Data collection

1) Create empty spectrum files in SpectrumsManager page of web app

1.1) In the bottom of the page, specify filename, multi inlet valve position from 1 and 16, initial M, step and amount of steps

1.2) Press "Create Spectrum" button

2) Create a task in Task Manager page of web app

2.1) In the bottom of the page, select type of the task (emergency, scheduled or regular)
2.2) Specify unique task name, multi inlet valve position from 1 to 16, filename of spectrum and amount of scans
2.3) If task type is emergency, specify the amount of executions – either integer or "inf". If task is scheduled, specify the frequency of executions (for example, if freq=3600, task would be executed once every 3600 seconds)

Priority of task execution is determined the following way:
1) If there is an emergency tasks the in list, first emergency task encountered is executed
2) Else, if there are scheduled tasks in the list, first scheduled task that wasn't executed on time, would be executed (for example, task with freq=3600 that was executed 3800 seconds ago)
3) Else, regular tasks are executed in alternating manner (for example, task1, then task2, then task3, than task1 if there are 3 tasks in the list)

3) To start data collection, execute StartSampling.py script

3.1) Navigate to Urania directory:

```
cd Desktop/Urania-Main
```

3.2) Execute script with sudo privileges:

```
sudo python3 StartSampling.py
```

3.3) Enter admin password

Data visualisation

1) To open data visualisation webapp, start streamlit localhost server

1.1) Navigate to Urania directory:

```
cd Desktop/Urania-Main
```

1.2) Execute script that starts localhost server:

```
streamlit run main.py
```

1.3) Navigate to localhost:8501 address through web browser (usually done automatically)

2) To open data visualisation page, select the name of desired page on left sidebar

2.1) Data visualisation works either in “last” or “search” mode of operation. It can be changed in Settings menu. If the mode of operation is “last”, program will automatically display the most recent scans. If the mode of operation is “search”, you should select the date and time through prompted widget. Program will display data for this moment of time

2.2) Select how much spectrums you want to display on screen by typing an integer into the field

2.3) There are two types of graphs displayed below: spectrum for given moment of time and concentration vs time graphs for given molar masses

2.4) Each type of graph can be displayed with values in pascals (raw output of RGA) or in PPM (automatically converted). Each type of graph can be displayed either on linear or logarithmic scale.

2.5) For concentration vs time graph, you can input molar masses (or “ox” or oxygen) splitted by comma into the field

2.6) To display table with values, press “display table with values” button. It can be copied and pasted to office programm such as LibreOffice or Excel

2.7) To compare results with tolerance values (file with tolerances can be selected in Settings), press “Find Abnormalities” button

Remote Access

1) System can be accessed remotely through RustDesk program

1.1) Install RustDesk on your system. On Debian systems it can be done by typing “sudo apt install rustdesk”

1.2) Enter 9-digit ID and password of system (not provided in this manual for security reasons)

1.3) Set up 2FA through any 2FA application (I recommend Aegis, as it is free and open source and 100% offline). Credentials are not provided in this manual

1.4) Click “Remote connect” to control the system or “file transfer” to use system as network attached storage.

Manual control of hardware

1) VSC and multi inlet valve can be controlled using VSC Page page in web app

2) RGA can be manually controlled through RGA Controls page. If RGA IP address can’t be found, navigate to Technical page and press “Netdiscover for RGA IP” button