

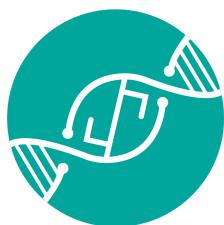


BioPipeline Creator

User Guide

Version 1.0

A user-friendly Java-based GUI for managing and customizing biological data pipelines.



BIOD
Bioinformatics, Omics and
Development



ENGBio
LABORATÓRIO DE ENGENHARIA BIOLÓGICA

SUMMARY

1. Presentation	3
2. Default installation using docker	3
3. Advanced installation using docker	4
4. Manual Installation	6
List of Dependencies and their Installation	6
5. Recommendations for running the software	8
6. Running the BioPipeline Creator Software	8
Running the server side:	9
BioPipeline Creator Client GUI	11
1. Main Window	11
2. The window for Adding and Configuring New Tools (Administrator)	12
3. The window for adding and setting up a new project.	14
Example of How to Create a Project	17
Example of a Pipeline Execution	19
Example of how to add a new tool to your pipeline	22
Curing Parameters	26
Creating your bot for monitoring in Telegram	27

1. Presentation

Welcome to the BioPipeline Creator User Guide - A user-friendly Java-based GUI for managing and customizing biological data pipelines. With this software you can create your bioinformatics tools pipeline and run it in a personalized way.

BioPipeline Creator version 1.0 has already added eight tools and is ready to use. The BioPipeline Creator_Server installation process can be performed in three different ways: Standard, Advanced and Manual. The first two use the Docker infrastructure, and the last method is done manually by the user.

2. Default installation using docker

When installing for standard use, the user follows the five steps described below and then BioPipeline Creator_Server is ready for use.

- System Update

```
sudo apt update  
sudo apt upgrade
```

- Docker installation

```
curl -fsSL https://get.docker.com -o get-docker.sh  
sudo sh get-docker.sh
```

- Add your user to the "docker" group

```
sudo usermod -aG docker $USER
```

- Create a Docker container

```
docker create -it --name pp2oa -p 16002:16002 biodufpa/pp2oa
```

- Start container

```
docker start -i pp2oa
```

3. Advanced installation using docker

With this installation, the user can make some configuration settings directly in the file (config.properties).

The parameters that can be set are: port - defines the default **port** used in the application, the default value is 16002; **maxClients** - defines the maximum number of clients that can connect simultaneously, the default value is 5; **maxParallelTools** - defines the number of tools that can be executed simultaneously by BioPipeline Creator_Server; **db** - defines the default database of the tool, in this case MySQL; **db.url** - specifies the IP address under which the database runs, default value is localhost; **db.user** - specifies the database user, default value is root; and finally **db.password** - specifies the user password in MySQL, default value is root.

Note: To open the file config.properties, the user must use the editor "nano". It is important to note that the database settings do not need to be modified as the database is running within the Docker structure.

- System Update
sudo apt update
sudo apt upgrade
- Docker installation
curl -fsSL https://get.docker.com -o get-docker.sh
sudo sh get-docker.sh
- Add your user to the "docker" group
sudo usermod -aG docker \$USER
- Create a Docker container
docker create -it --name pp2oa -p 16002:16002 biodufpa/pp2oa:advanced
- Start container
docker start -i pp2oa
- Running BioPipeline Creator Server
sudo java -jar Server_PP2OA.jar

After initializing the server, the message below will be displayed for activating the service and sending messages about processing status via Telegram.

Do you want to activate the telegram bot function? (Yes/No):

If the user has chosen to activate the process, he will be asked to enter the ApiBot key, as shown below.

Insert Your Telegram ApiBot Key:

If everything goes right during the initialization process, you will see the following message and can now start the client side.

The server is Listening on Port 16002
Waiting for the clients

Note: See how to generate this key in the "[Creating your bot for monitoring in Telegram](#)" session. It is worth pointing out to the user that the activation of this service is not mandatory and does not interfere with the operation of the pipeline.

4. Manual Installation

List of Dependencies and their Installation

- Several of these dependencies require the user to have superuser powers, so we advise installing them with sudo.
- To run the tool, it is necessary to have the Java Virtual Machine in version 17 or higher installed on the operating system. Download link: <https://www.oracle.com/java/technologies/downloads/>. After installation, you can check that everything is correct by running the test below in your terminal:

```
gislenne@gislenne-moia ~\0] $ java -version
java version "17.0.1" 2021-10-19 LTS
Java(TM) SE Runtime Environment (build 17.0.1+12-LTS-39)
Java HotSpot(TM) 64-Bit Server VM (build 17.0.1+12-LTS-39, mixed mode, sharing)
gislenne@gislenne-moia ~\0] $ █
```

Figure 1. Checking the Java version.

- BUSCO, Megahit, Prokka, and SPAdes installations.

Below is the step-by-step procedure that we used, but if you want you can also consult the manuals directly on the websites of the tools' developers.

a. **BUSCO installation:**

- i. sudo apt update &&
- ii. sudo apt install wget build-essential git default-jre bioperl python3-pip ncbi-blast+ augustus prodigal hmmer r-base --yes &&
- iii. sudo pip install BIO pandas &&
- iv. sudo wget https://mmseqs.com/metaeuk/metaeuk-linux-sse41.tar.gz &&
- v. sudo tar xzvf metaeuk-linux-sse41.tar.gz &&
- vi. echo "export BBMAP=/opt/bbmap" | sudo tee -a /etc/profile &&
- vii. echo "export METAEUK=/opt/metaeuk/bin" | sudo tee -a /etc/profile &&
- viii. echo "export PATH=\$BBMAP:\$METAEUK:\$PATH" | sudo tee -a /etc/profile &&
- ix. source /etc/profile &&
- x. cd /opt &&

- xi. sudo git clone https://github.com/CleoCordeiro/sepp.git &&
- xii. cd sepp &&
- xiii. sudo python3 setup.py config -c &&
- xiv. sudo python3 setup.py install &&
- xv. cd /opt &&
- xvi. sudo git clone https://gitlab.com/ezlab/busco.git busco &&
- xvii. cd busco &&
- xviii. sudo python3 setup.py install &&
- ix. wget https://cloud.r-project.org/bin/linux/ubuntu/marutter_pubkey.asc | sudo gpg --dearmor -o /usr/share/keyrings/r-project.gpg &&
- xx. echo "deb [signed-by=/usr/share/keyrings/r-project.gpg] https://cloud.r-project.org/bin/linux/ubuntu jammy-cran40/" | sudo tee -a /etc/apt/sources.list.d/r-project.list &&
- xi. sudo apt update && sudo apt install --no-install-recommends r-base --yes
- xxii. cd ~

b. Megahit installation:

- i. cd /opt
- ii. sudo wget https://github.com/voutcn/megahit/releases/download/v1.2.9/MEGAHIT-1.2.9-Linux-x86_64-static.tar.gz
- iii. sudo tar zvxf MEGAHIT-1.2.9-Linux-x86_64-static.tar.gz
- iv. sudo mv MEGAHIT-1.2.9-Linux-x86_64-static MEGAHIT
- v. sudo ln -s /usr/bin/python3 /usr/bin/python

c. Prokka installation:

- i. sudo apt update
- ii. sudo apt install libdatetime-perl libncbi6 libxml-simple-perl libdigest-md5-perl git default-jre bioperl --yes
- iii. sudo cpan Bio::Perl Bio::SearchIO::hmmer
- iv. sudo git clone https://github.com/tseemann/prokka.git /opt/prokka
- v. sudo /opt/prokka/bin/prokka --setupdb
- vi. To install tbl2asn: <https://anaconda.org/bioconda/tbl2asn>

d. **SPAdes installation::**

- i. cd /opt
- ii. sudo apt install cmake
- iii. sudo wget http://cab.spbu.ru/files/release3.15.5/SPAdes-3.15.5.tar.gz &&
- iv. sudo tar -xzf SPAdes-3.15.5.tar.gz &&
- v. sudo mv SPAdes-3.15.5 SPAdes &&
- vi. cd /opt/SPAdes &&
- vii. sudo ./spades_compile.sh &&
- viii. sudo ln -s /usr/bin/python3 /usr/bin/python &&
- ix. sudo rm -rf *[!"bin"]*[!"share"]*

Note: The tools must be in the /opt directory. The installation process for these tools has been added to this manual to help the BioPipeline Creator user. However, due to the advances that can occur in the tools. We advise you always to check the tools' official homepage.

5. Recommendations for running the software

It is recommended that the software be run on machines that support the tools already included in the BioPipeline Creator pipeline. Therefore, check the hardware requirements of each tool in their manual.

6. Running the BioPipeline Creator Software

This software is composed of two parts, a server and a client. We recommend that the execution order be server and client, respectively.

- Download the BioPipeline Creator compressed file.
- Place it in the directory of your choice.
- Unzip the BioPipeline Creator package (optionally, if you want to do this in a terminal, use the command unzip file.zip).
- Change the permissions on the software directory, it is advisable to have full permissions:

Example: chmod -R 777 <Directory>

chmod -R 777 /opt/biopipe/server

Running the server side:

- .1. Run the command below in the BioPipeline Creator directory to start the server:

```
java -jar Server_PP2OA.jar
```

- .2. After initializing the server, the message below will be displayed for activating the service for sending messages about processing status via Telegram.

Do you want to activate the telegram bot function? (Yes/No):

- .3. If the user has chosen to activate the process, he will be asked to enter the ApiBot key, as shown below.

Insert Your Telegram ApiBot Key:

- .4. If everything goes right during the initialization process, you will see the following message, and can now start the client side.

Server is Listening on Port 16002
Waiting for the clients

Note_1: See how to generate this key in the "**Creating your bot for monitoring in Telegram**" session. It is worth pointing out to the user that the activation of this service is not mandatory and does not interfere with the operation of the pipeline.

Note_2: If the user opts for a manual installation, the IP of the server must be configured in the bpc_Client config.properties file so that the connection can be established. To do this, carry out the following steps:

1 - sudo ifconfig

When you run this command it will return the network interfaces of your system, look for something like eth0 or as in the case of this example below running on Debian 12, we have:

```
enp2s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 200.239.92.235 netmask 255.255.255.128 broadcast 200.239.92.255
        inet6 fe80::5653:436f:973f:67e3 prefixlen 64 scopeid 0x20<link>
          ether 98:83:89:f8:1d:03 txqueuelen 1000 (Ethernet)
            RX packets 168084 bytes 171077262 (163.1 MiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 79859 bytes 19919454 (18.9 MiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

The IP address comes right after the word `inet`, in this example: 200.239.92.235, just copy this information and put it in the config.properties file into item `serverIp`.

Running the client side:

- .5. Run the command below in the BioPipeline Creator directory to start the client side:

```
java -jar Client_PP2OA.jar
```

BioPipeline Creator Client GUI

1. Main Window

- 1.1. After starting BioPipeline Creator, the main window will be shown (Figure 2). Its items are numbered and described in Table 1.

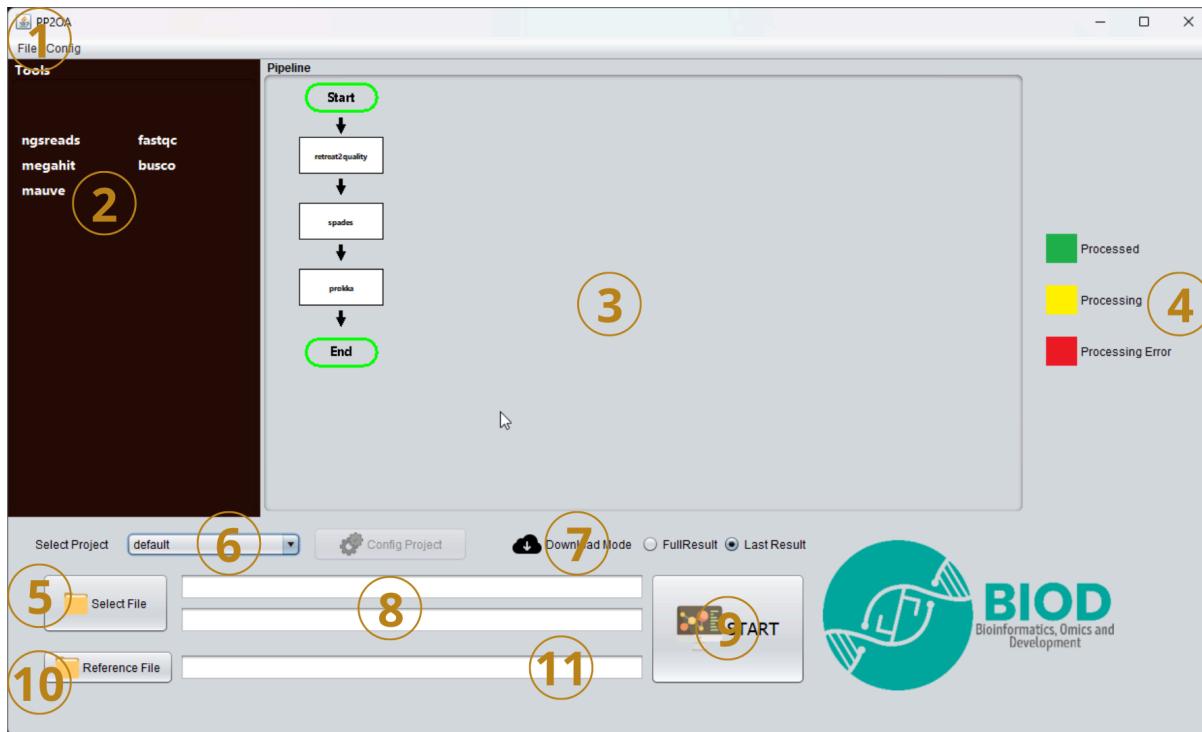


Figure 2. The main window of the BioPipeline Creator software.

Table 1. Description of the items present in the main window of the software.

Items	Description
1	Software configuration (administration password required).
2	The panel of available tools.
3	The panel of tools in the pipeline to be executed.
4	Pipeline execution status legend.
5	Input file selector.
6	Project selector.

7	The selector to download the complete result or just the last tool run.
8	Viewing the files in order.
9	Button to start the pipeline execution.
10	Reference file selector.
11	Adding the reference files.

2. The window for Adding and Configuring New Tools (Administrator)

- 2.1. To add and configure new tools in the software, go to the "config" tab at the top of the main window. After that, a window will appear to log in as administrator (Figure 3).

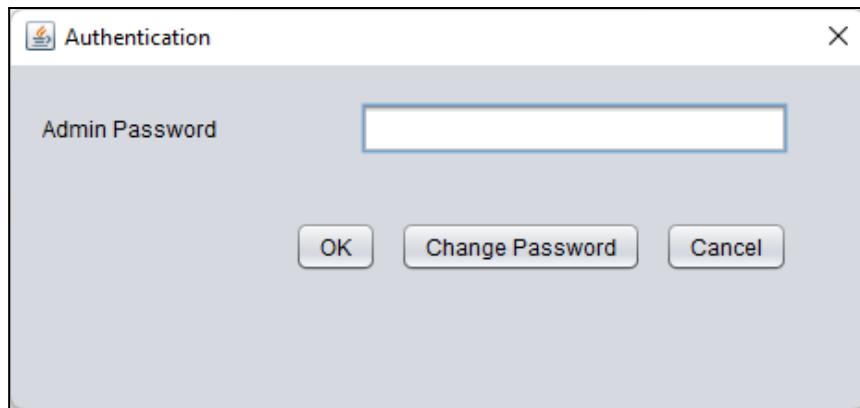


Figure 3. Authentication Window.

Note: The default password is "**admin**". We advise you to change the default password on first access.

- 2.2. After authentication as an administrator, the window for adding and configuring the new tools will be shown (Figure 4). The items are numbered and described in Table 2.

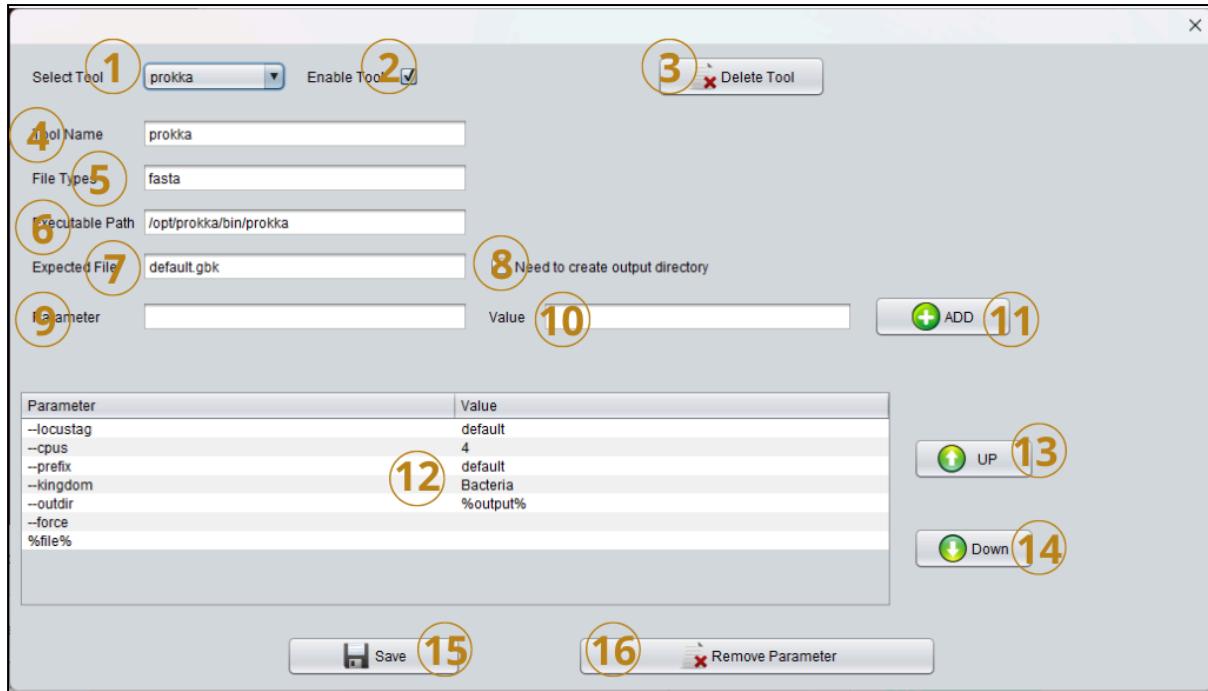


Figure 4. The window for adding and configuring new tools.

Table 2. Description of the items present in the window for adding and configuring tools.

Items	Description
1	Tool selector.
2	Activates or deactivates a tool.
3	Deletes a tool from the database.
4	Enter the tool name.
5	File types are supported in the tool.
6	Path to the tool executable.
7	Expected Result File.
8	Select if the tool does not make its own output directory.
9	Parameter.
10	Parameter value.

11	Button for adding parameter and value.
12	Table with parameters and values.
13	Moves up the selected parameter and value in the table (in case the tool needs an exact order of parameters it can be sorted here).
14	Moves down the parameter.
15	Saves the tool to the database.
16	Removes the selected parameter from the table.

3. The window for adding and setting up a new project.

- 3.1. To add and configure new tools in the software, access the selector named "**Select a project**" at the bottom of the main window and choose the option "**New Project**", after that, a button named "**Config Project**" will appear, clicking it, a window will appear to name the project (Figure 5) and after that, the window for configuring the new project (Figure 6).

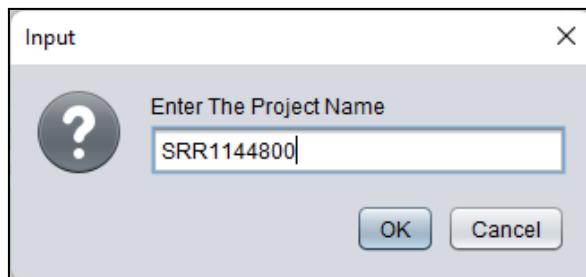


Figure 5. The window for entering the project name.



Figure 6. Project configuration window.

Table 3 lists information about each item present in the configuration window of a project.

Table 3. Description of the items present in the project configuration window.

Items	Description
1	Tool selector.
2	It fills all fields with the default settings of the selected tool.
3	File types are supported in the tool.
4	Project name.
5	Enables telegram bot alerts (if configured on the server)
6	Expected Result File.
7	Parameters
8	Parameter Value

9	Button for adding parameters and value.
10	Table with parameters and values.
11	Moves up the selected parameter and value in the table (in case the tool needs an exact order of parameters it can be sorted here).
12	Moves down the parameter.
13	Saves the tool (to be used in the pipeline that is being built).
14	Removes the selected parameter from the table.

Example of How to Create a Project

In this topic, we will show you step-by-step how to create a new project in the software.

Step 1: In the selector labeled "**Select Project**", choose the option "**New Project**", after that, a button named "**Config Project**" will appear, click on it to create a new project.

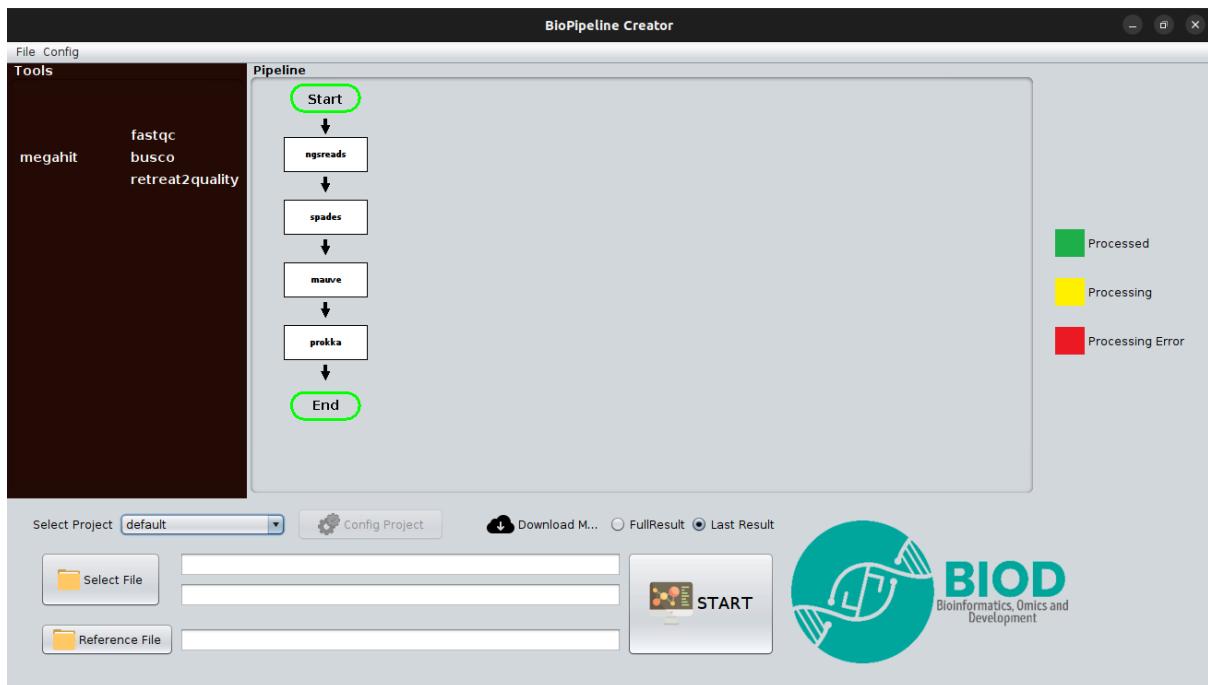


Figure 7. The main window

Step 2: Next, a window will appear to enter the name of the project to be created (Figure 8). Add the name and click on the button named "ok".



Figure 8. Window to enter the project name.

Step 3: Select the tool you want to use from the "**Select Tool**" dropdown, then click the "**Reset to Default Value**" button to restore the default parameters registered in the tool.

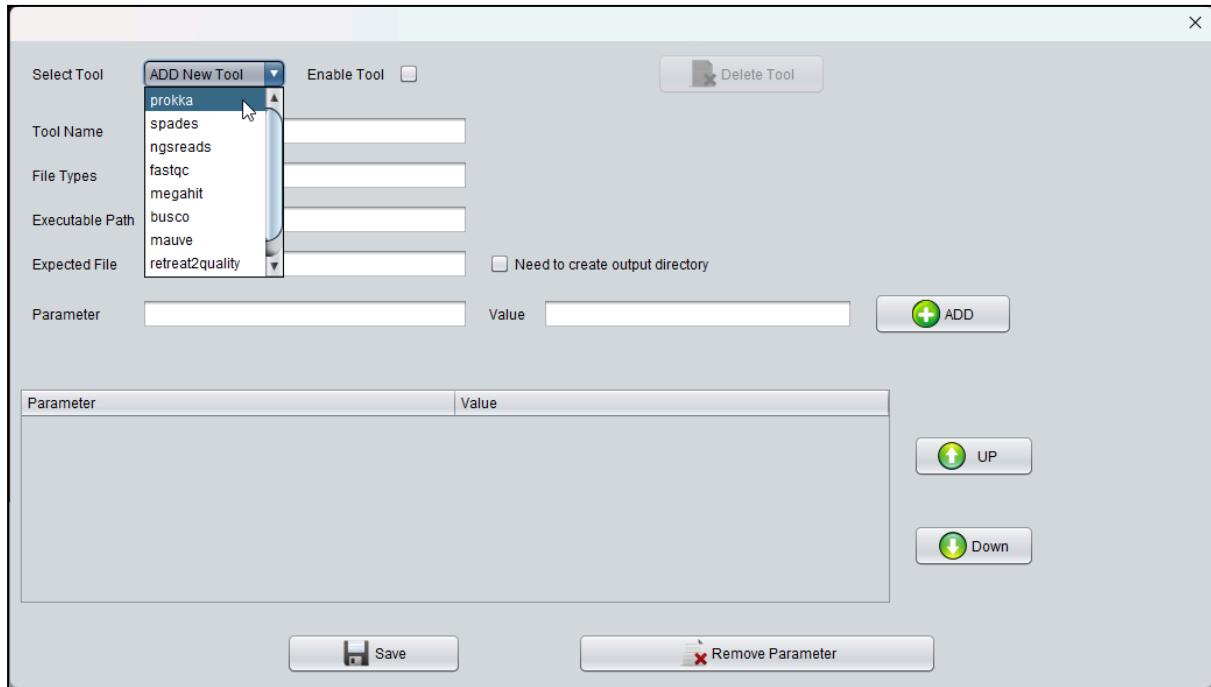


Figure 9. The window for setting up the tools for the pipeline

Step 4: After verifying the filled-out data, click "**Save**". Repeat this for each tool set up in the pipeline.

Note: The process of adding tools to a project can be done as often as needed. The software also allows you to create specific parameters to be used in a certain project. Note that this parameter will only be valid for the project in which it was registered. If you wish to make it the default for other projects, see the section "[Example of how to add a new tool to your pipeline](#)" in the manual.

Example of a Pipeline Execution

In this section, a practical example of running a project in the pipeline will be shown. As example tools, NGSreads1.4, SPAdes, and Prokka will be used and will have as input file, a read in the fastq format.

Step 1: As a first step, it is necessary to add the tools to a given project, figure 10 shows the insertion of the NGSreads1.4 tool, Figure 11 the SPAdes tool, and finally, Figure 12 shows the Prokka tool.

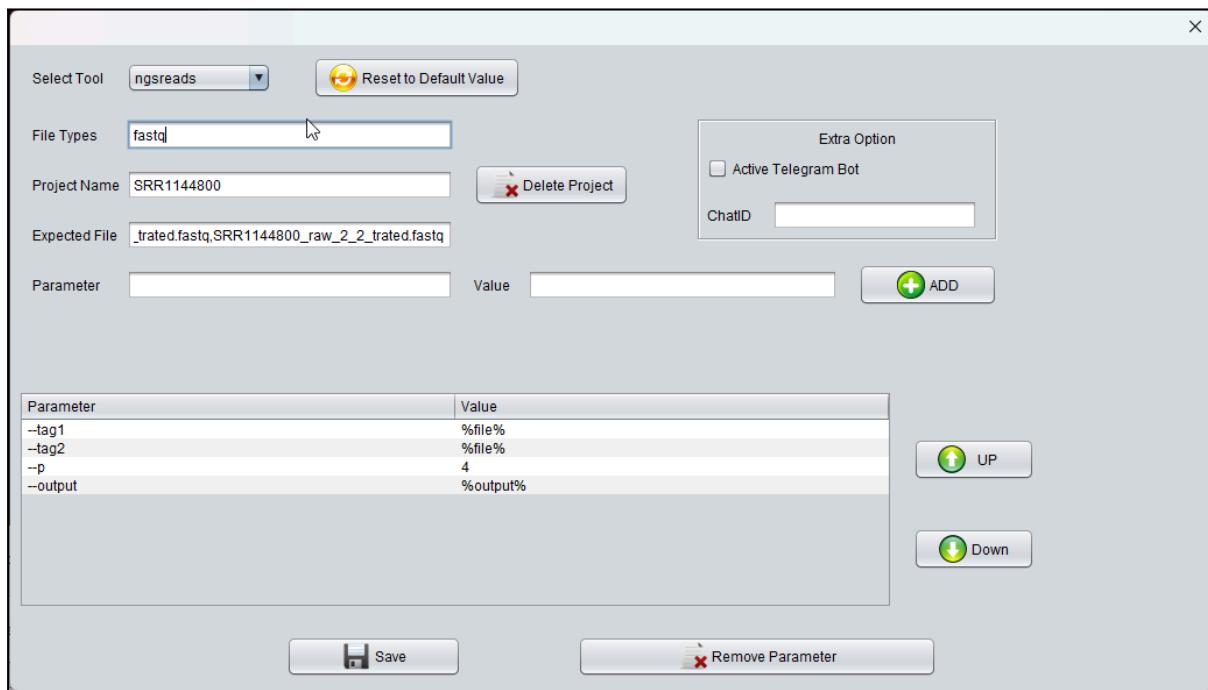


Figure 10. Inserting the NGSreads1.4 tool.

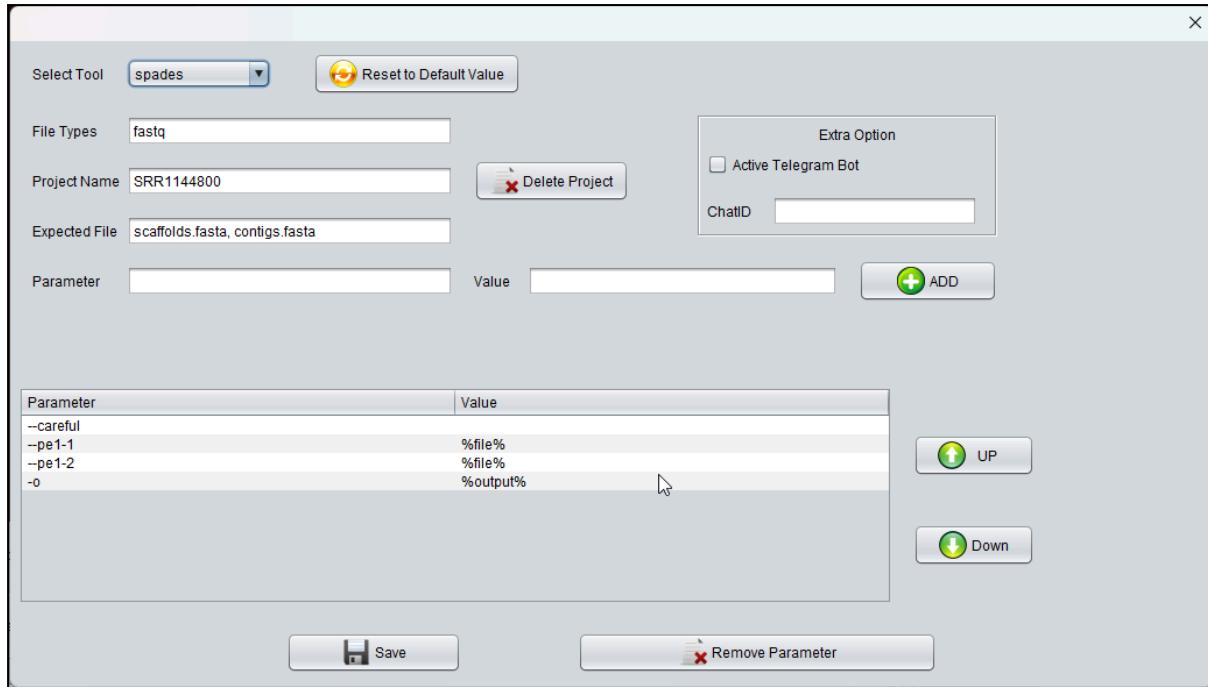


Figure 11. Insertion of the SPAdes tool.

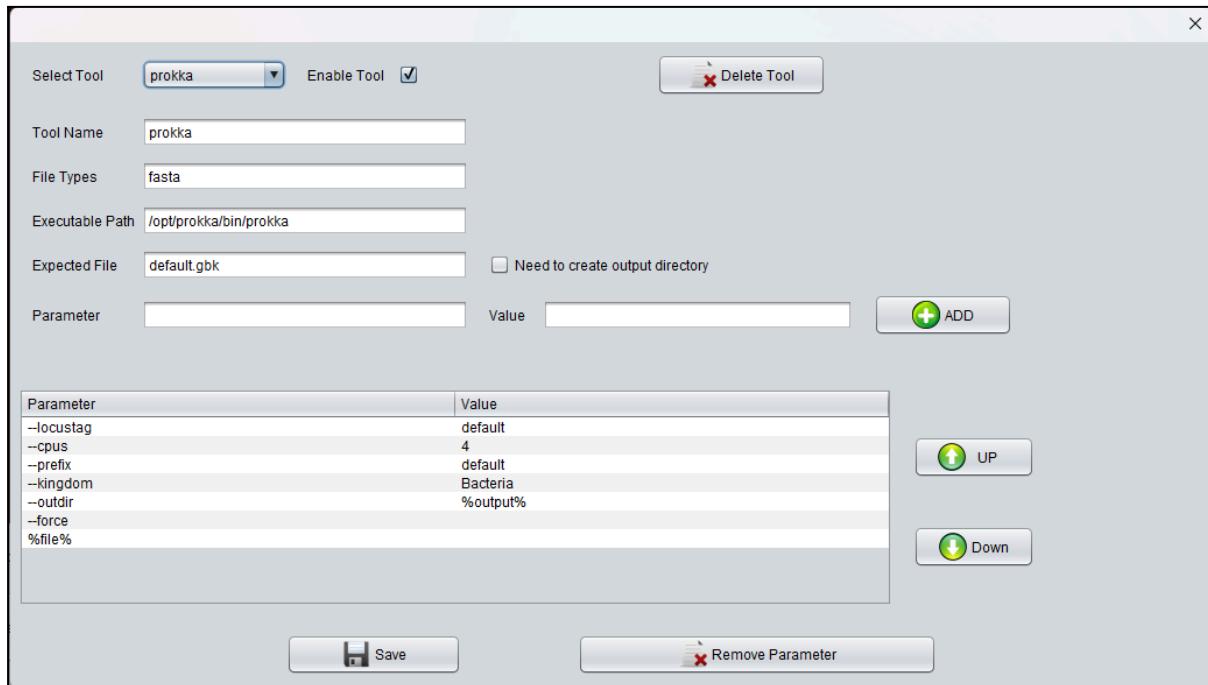


Figure 12. Insertion of the Prokka tool.

Step 2: After inserting the tools, access the Main Window of the software, and through the selector named "**Select Project**", select the project, in the case of the example, the name is "SRR1424625".

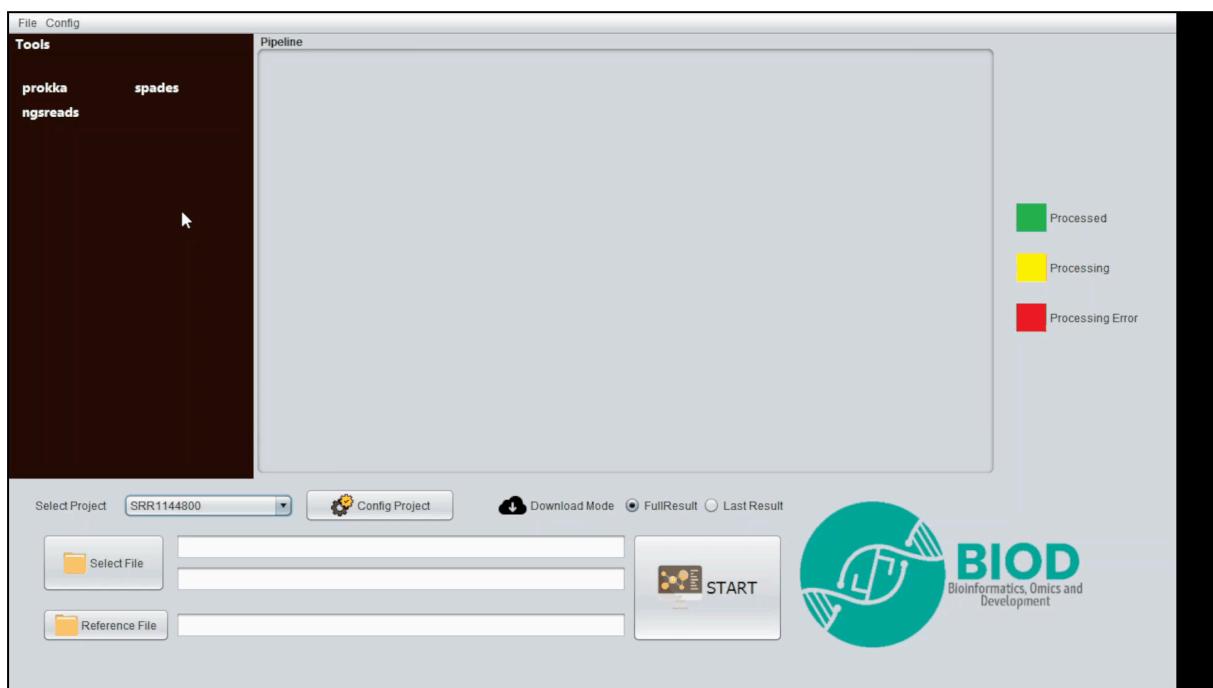
Step 3: Drag the tools to the Drag Tools frame in the desired order and this will form a pipeline.

Step 4: Select the input files via the "**Select File**" button.

Step 5: Choose the download type you wish to receive the files in the "Download Mode" option.

Step 6: Click the "**Start**" button to start the pipeline.

Now just wait for it to process, and at the end of all processing, the result will be downloaded according to the selected mode. Gif 1 shows the step-by-step execution of the pipeline.



Gif 1. Running the pipeline.

Example of how to add a new tool to your pipeline

The following will show the step-by-step process for adding a new tool to the pipeline.

Step 1: Go to the "**config**" tab in the main window of the software (figure 13).



Figure 13. Main Window.

Step 2: Next you will be asked for the administrator password (Figure 14).

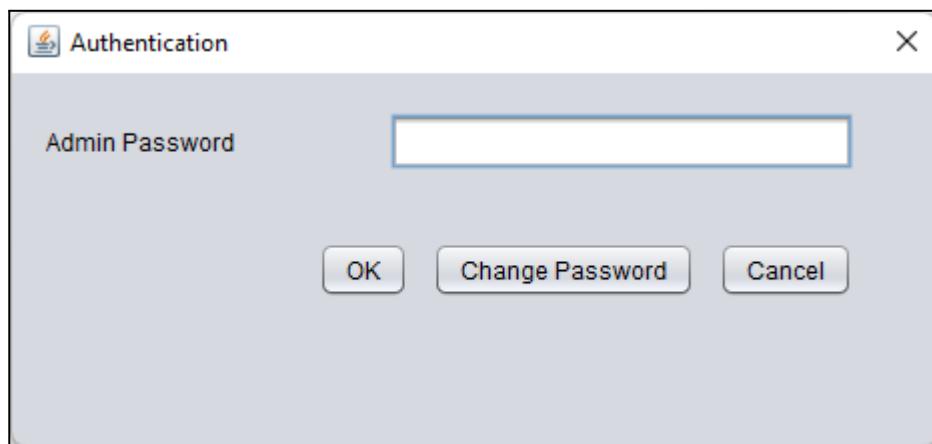


Figure 14. Authentication window for accessing the settings.

Step 3: After performing authentication, the tool configuration window will be shown. Select the "**ADD New Tool**" option included at the bottom of the dropdown at the top (Figure 15).

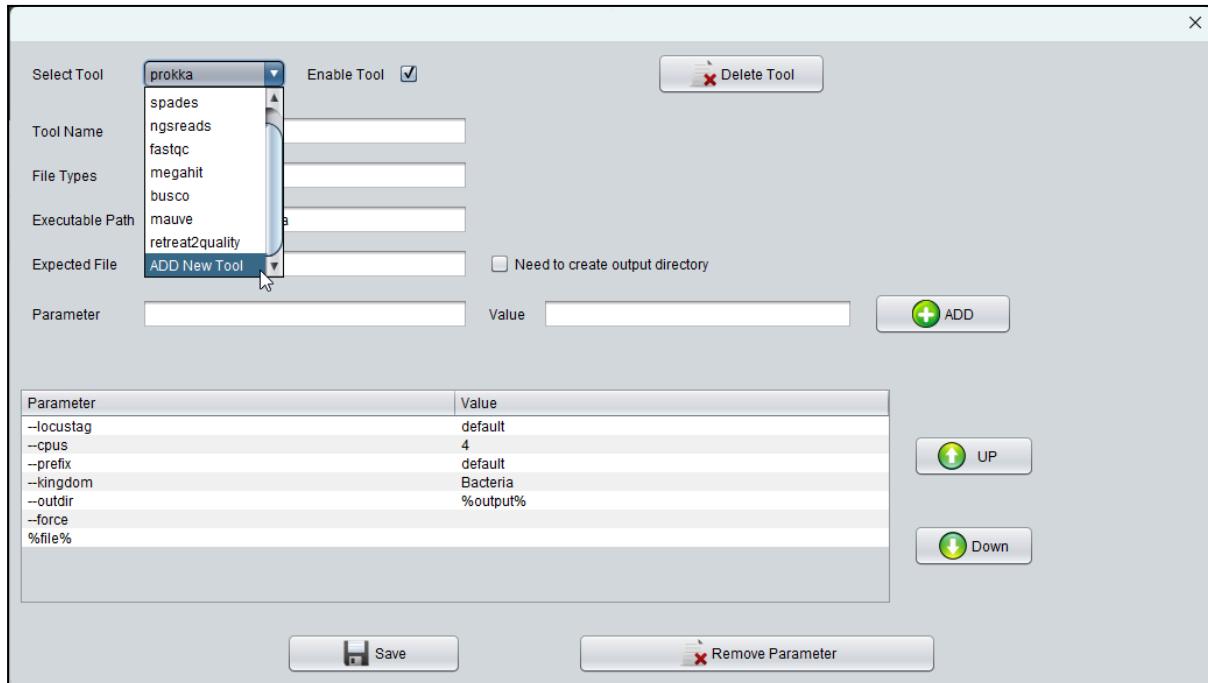


Figure 15. Tool configuration window.

Step 4: When you select the "**ADD New Tool**" option, the window is ready to receive in its fields the data regarding the new tool to be registered in the pipeline (Figure 16).

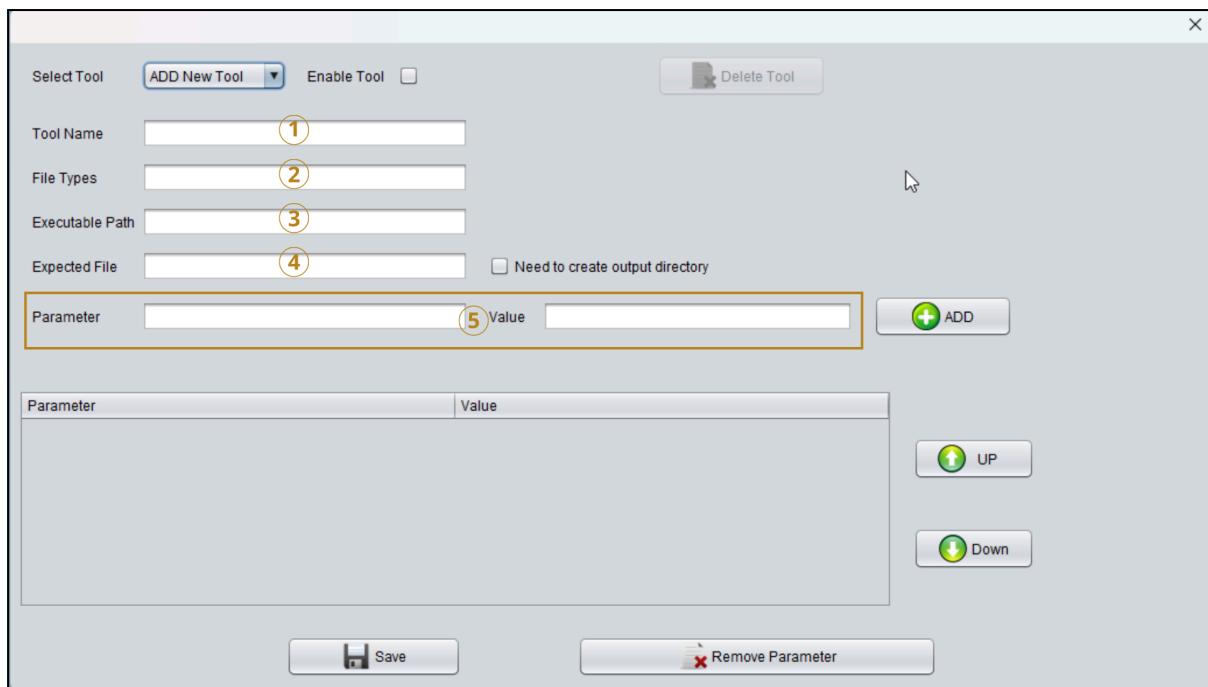


Figure 16. The window for adding the new tool settings.

Step 5: Enter the information about the tool you want to include in your custom pipeline. Table 4 lists the description of each field to be filled in, according to the numbering in Figure 16.

Table 4. Descrição dos itens presentes na janela de configuração da ferramenta.

items	Name	Description
1	Tool name	Define the name of the tool, it is recommended to use the general name of the tool. Example: Prokka.
2	File Types	What is the input file format of this tool, for example, FASTA, FASTQ. If there is more than one input format supported by the tool inform them, separated by commas. Example: fasta, fastq.
3	Executable Path	Enter the installation directory of the tool, it is recommended to install it in the "/opt" directory. Example: /opt/prokka.
4	Expected Files	Which file or files are expected after processing by this tool. Example: default.gbk or MYFILE.fastq. Please refer to the manual of the tool you are installing for this information.
5	Parameter and Value	Enter the list of parameters and their values, if any. Example: --locustag default, parameter --locustag, value default. --force, parameter --force, value empty. For each registered parameter click the ADD button, and repeat this process until you add all the necessary parameters.

Note: All these parameters can be modified, and rearranged when creating the project. These are the default parameters, if you need to create a project you don't need to remember all of them, just use the ones you registered in this step.

Step 6: After the information has been filled in, the add and configure tool window should look like Figure 17. Click the "**Save**" button, and your tool is now available for use.

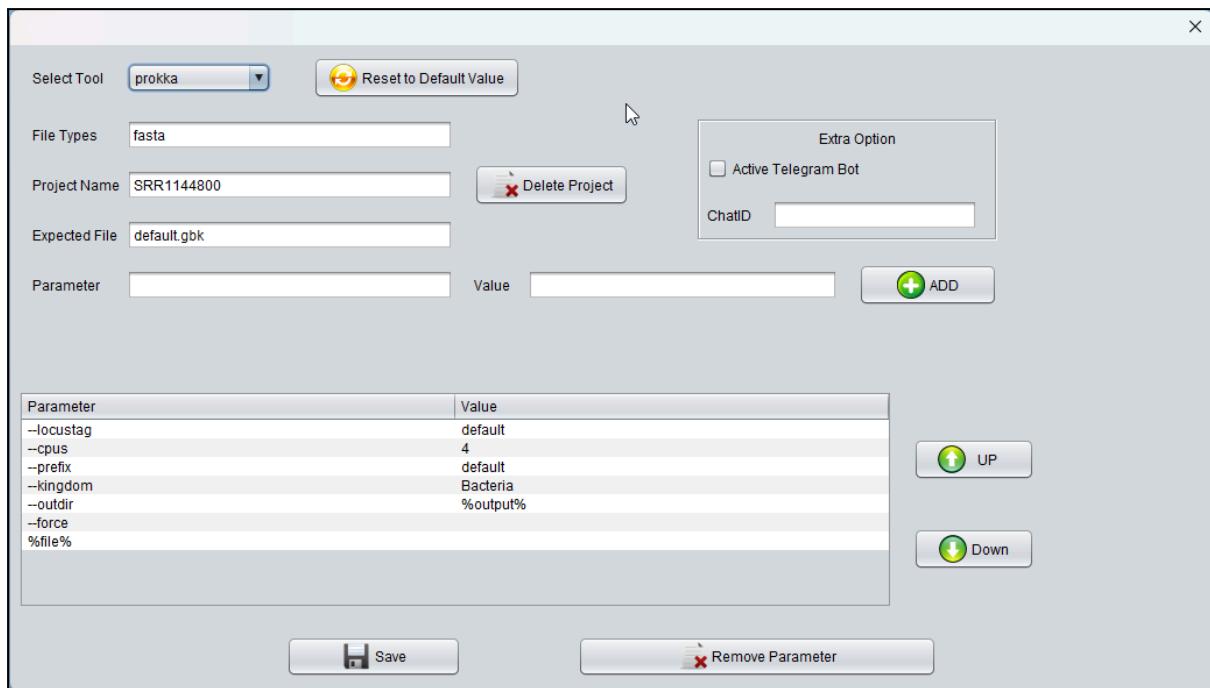


Figure 17. The window for adding the settings of the new tool with the data filled in.

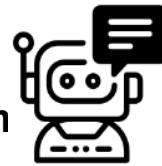
Note: The "**Enable Tool**" checkbox, when activated, makes your tool visible in the tool panel.

Curing Parameters

Table 5 shows some wildcard parameters that are configured in the BioPipeline Creator software.

Table 5. Curing parameters.

Curing	Description
%file%	Used to pass the file that was selected as input
%pwd%	Current execution directory.
%previousOutputDir%	Directory for the output of the previous tool in the pipeline.
%currentOutputDir%	Directory for the output of the currently running tool in the pipeline.
%currentInputDir%	Directory for the input of the currently running tool in the pipeline
%firstInputDir%	Directory for the input of the first tool in the pipeline



Creating your bot for monitoring in Telegram

The software also has a bot to monitor the projects through the Telegram messaging service (<https://web.telegram.org/k/>), to promote greater dynamism in the process of controlling the stages, without the need for the user to follow the processing directly in front of his equipment. The step-by-step configuration of the monitoring bot will be shown below. It should be noted that the use of the Bot is not mandatory.

Step 1: Log into your Telegram and search for the user @BotFather (Figure 18).

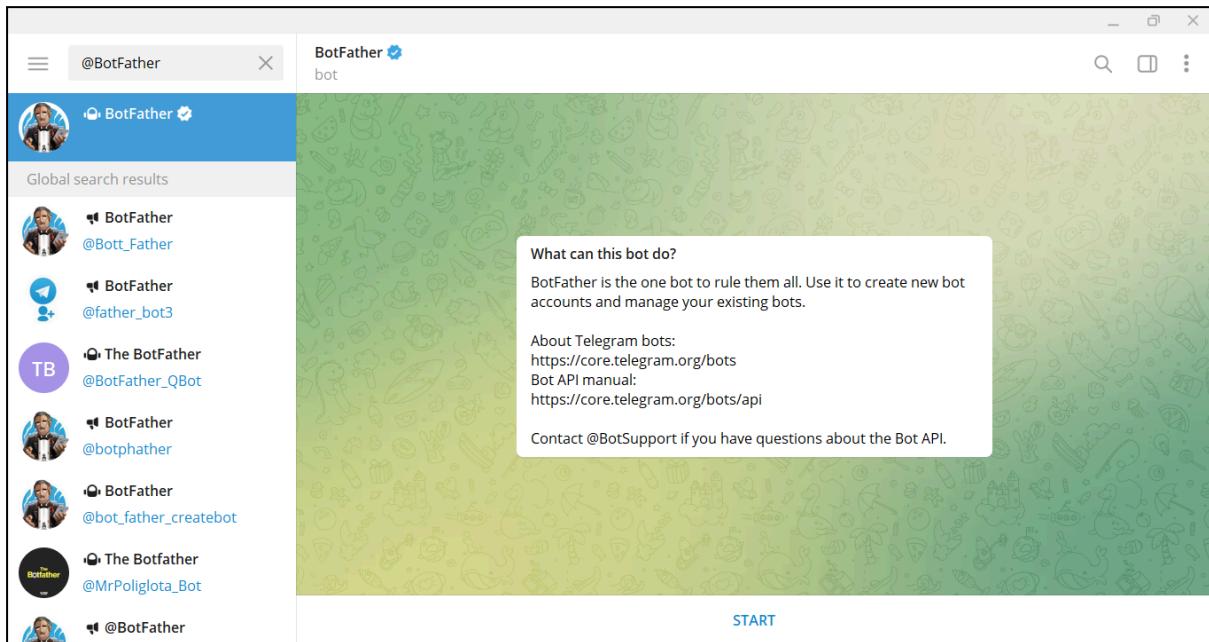


Figure 18. Telegram startup and user search.

Step 2: Click "Start", right after that, a list of settings for creating Telegram bots management will be shown. Click on the "/newbot" option (Figure 19).

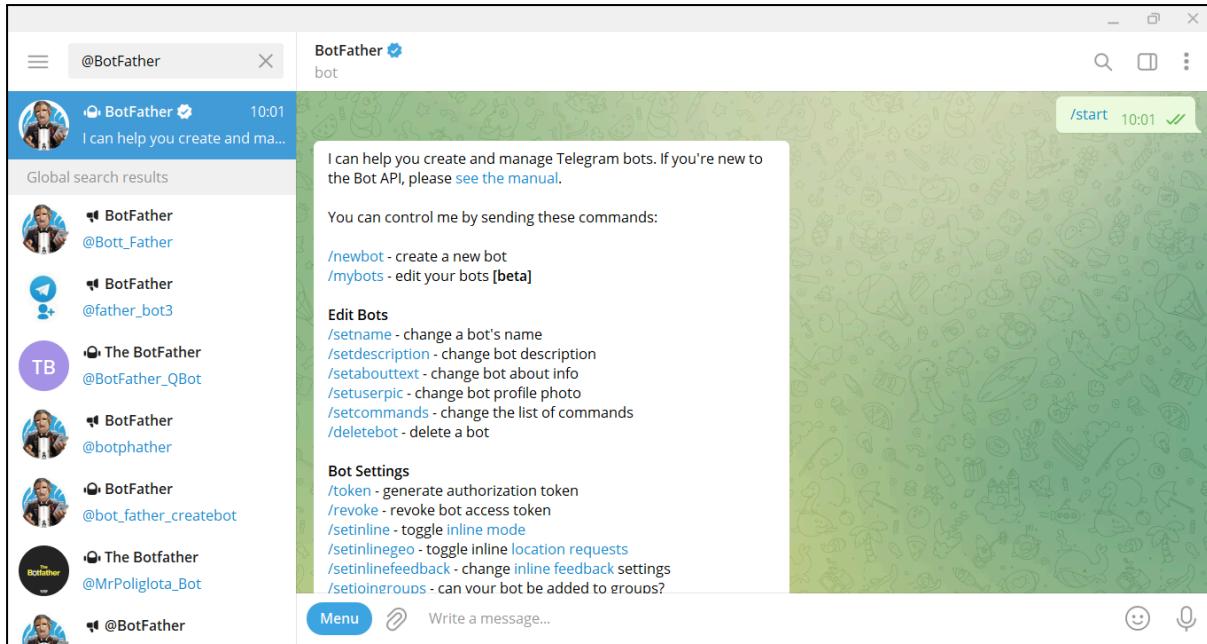


Figure 19. Initializing the bot configuration.

Step 3: Enter a name for your bot, as an example, the name "PPO2A" was used (Figure 20).

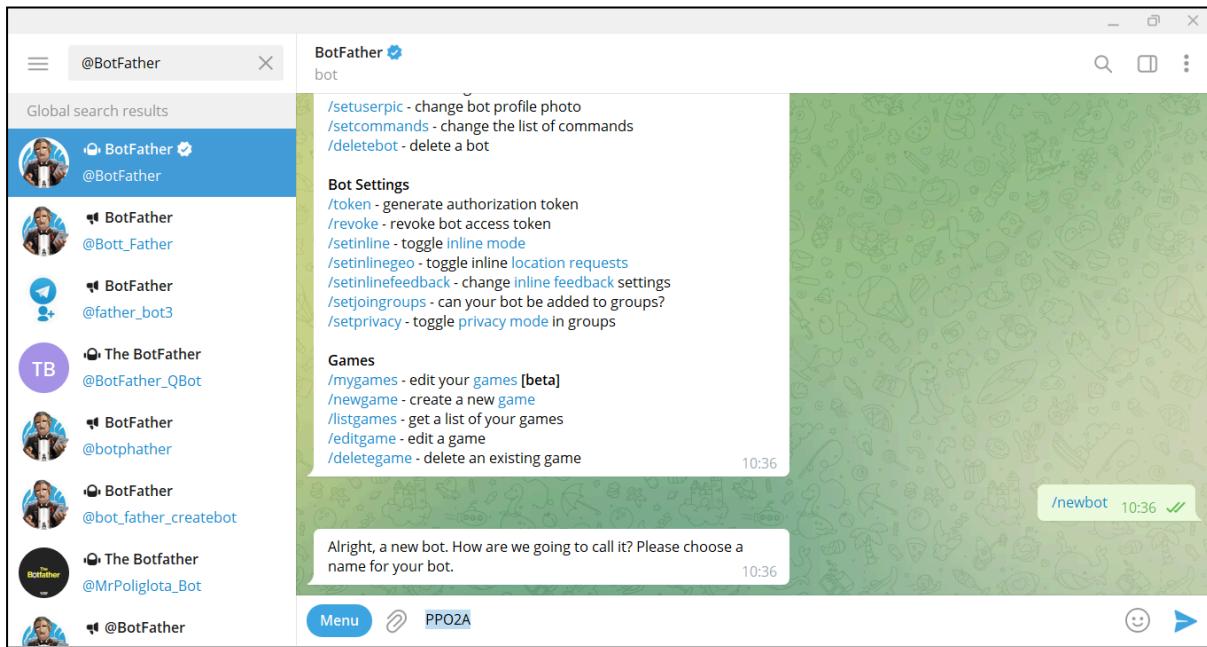


Figure 20. Name for the bot.

Step 4: Next, enter a username. In this step, it is mandatory that the name contains the word "bot" included. Ex: TestBot, Test_bot or PPO2A_bot.

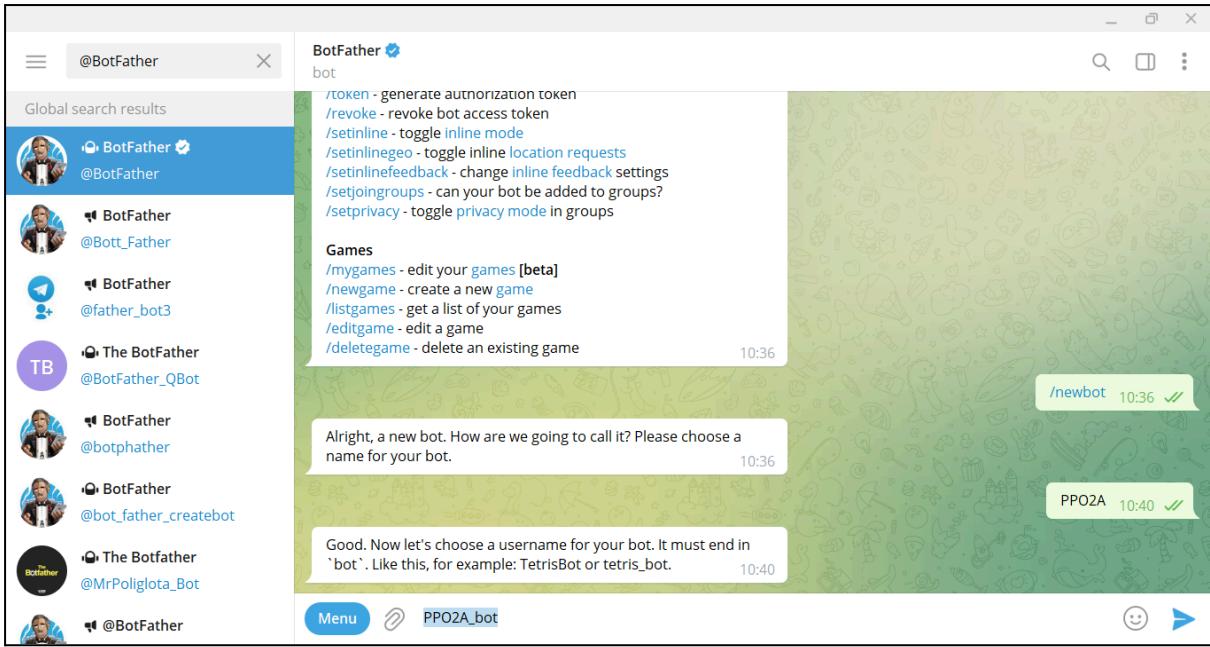


Figure 21. Username for the Bot.

Step 5: If there is no bot registered with the same username as yours, the token will be informed for validation (Figure 22).

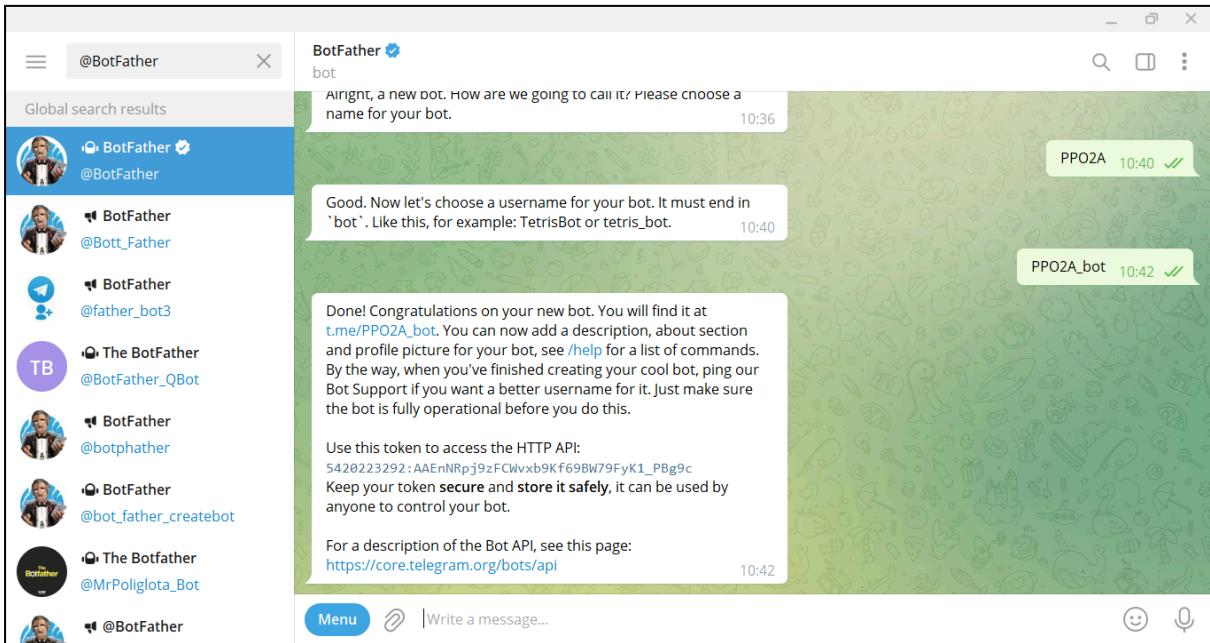


Figure 22. Configuration token.

Step 6: Copy the access token and add it in step 1.2 of the server initialization (Figure 23).

```
Do you want to activate the telegram bot function? (Yes/No): y
Insert Your Telegram ApiBot Key: 5420223292:AAEnNRpj9zFCWvxb9Kf69BW79FyK1_PBg9
```

Figure 23. Adding the access token.

Step 7: After adding the access token and initializing the server, send a message to your created bot, and will return the chatID to add to the project. To access your bot, just click on the link shown in Figure 22, as in the example: t.me/PPO2A_bot, or search by the name of your bot, and click START.

Note: The server must be turned on for the bot to send the `chat_id`.

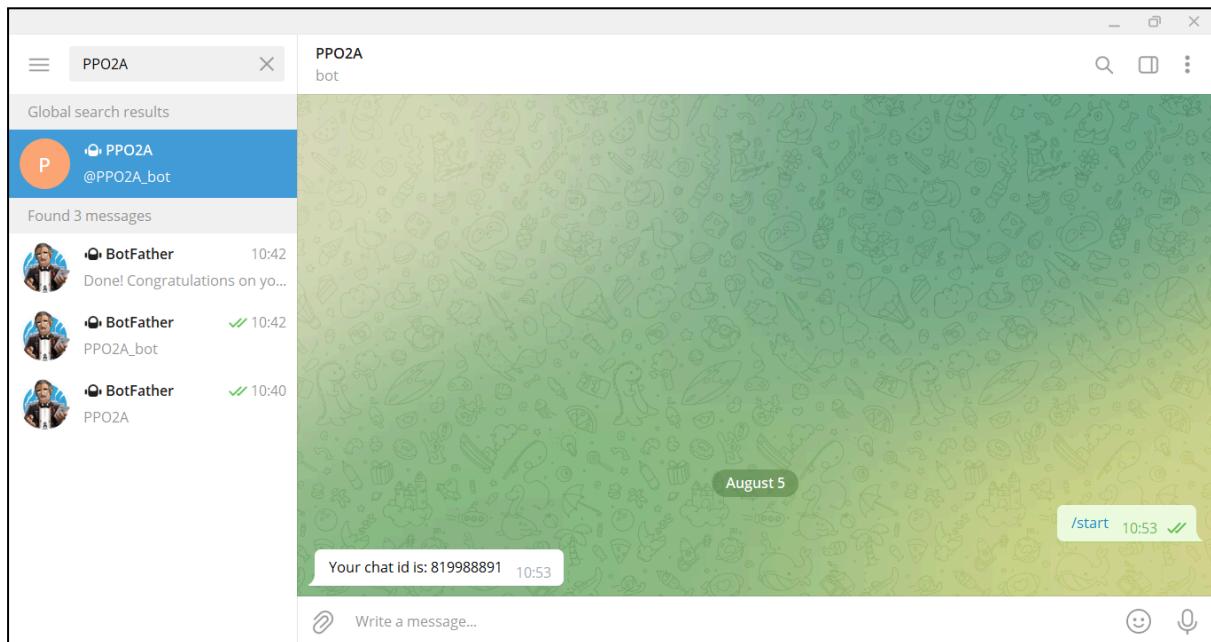


Figure 24. Getting chatID.

Step 8: Copy the chatID, go to a project's configuration screen, enable the "Active Telegram Bot" option, and then paste the identifier into the chatID field (Figure 25).

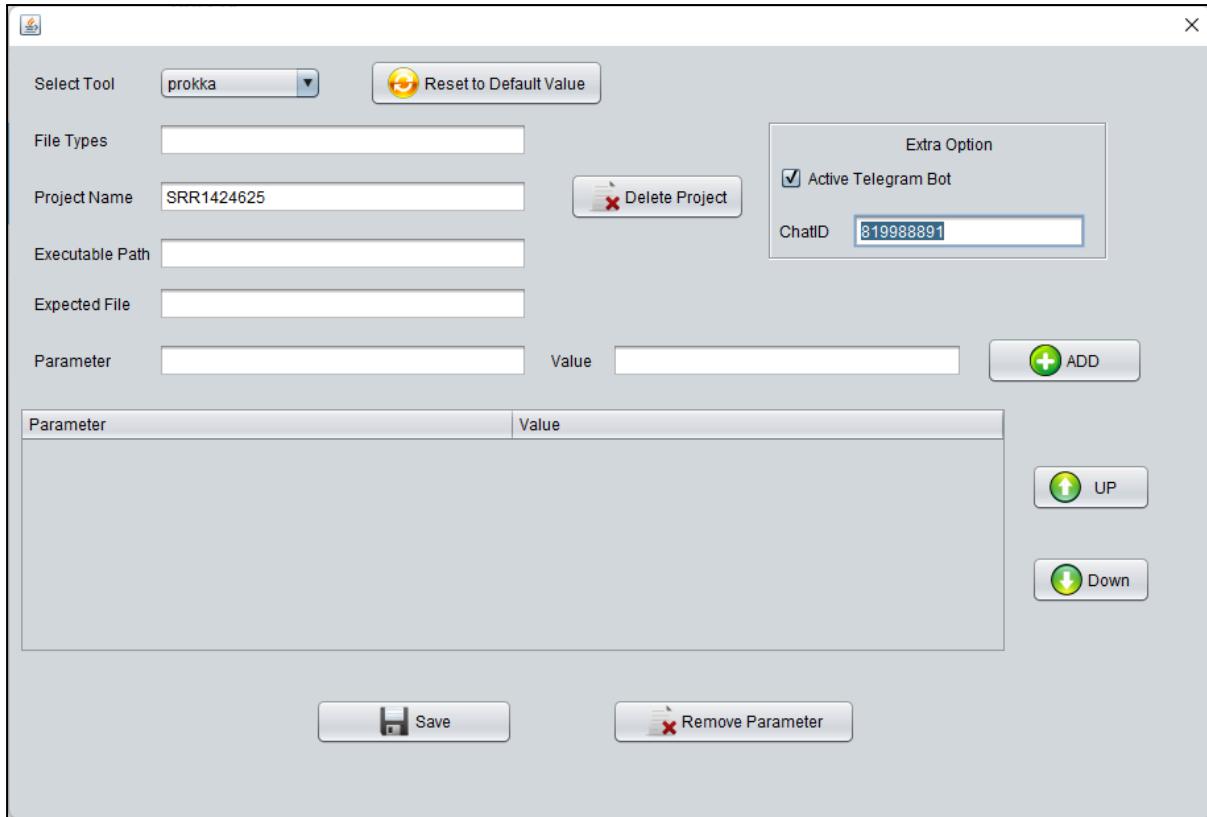
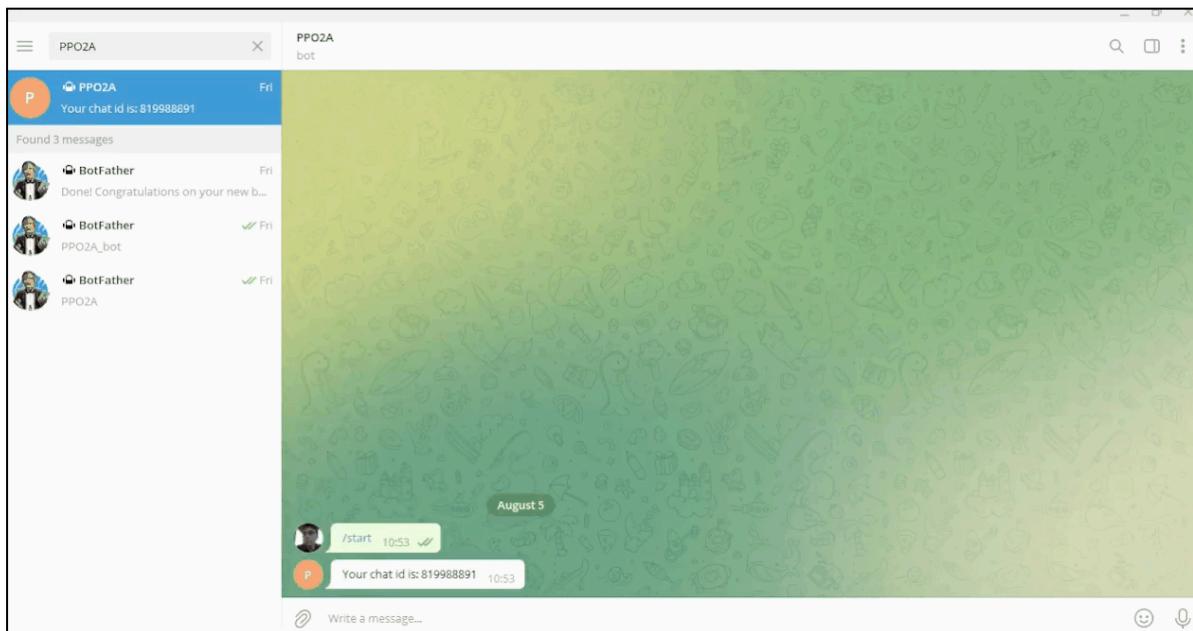
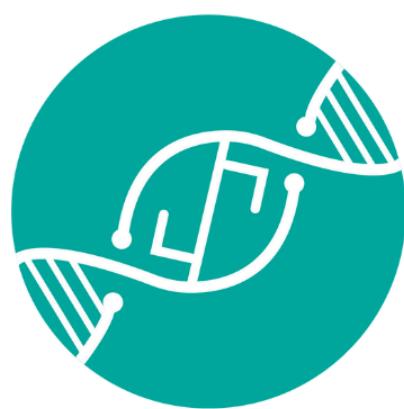


Figure 24. Added chatID identifier to the project.

Step 9: After these steps, your bot is already configured. It will send a message to your Telegram chat whenever the project changes. The sample gif below demonstrates the alerts generated during the execution of a project.



Gif 2. Project alerts.



BIOD
Bioinformatics, Omics and
Development

