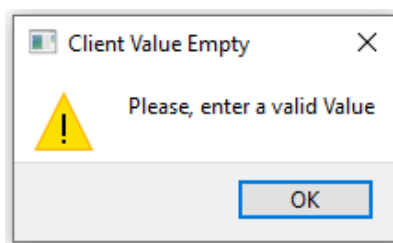

Fig. 16 Validation screen for empty entries in the Customer data box "**Client**"

ii) If no value is entered in the Network text box, leaving it empty, a window will appear indicating this error when performing a search, allowing the user to enter a valid value.
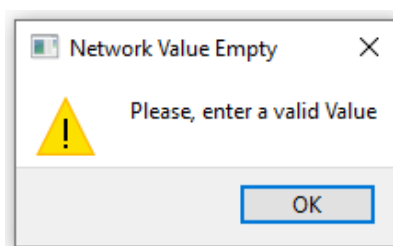

Fig. 17 Screen for validating empty entries in the "**Network**" data box

iii)   If no value is entered in the **Station** text box, leaving it empty, a window will appear indicating this error when performing a search, allowing the user to enter a valid value.

Fig. 18 Validation screen for empty entries in the **Station** data box.

iv)   If no value is entered in the **Channel** text box, leaving it empty, a window will appear indicating this error when performing a search, prompting the user to enter a valid value.

Fig. 19 Validation screen for empty entries in the **Channel** data box.

v)   If invalid entries, incorrect formatting, or incorrect data are entered in any of the text boxes above, a window will appear indicating the error, allowing the user to enter a valid value.
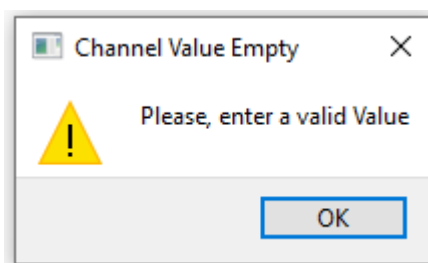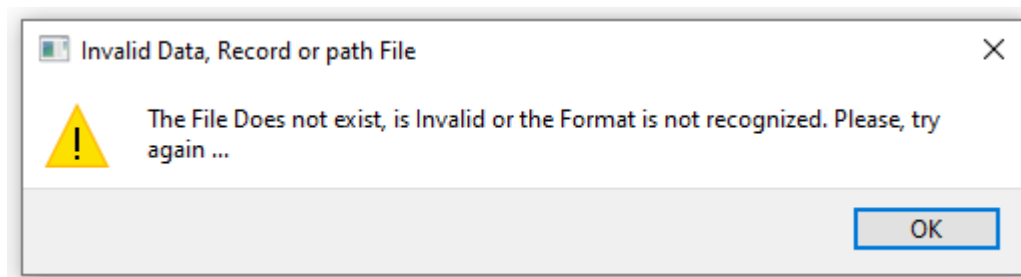

Fig. 20 Pantalla de Validación de entradas erróneas o inválidas en los cuadros de datos de texto.
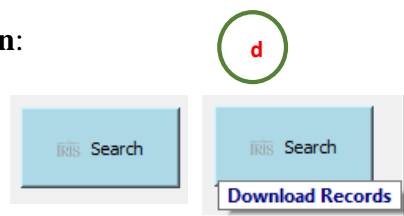
d) **Search** Button:

Fig. 21 Search button. Downloads the seismic record according to the entered parameters. An action message will appear on the button.

Once all parameters have been entered, the search and download of record from the **FDSN** network is performed by clicking this button. Here, the user also decides in which format (**MSEED** or **SAC**) to save the record. This will be covered in more detail in section 6 (*Process of selecting the path and downloading logs from the FDSN network*).

5.- Route, Return and Exit Block

This block is configured with: Two command buttons: a) *Go Back* and b) *Exit*. Additionally, c) the area where the physical path to the seismic record for loading will be displayed (*Record Path to Load*).

Fig. 22 Command button block (a) Go Back, (b) Exit) and physical path of the file to load (c).

This last block is made up of the following elements:

5.1.- *Go Back button*

Allows you to return to the system's initial splash screen (**Menu.py**). When you hover the mouse pointer over it, it displays a message indicating its function.

Fig. 23 *Go Back Button*.

5.2.- *Exit button*

Allows complete system exit (after displaying a screen asking if you wish to exit). Hovering the mouse pointer over it displays a message indicating its function.

Fig. 24 *Exit Button*.

Similarly, to the home screen, if you press or click the "**Exit**" button, a window appears asking the user if they are sure they want to leave the system.

Fig. 25 Text box indicating whether you wish to exit the system.

 Clicking "**Yes**" (**OK**) closes the screen and completes the system exit. Clicking "**Cancel**" (**No**) continues to the analysis screen.

5.3.- File path display area (Record path to Load).   c

This area displays the path (***Disk/folder/file***) to the physical file where the downloaded record will be stored on the computer. This path is accessed via the "***Path/Save***" button, as seen previously.
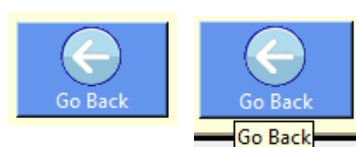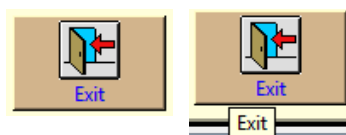

An example of the output when selecting the folder can be seen in the following image.

The "***Path***" (on the computer) to the physical location of the record to be downloaded is displayed.

6.- Process of selecting the route and downloading the records from the FDSN network.

The process of downloading the records is extremely simple and includes a series of steps that are best followed in the order presented. These steps are:

a) Select the folder where the records will be saved using the "Path/Save" button.
b) Select the start time (**Starttime**).
c) Select the end time (**Endtime**).
d) Select the **FDSN network** client to use (e.g., IRIS).
e) Select the **network** to use within that client.
f) Select the **station** belonging to that network.
g) Select the desired **channel**(**s**).
h) Click the "**Search**" button to start the download.
i) Select the desired file format (**MSEED** or **SAC**).

If for any reason you wish to cancel and reset or re-enter the parameters, you can use the "Clean Inputs" button to enter new parameters and folders from scratch.

Once all parameters have been entered, the main search interface will look similar to the following image.

Fig. 26 Main interface for searching and downloading records with the established parameters.

As shown in the image above, the parameters to be entered are the following:

a) **Startime**:   2012,12,14,20,36,01 (*From the 14th*)
b) **Endtime**:   2012,12,15,20,36,01 (*Until the 15th, it indicates 24 continuous hours*)
c) **Client**:   IRIS
d) **Network**:   TA
e) **Station**:   E42A
f) **Channel**:   BH? [*Wildcard "?" all components BHE,BHN and BHZ*]

Clicking the Search button will initiate the process of connecting to the **FDSN** network via the internet and performing the search. This may take a few minutes, depending on your computer's connection speed. Please be patient and wait for the process to complete. At the start of the process, a screen will appear allowing you to select the desired format (**MSEED** or **SAC**) for download from the network. (*As specified, 24 hours of continuous seismic data will be available for download*). The screen is as follows.

Fig. 27 Dialog box for selecting the desired download format: MSEED or SAC. MSEED is selected by default.

The user must click on the desired format. If no record is to be downloaded because a parameter needs to be modified, the user clicks "***Cancel***," which returns the action to the main interface screen, allowing the desired parameters to be modified.

Once the registry entry, whether **MSEED** or **SAC**, has been downloaded to the desired folder, a screen will appear indicating that the download was successful. It will look similar to the following.
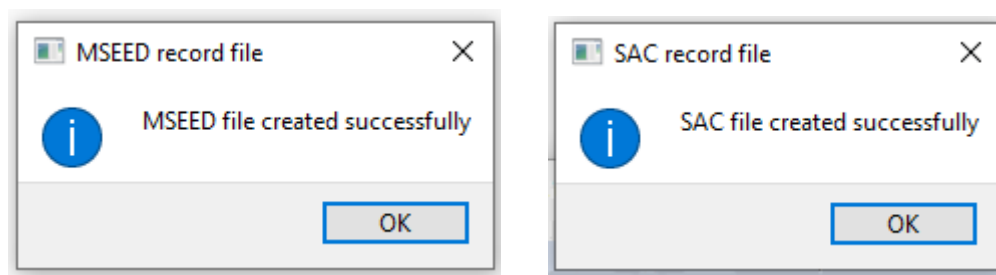

Fig. 28 Dialog box indicating that the record has been successfully downloaded.

Once one of the formats has been downloaded and "OK" has been clicked, the user is returned to the main interface, allowing them to download the other format if they wish.

IMPORTANT NOTE: *MSEED files are formatted to include all three components, so a single **MSEED** file contains all three stations (these are the traces in the log). **SAC** files, however, usually consist of only one component. Therefore, the download will result in one **MSEED** file for every three **SAC** files.*

The format that has been designated for storing the records is as follows:

Record + Red + Estación + Año + Mes + Día . (Tipo de formato)

a) MSEED output format: (1 only)

Record_TA_E42A_2012-12-14.mseed

b) SAC output format: (3, one for each component: H-N-Z)

Record_TA_E42A_2012-12-1401.sac
Record_TA_E42A_2012-12-1402.sac
Record_TA_E42A_2012-12-1403.sac

It is observed that the result of downloading in the SAC format generates three files, marked as 1, 2, 3 which correspond to the H, N, Z components [East-West, North-South, Up-Down (Vertical)].

Once downloaded, the user can use any of the analysis modules that make up this system to perform the necessary studies with these records.

CONCLUSION: The system is designed to be an easy-to-use, accessible, and understandable tool. It features a user-friendly interface that provides reliable technological support to the human operator in the process of downloading continuous and individual seismic records from the global FDSN (International Digital Seismograph Networks). It can download both tectonic and volcanic records. The simplicity of this first version lies in its single module, which includes the ability to store records in two of the formats frequently used in the study of a given seismic signal (MSEED and SAC). This allows for subsequent spectral analyses and studies of the requested seismic records. Subsequent versions or updates to the current version (1.0) may include additional tools with different functionalities or formatting and calculation methods that could be useful for improving the study and research of the scientific community.
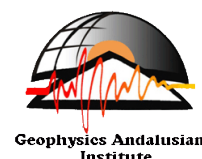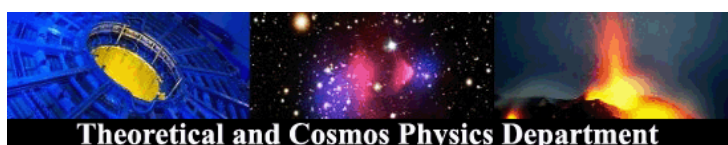
END of the document.

Ligdamis A. Gutiérrez E. PhD.
Theoretical and Cosmos Physics Department, Faculty of Sciences.
Geophysics Andalusian Institute.
Granada University (Ugr)
Granada, Spain – 2021 – 2023

APPENDIX A

A1.- Installation of Python and Additional Libraries.

A1.1. Package Content.

The main folder "***Analysis_System_1***", It contains two folders with the programs (codes and interfaces in English and documentation in Spanish and English), organized as follows:

a) Folder "***Reliable_set_tools_system_1***": (*seismic analysis system assembly*). This folder must be copied to "*My Documents*". Contains the following elements:

   a. Subfolder: "***Images***" Images necessary for program interfaces.
   b. Program: ***Menu.py***. Startup program, presentation and calling of individual modules.
   c. Program: "***ModIris_2.py***Main program (interface) for searching and downloading seismic records from the FDSN network and converting them into MSEED and SAC formats.

b) Folders (*Document_ES and Document_EN, depending on the language; English or Spanish*). For *English* "***Document_EN***": It consists of the following items:

   a. User Manual for the module: "***10_Manual_Individual Seismic Data Download FDSN_EN.pdf***" in PDF, written in English, with the necessary documentation for the use of the system interfaces. In addition, the rest of the manuals for the other modules (1-10) that make up the system are found.

   b. "***Initials Requirements.txt***" file. File containing the libraries needed to be installed on Windows through "Pip", once Python is installed.

   c. File "***README.txt***": File with general instructions for system installation.

   d. File "***Set_tools_System_1_1.bat***", batch processing executable file. It must be copied to the desktop, from there by right clicking "run as administrator", it will start the system by calling the main menu. The file will automatically search for the startup program (Menu.py) that is located in the "Set_tools_System_1_1" folder that has been previously copied to "My Documents" and will start Python, executing said program.

The system has all the elements (*programs and interfaces*) in English, except for the user manual, which is written in both Spanish and English. To install on Windows, two main actions should be taken after downloading and extracting the ". Rar" files. The first is to copy the entire folder (a) to the "*My Documents*" folder on the PC.

a) From the main folder (**Analysis_System_1**), copy the subfolder "**Document_EN**" to "**My Documents**" in Windows.

b) Copy the file "**Set_tools_System_1_1.bat**", from the "(Document/Document_ES or Document_EN)", depending on the version (Spanish or English), to the Windows desktop.

This ensures the proper use of the program. Now, we will proceed with the installation of the Python language and additional Python libraries on Windows.

A1.2.- Installing Python on Windows.

Python is an interpreted, multi-platform, and multiparadigm programming language (*it works on various operating systems, including Windows, Linux, and Mac*), utilizing two or more programming paradigms within a program-object-oriented, reflective, imperative, and functional.

In addition, Python can be enriched by a large number of programming modules, libraries, packages, or libraries installed through its package manager, "**Pip**." On Linux, the Python program and its manager "**Pip**" are installed together with the operating system. In Windows systems, however, where Python is not a native language, it is necessary to install this language beforehand by downloading the appropriate version from the Python distribution website at the following address: https://www.python.org/downloads/

On the website, the correct version should be selected based on the type of operating system on the computer, including whether it is 32 or 64 bits.

To be installed on both 32 and 64-bit systems, it is essential to note that this document and the software were created with the version available at that time, which was "Python 3.8.6", and many more versions have emerged since then. A more modern and adaptable version to the software (recommended) is "Python 10.10".

Users need to check if more advanced versions do not interfere with some of the installed libraries, such as "*Obspy*," for example. This is because everything related to Linux systems is constantly changing with updates that Python and Linux-based systems make. It is advisable to visit the website and download the most stable or tested updated version of Python that works well with this software.

Once downloaded, run it as an administrator (*right-click and "**run as administrator**"*), and the software installation wizard will guide you through the necessary steps (*just follow the instructions*).

The process takes only a few minutes. It is "recommended" to indicate during the process, when asked, to include an access path in the system's "**Path**" so that Python can be accessed from any location in Windows. If this is not done during the installation process, it must be done manually by modifying the environment variables (*more complicated*) to include the path from where Python is installed. This will not be necessary (*if indicated at the beginning*) through the installation wizard.

A1.3.- Installation of Additional Libraries.

The next step is to ensure that Python and its file manager or package manager (**Pip**) have been installed correctly. "**Pip**" (*file and library manager*) is crucial because it allows the installation of additional libraries that Python needs to run the created programs. To do this, open the Windows console window, or "*CMD*." The CMD, or command prompt, is a command-line interpreter.

Accessing the CMD is possible by typing, searching for the Windows logo key (a window), located between the "Ctrl" and "Alt" keys at the bottom left of the keyboard. Pressing this key, plus (+) the letter "**R**" key, will open a "Run" program window, similar to the following.
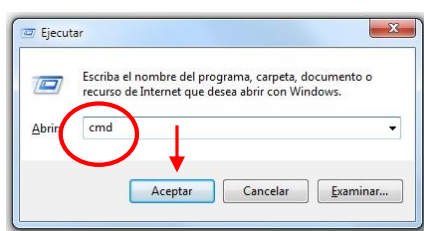
Fig. A1 Screen run in Windows. In the red circle, type "cmd" and click "OK."

As seen in the figure above, type "cmd," click "OK," which will open the Windows command prompt window.

Another way to do this is at the bottom of the desktop, in (W7) or next to (W10) the Windows "Start" button. There is the search section, indicated by the magnifying glass icon. This indicates a search for programs, similar to the following.

Fig. A2 Windows Program Search Screen.

In the box that says "Search programs and files" (Windows 7) or "*Type here to search*" (Windows 10), type "cmd" as well. This action or the previous one will bring up the Windows command prompt (CMD), similar to the following (W7).
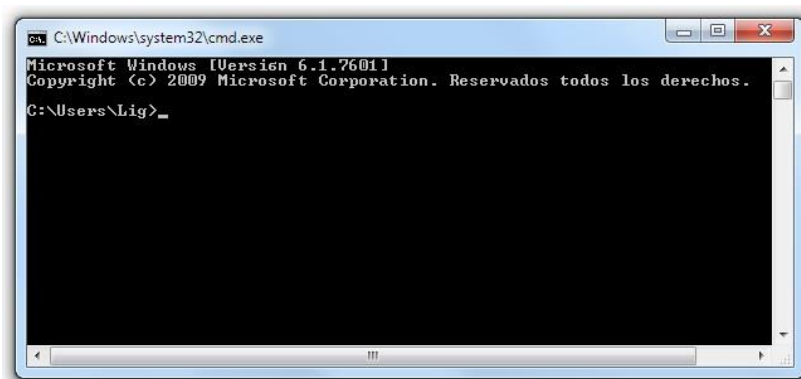

Fig. A3 Windows 7 Command Prompt (CMD) Screen.

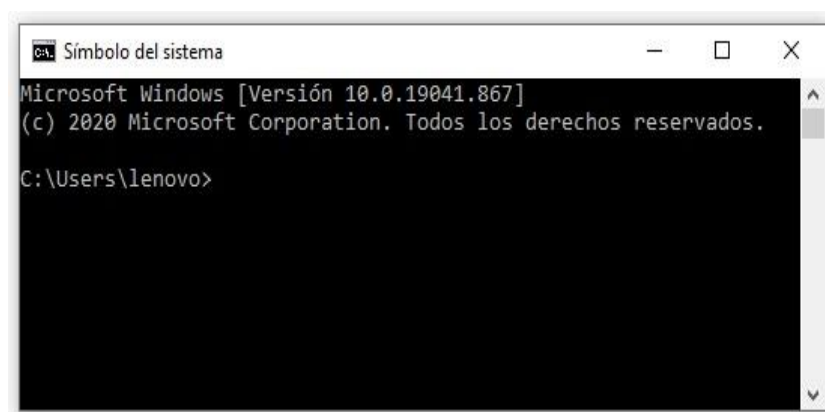The same applies to versions: Windows 10 (W10) or Windows 11 (W11).

Fig. A4 Command Prompt (CMD) Screen in Windows 10.

Once on this screen, to verify that both Python and its package manager "pip" have been installed correctly, type the following commands: Python -V, and to check "pip," type: pip -V. This is shown in the following figure.
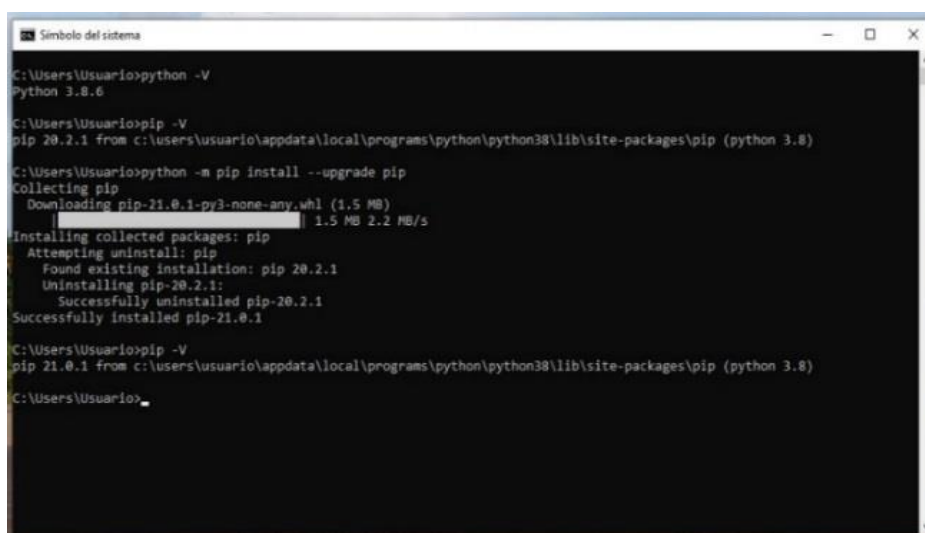

Fig. A5 CMD Screen, indicating Python and pip versions in Windows.

The output of typing "-V" in Python indicates invoking the installed version. In this case, it can be seen that it is "3.8.6" (*The library set is compatible with 3.10.10 version*). This has been possible from any location in the system because the Python script has been installed, remember, in the "**path**" or route found in the system's environment variables. Also, after typing "pip -V", it can be seen that the Pip version is "*20.2.1.*" At this point, it is recommended to update this version since, by default, "Pip" is installed along with "Python", but it does not install the latest or most up-to-date version. To do this, in the CMD window or console, type the following command (*Windows/Linux*): On Windows, type "python," and on Linux, type "python3".

Windows: > python –m pip install –upgrade pip | Linux: $ sudo python3 –m pip install –upgrade pip

This indicates that "Pip" will be updated to its most recent version (*On Linux, as a "superuser," i.e., with "sudo" at the beginning*). It is shown in the following screen.

Fig. A6 Screen showing the update and verification of the new version of pip in Windows.

As can be seen, when typing again (**pip -V**), once Pip is updated, the version is 21.0.1. With this, Python and Pip are already installed and updated. Pip, as mentioned, is very important because with this manager, all the necessary libraries and packages are installed so that Python applications can be executed correctly and without errors. To use the system, you must proceed to install the necessary packages or libraries through Pip.

Next, we will proceed to explain how, in a simple and completely automatic way, the most commonly used and general libraries that Python needs will be installed on the system. Libraries such as, for example, "*obspy*," which is the open-source library or software based on Python for processing seismological data. Also, "*matplotlib*," which is a library for generating graphics from data contained in lists or arrays in Python and its mathematical extension "*NumPy*," among others, which the system needs for its execution (*See Annex B*).

A1.4 Automatic Installation of Libraries on Windows and Linux from PIP.

The advantage of having already installed and updated Pip in Windows is that you can install all the libraries that Python needs to run the system.

Additionally, in the "*Document*" folder, the "*Readme.txt*" file contains instructions for this installation. So the user only needs to follow the instructions, and the necessary packages will be installed on the computer (PC) automatically by Pip, both on Windows and Linux. The required libraries are in the file called "*Initial_requirements.txt*", included in the "*Document*" folder of the downloaded installation files and in **Annex B**.

In a Windows Command Prompt (**Cmd**) window, actions are taken for each of the commands indicated in the file, following the instructions. The installation should not present problems on Windows and Linux systems. If any library encounters an error during installation (*shown in red in CMD*), you should consult the documentation for that library or check if the correct or recommended version of Python is being installed (*version 3.8.6 and/or 3.10.10*). The installation on Linux systems (*See README.txt*) is similar and simpler. Copy the main folder to the desktop, the personal folder, etc. From that location, open a command prompt, and simply type:

"*$ python3 Menu.py*" to start the system.

APPENDIX B:

INSTALL PYTHON LIBRARIES FOR THE PROPER FUNCTIONING OF THE SYSTEM.

1.- **PIP**: The **Pip** (*Preferred Installer Program*) is the package or package management manager used to install and manage software packages written in Python. When installing Python, Pip is installed by default. To check the version of Python or PIP, type the following in a console or CMD:

python - V /   pip – And to see the list of installed pip packages: ->   pip list

Usually, you need to update the version of pip with which Python is installed. For this, type the following command in the command prompt (CMD). In Linux and Mac systems, "**sudo**" is placed at the beginning to indicate super-user permissions.

   Python –m pip install - - upgrade pip   / (LINUX) -> sudo python –m pip install - - upgrade pip

Once downloaded and installed, you can check the version again with the first command, and you will see that the version has changed and been updated. Now that pip is updated, we will proceed to install the necessary packages for Python to work correctly with the applications.

2.- .- **PyQt Installation**: This is a Python binding for the Qt library written in the C++ language. It is used for creating and using graphical user interfaces (GUI) in Python. Type the following in the command prompt (CMD).

      pip install PyQt5   / (LINUX & Mac) -> sudo python install PyQt5

3.- **Matplotlib library Installation**. Matplotlib is the library that allows the creation and visualization of graphics. Type the following:

      pip install matplotlib   / (LINUX & Mac) -> sudo python install matplotlib

4.- Install the **Obspy** library. This library is for handling seismic signals. Type the following:

      pip install obspy   / (LINUX & Mac) -> sudo python install obspy

5.- Install **Thinter**: Thinter is a graphical user interface (GUI). Type the following:

      pip install tk   / (LINUX & Mac) -> sudo python install tk

6.- Install **quantecon**: This library is used for spectrum estimation, Periodogram, Fourier transform. Type the following:

   pip install - - upgrade quantecon   / (LINUX & Mac) -> sudo python install - - upgrade quantecon

7.- Update a library for **matplotlib**. To avoid problems with graphics, install the following:

      pip install msvc-runtime   / (LINUX & Mac) -> sudo python install msvc-runtime

8.- Install **easygui** for the graphical interface:

      pip install easygui   / (LINUX & Mac) -> sudo python install easygui

9 Install **PyWavelets** for CWT handling.

      pip install PyWavelets   / (LINUX & Mac) -> sudo python install PyWavelets

10.- Install **plotly** for handling and assisting with graphics along with Matplotlib.

      pip install plotly   / (LINUX & Mac) -> sudo python install plotly

11.- Install "**pyaudio**", for audio management. Python bindings for PortAudio v19, the cross-platform audio I/O library

      python -m pip install pyaudio / (LINUX & Mac) -> sudo apt-get install python3-pyaudio

Finally, type "pip list" to see the installed libraries. Optionally, you can create a file called "**requirements.txt**" that will contain all the libraries that the PC will use. The "**requirements.txt**" file must be in the current directory. The instruction to do this is as follows:

      pip freeze > requirements.txt