



Clab: Innovación en procesos de fabricación circular y abierta para entornos educativos

N.º Expediente: AEI-010500-2022b-177

PT2-E4: Kit de datos

Índice

Índice.....	2
Introducción.....	3
Tutoriales.....	4
Arduino IDE.....	5
Gráficos de sensores en tiempo real.....	5
Modificar parámetros del Smart Citizen Kit.....	14
Herramientas web.....	19
Acceso a los datos a través de la plataforma web.....	19
Acceso a los datos a través del dashboard.....	21
Orange Data Mining.....	24
Configuración de Orange Data Mining.....	24
Acceso a datos de Smart Citizen a través de Orange.....	29
Representación de datos geolocalizados con Orange.....	41
Processing.....	49
Configuración de Processing para su uso con el Smart Citizen Kit.....	49
Animación de Processing con datos en tiempo real.....	52
Recursos adicionales.....	56
Configuración del kit.....	56
Datos.....	56
Otras referencias de educación.....	56
Proyectos externos.....	57

Introducción

En este entregable se incluyen los materiales desarrollados para dar soporte a diversas actividades en el aula en torno al uso de sensores ambientales educativos. Estos son materiales para el uso del Smart Citizen Kit con diferentes objetivos. La descripción desde un punto de vista pedagógico del proyecto, así como su relación con las competencias, saberes y asignaturas regladas en la ley de educación vigente en el momento de escritura de este entregable (Ley Orgánica 3/2020, de 29 de diciembre, LOMLOE), se encuentran listados en el entregable PT2-E6, siendo el objetivo del presente entregable consolidar dichos materiales en un único documento.

El objetivo principal de este documento es el de proveer a los usuarios de las diferentes herramientas de datos descritas en PT2-E3 de manuales de uso, guías y tutoriales necesarios para su utilización. Para facilitar la planificación del docente, cada uno de los materiales descritos incluye un tiempo estimado de preparación, así como los requisitos de materiales o recursos informáticos para su implementación. Finalmente y, en la medida de lo posible, se incluyen descripciones detalladas de cada paso a seguir para que el lector pueda seguir los recursos de manera independiente.

Estos se complementan con documentación adicional (en inglés) en docs.smartcitizen.me.

Tutoriales

En esta sección se detallan los diferentes tutoriales, de forma secuencial y organizados por herramienta, que dan soporte a la utilización de las diferentes soluciones técnicas propuestas en PT2-E3 y PT2-E5. Las categorías se listan a continuación:

- *Arduino IDE*: materiales en torno a la utilización del *Arduino IDE* con el Smart Citizen Kit para la interacción con los datos en tiempo real, o la configuración avanzada del dispositivo.
 - La visualización de datos en tiempo real permite la experimentación directa con los sensores, ayudando a su explicación.
- *Herramientas Web*: materiales entorno a la utilización de las herramientas web (plataforma, dashboard web) que permiten un análisis básico de los datos, o su acceso de forma programática
 - Acceder a través de la herramienta web permite un análisis básico de cualquier dispositivo de forma rápida y sencilla sin necesidad de tener un kit en particular.
- *Orange Data Mining*: materiales de guía para la instalación y utilización de los módulos de programación visual desarrollados en el entorno de *Orange Data Mining*¹.
 - Acceder a los datos a través de *Orange Data Mining* permite un análisis profundo de datos tomados en algún experimento concreto.
- *Processing*: materiales de soporte para la creación de aplicaciones en el entorno de *Processing*² en el contexto de programación creativa
 - Usar *Processing* permite construir visualizaciones de datos de manera creativa, por ejemplo a través de animaciones.

¹ Orange Data Mining es un proyecto abierto que proporciona herramientas de programación visual. La página oficial se encuentra en <https://orangedatamining.com>

² Processing es un proyecto abierto de programación creativa. La página oficial se encuentra en <http://processing.org>

Arduino IDE

Representación de datos en tiempo real

Introducción

En este tutorial se muestra cómo representar datos en tiempo real de un Smart Citizen Kit (SCK) para la realización de demostraciones, en un nivel básico, del funcionamiento de los mismos.

Este tutorial se puede encontrar en inglés en la documentación oficial³.

Requisitos

- **Software:** los materiales indicados a continuación requieren de la instalación previa del *Arduino IDE*, el cual se puede encontrar en la web del proyecto⁴. Esta instalación es posible en varias plataformas (*Windows, Mac y Linux*) y tiene una duración estimada de 5 minutos.
- **Materiales:** se necesita un ordenador con puerto *USB*, un cable *USB a USB-micro* y un Smart Citizen Kit.

Tiempo estimado

15 minutos

Descripción detallada

En esta sección, se detallarán los pasos a seguir en este tutorial.

1. En primer lugar, conecta el cable USB-micro al Smart Citizen Kit y al ordenador. En la *Figura 1* se muestra la ubicación del puerto USB.

³ Documentación oficial: [https://docs.smartcitizen.me/Resources/Tutorials/Arduino Serial/](https://docs.smartcitizen.me/Resources/Tutorials/Arduino%20Serial/)

⁴ Web del proyecto Arduino: <https://www.arduino.cc/en/software>



Figura 1. Ubicación del puerto USB micro en el Smart Citizen Kit

En segundo lugar, en el ordenador, abre el software *Arduino IDE* (Figura 2). Una vez conectado el SCK al ordenador, en el *Arduino IDE*, se debe seleccionar el puerto USB que debe usar para la comunicación con el mismo. Para ello, accede a la selección de puertos a través del menú *Tools > Port* (o *Herramientas > Puertos*). En el menú desplegable, se debe escoger aquel marcado como (Arduino Zero (Native USB Port)) o (Arduino/Genuino Zero (Autodetect Port)). Este menú se muestra en la *Figura 2* y su apariencia puede variar en función de la plataforma utilizada⁵.

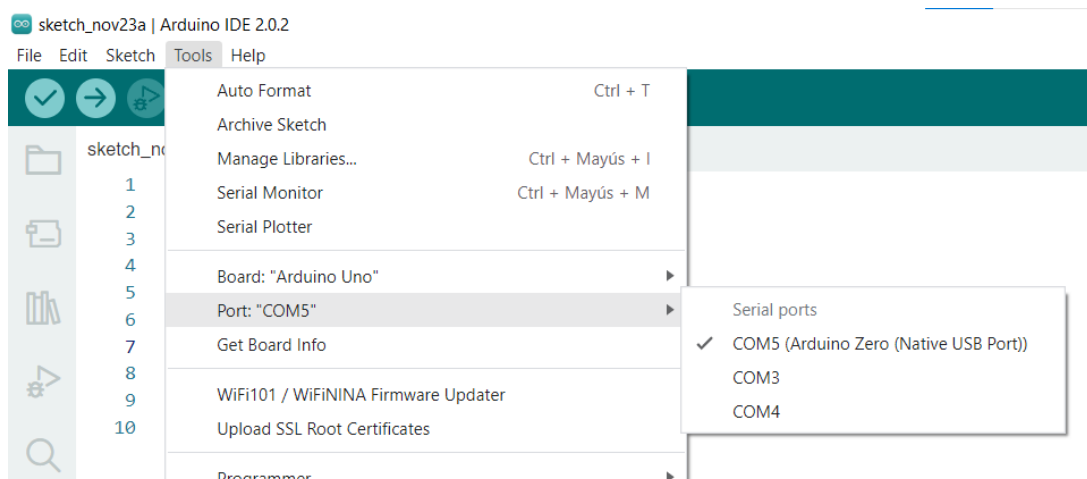


Figura 2. Selección del puerto en Arduino IDE

2. El *Serial Plotter* (*Monitor Serie* en castellano) es una herramienta que permite visualizar datos en tiempo real generados por un dispositivo conectado al puerto USB del ordenador (también llamado *Serial* en algunos tutoriales). El acceso al Serial

⁵ Las figuras que muestran el Arduino IDE han sido creadas con la versión Arduino IDE 2.0, en un ordenador con plataforma Windows. Aunque otras versiones y plataformas pueden variar en la disposición de los menús y elementos, la funcionalidad es la misma.

Plotter se realiza a través del menú *Tools > Serial Plotter* (o *Herramientas -> Serial Plotter*).

Nota:

En el caso de que aparezca el mensaje: *Couldn't open serial plotter* (Figura 3), revisa si: (1) el cable *USB* está correctamente conectado, y que es defectuoso, (2) si el *SCK* está encendido (el *LED* indicador está encendido), o (3) si el puerto está correctamente seleccionado en el *Arduino IDE*. A continuación se detallan posibles soluciones a estos problemas.

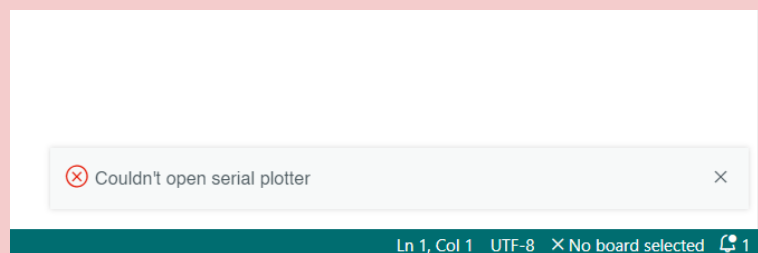


Figura 3. Mensaje de error: "Couldn't open serial plotter"

- Comprueba que el cable no sea defectuoso simplemente utilizando otro cable. En ocasiones, es posible que el puerto *USB* del ordenador también esté deshabilitado.
- Si el cable está conectado pero el dispositivo no aparece listado en el *Arduino IDE* (i.e. en el listado de puertos no aparece (*Arduino Zero (Native USB Port)*)) se puede deber a dos posibles motivos: o bien que hay un problema con el *SCK* (puede simplemente estar apagado), o bien puede que no esté bien instalado el software.

Para encender el dispositivo, pulsa el botón de *ON/OFF* del *SCK* (Figura 4).



Figura 4. Botón ON/OFF

- c. Prueba a seleccionar el puerto en el *Arduino IDE*. En *Arduino 2.x* se puede seleccionar directamente el puerto desde un selector en la parte superior (Figura 5).

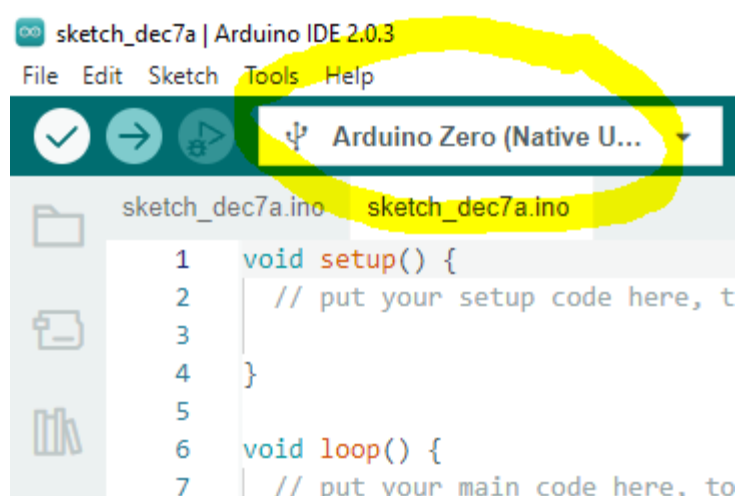


Figura 5. Selección del puerto serial en interfaz de *Arduino IDE*

Si se hace uso de este menú desplegable, es posible que el programa requiera instalar una placa adicional (se indicará como *Smart Citizen*) en la parte inferior del programa. Adicionalmente, en algunas plataformas, por ejemplo en *Windows* es posible que se requieran permisos adicionales para la instalación (es importante realizar estos pasos con *Permisos de Administrador*). Los mensajes del terminal se pueden ver en la Figura 6.

Figura 6. Instalación de la placa de Smart Citizen Kit

Una vez funcional y abierto el *Serial Plotter*, aparecerá una ventana similar a la de la Figura 7.

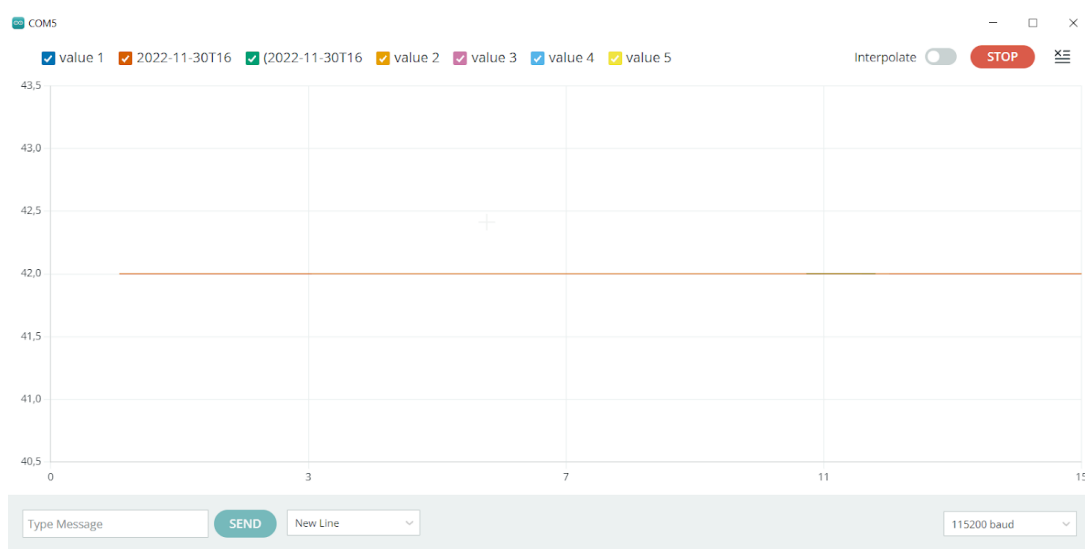


Figura 7. Serial plotter

Más información sobre el plotter serial en los tutoriales de *Arduino*:

- <https://docs.arduino.cc/software/ide-v2/tutorials/ide-v2-serial-plotter> [EN]
- <https://arduino.cl/como-usar-el-serial-plotter-en-arduino-ide/> [ES]

3. En la *caja* donde indica **Type Message** se pueden enviar mensajes al *SCK*. Para saber más sobre comandos disponibles hay documentación sobre cómo se usa la línea de comandos visita la siguiente guía:

- https://docs.smartcitizen.me/Guides/getting_started/Using_the_Shell/#connectivity-or-configuration-issues

Adicionalmente, existe una serie de *webinars* sobre la línea de comandos (*Shell*) disponibles en *youtube*:

- <https://www.youtube.com/watch?v=ZNStSudpg0A&list=PL33KKs9g8Y1IWstZmDc-46yFuulRZEmi&index=5>

4. Una vez configurado el *Monitor Serial*, se pueden introducir comandos para que sea interpretado por el *SCK*. Aquí se presentan una serie de comandos para visualizar diferentes sensores de un *SCK* genérico. El *stream* de datos no tendrá asociado un *timestamp* (fecha y hora), y el valor de interés se suele representar en *value2* en el *Serial Plotter*. A continuación se detallan diversos comandos disponibles, sus efectos y una tabla al final resumen.

Micrófono

Enviando este comando:

```
monitor noise -notime -noms
```

Se obtendrá una salida similar a la de la *Figura 8*. Los datos están en *escala dBA*.

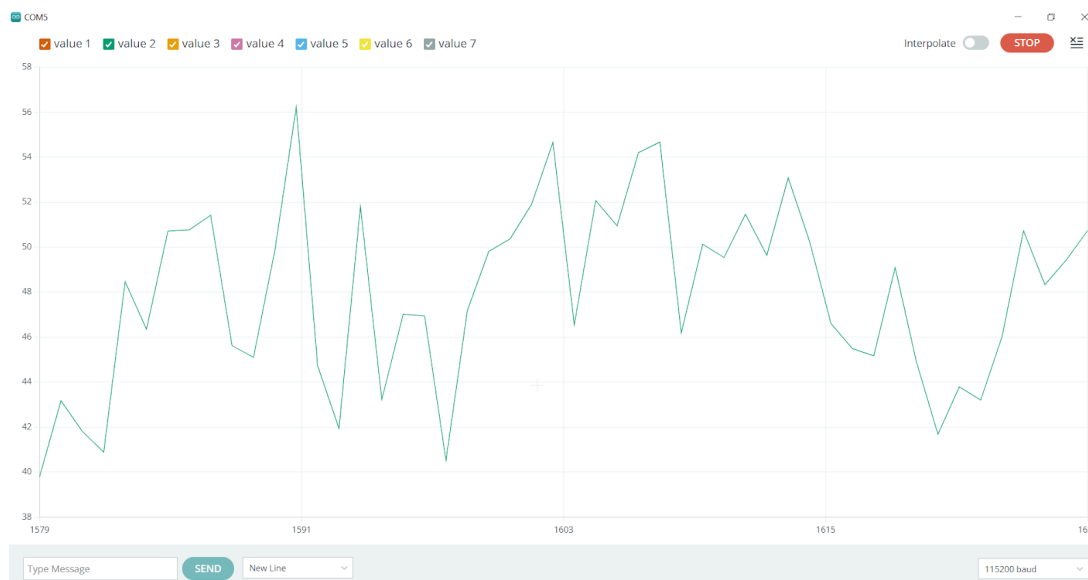


Figura 8. Ejemplo de datos del micrófono

Luz

Enviando este comando:

```
monitor light -notime -noms
```

Se obtendrá una salida similar a la de la *Figura 9*. La unidad de los datos es *luxes*. Al cubrir o acercar el sensor a una fuente de luz se perciben los cambios.

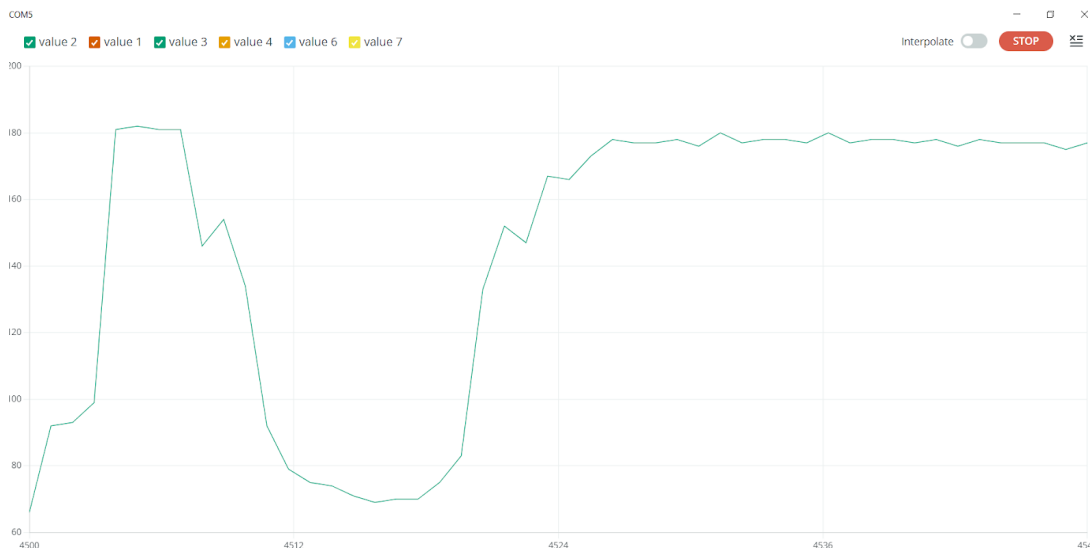


Figura 9. Ejemplo de datos del sensor de luz

Temperatura

Enviando este comando:

```
monitor temp -notime -noms
```

Se obtendrá una salida similar a la de la Figura 10. La unidad de los datos es grados centígrados (°C). Puede tocar el sensor de temperatura y humedad para ver cómo varían los datos.

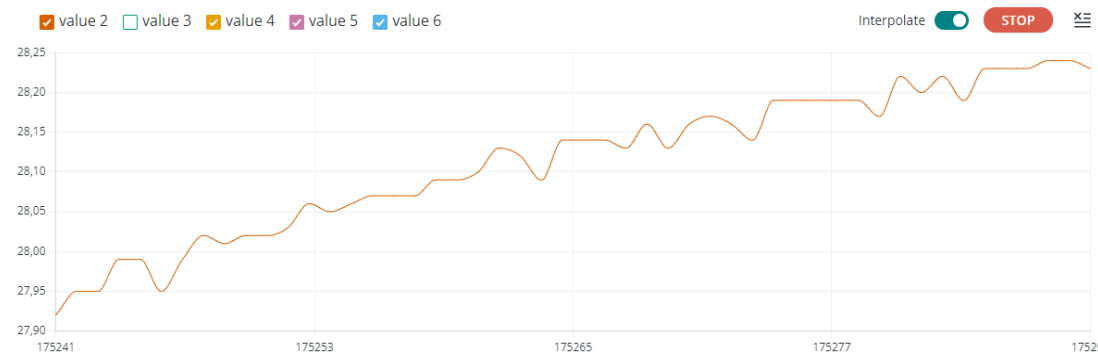


Figura 10. Ejemplo de datos del sensor de temperatura

Humedad relativa

Enviando este comando:

```
monitor humidity -notime -noms
```

Se obtendrá una salida similar a la de la *Figura 11*. La unidad de los datos es humedad relativa en porcentaje (%rh). Prueba a tocar el sensor de temperatura y humedad para ver cómo varían los datos.

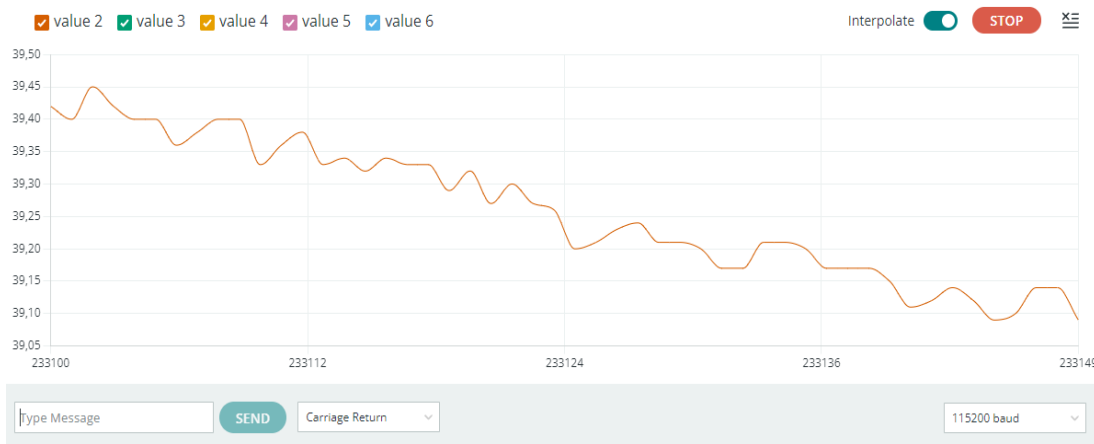


Figura 11. Ejemplo de datos del sensor de humedad relativa

Presión barométrica

Enviando este comando:

```
monitor pressure -notime -noms
```

Se obtendrá una salida similar a la de la *Figura 12*. La unidad de los datos es *kPa*. Esta variable no muestra mucha variación al manipular el sensor.

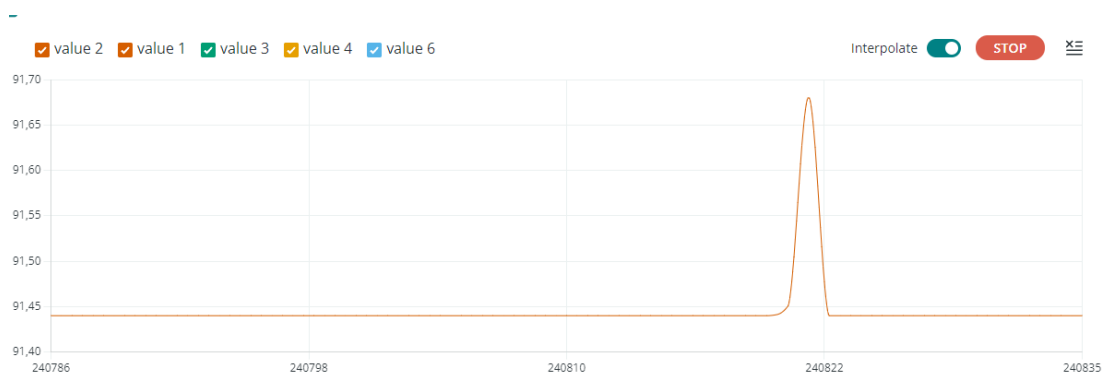


Figura 12. Ejemplo de datos del sensor de presión

Material Particulado

Enviando este comando:

```
monitor pm -notime -noms
```

Se obtendrá una salida similar a la de la *Figura 13*. La unidad de los datos es $\mu\text{g}/\text{m}^3$. Se puede utilizar un mechero o un borrador de tiza para producir cambios en la lectura del sensor.

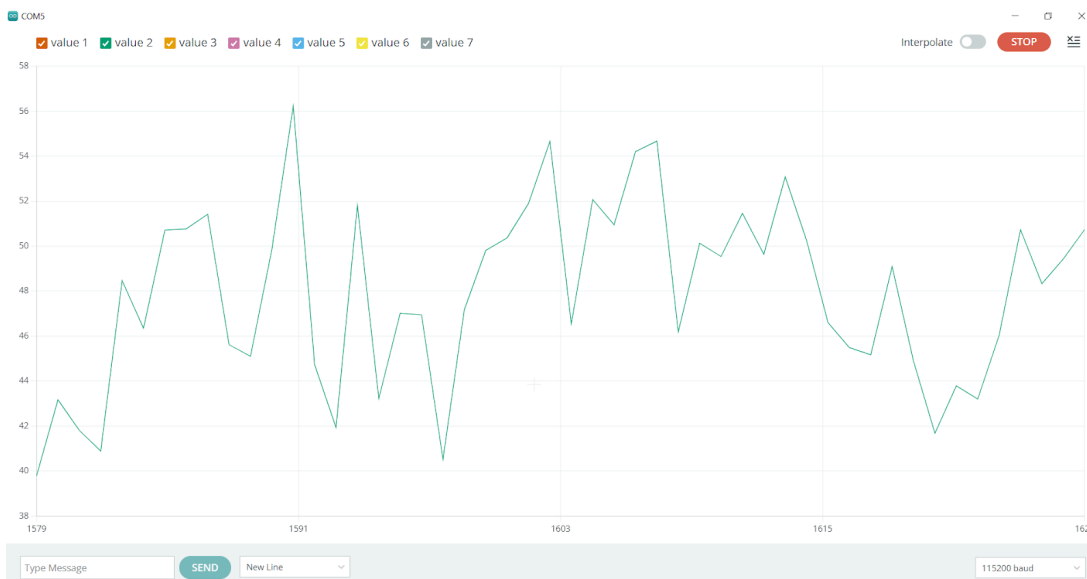


Figura 13. Ejemplo de datos del sensor de partículas

Tabla resumen de comandos

Sensor	Comando
Micrófono	monitor noise -notime -noms
Luz	monitor light -notime -noms
Temperatura	monitor temp -notime -noms
Humedad relativa	monitor humidity -notime -noms
Presión barométrica	monitor pressure -notime -noms
VOC Gas	monitor VOC -notime -noms
eCO2 Gas	monitor eCO2 -notime -noms
PM 1.0	monitor PM -notime -noms
PM 2.5	monitor PM2.5 -notime -noms
PM 10.0	monitor PM 10 -notime -noms

Modificar parámetros del Smart Citizen Kit

Introducción

Este tutorial detalla cómo modificar parámetros de medida del Smart Citizen Kit.

Requisitos

- **Software:** los materiales indicados a continuación requieren de la instalación previa del *Arduino IDE*, el cual se puede encontrar en la web del proyecto⁶. Esta instalación es posible en varias plataformas (*Windows, Mac y Linux*) y tiene una duración estimada de 5 minutos.
- **Materiales:** se necesita un ordenador con puerto USB, un cable USB a USB-micro y un Smart Citizen Kit.

Tiempo estimado

15 minutos

Descripción detallada

En esta sección, se detallarán los pasos a seguir en este tutorial.

1. En primer lugar, conecta el cable USB-micro al Smart Citizen Kit y al ordenador. En la *Figura 1* se muestra la ubicación del puerto USB.



Figura 1. Ubicación del puerto USB micro en el Smart Citizen Kit

En segundo lugar, en el ordenador, abre el software *Arduino IDE* (Figura 2). Una vez conectado el SCK al ordenador, en el *Arduino IDE*, se debe seleccionar el puerto USB

⁶ Web del proyecto Arduino: <https://www.arduino.cc/en/software>

que debe usar para la comunicación con el mismo. Para ello, se accede a la selección de puertos a través del menú *Tools > Port* (o *Herramientas > Puertos*). En el menú desplegable, se debe escoger aquel marcado como (Arduino Zero (Native USB Port)) o (Arduino/Genuino Zero (Autodetect Port)). Este menú se muestra en la *Figura 2* y su apariencia puede variar en función de la plataforma utilizada⁷.

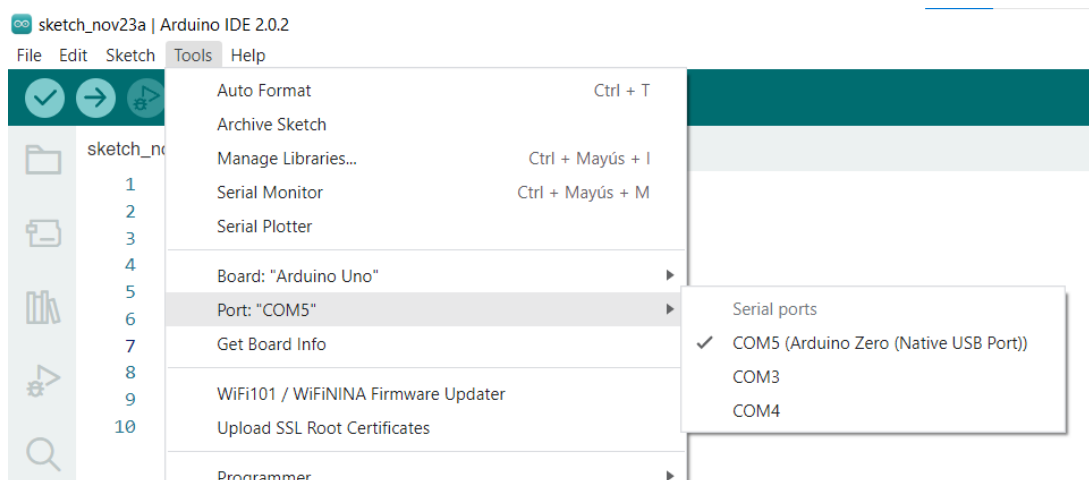


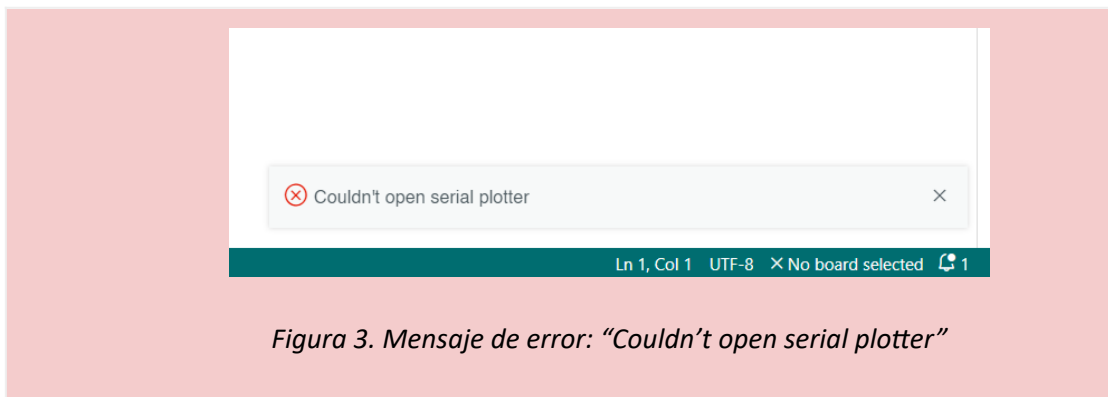
Figura 2. Selección del puerto en Arduino IDE

2. El *Serial Plotter* (*Monitor Serie* en castellano) es una herramienta que permite visualizar datos en tiempo real generados por un dispositivo conectado al puerto *USB* del ordenador (también llamado *Serial* en algunos tutoriales). El acceso al *Serial Plotter* se realiza a través del menú *Tools > Serial Plotter* (o *Herramientas -> Serial Plotter*).

Nota:

En el caso de que aparezca el mensaje: *Couldn't open serial plotter* (*Figura 3*), revisa si: (1) el cable *USB* está correctamente conectado, y que es defectuoso, (2) si el *SCK* está encendido (el *LED* indicador está encendido), o (3) si el puerto está correctamente seleccionado en el *Arduino IDE*. A continuación se detallan posibles soluciones a estos problemas.

⁷ Las figuras que muestran el Arduino IDE han sido creadas con la versión Arduino IDE 2.0, en un ordenador con plataforma Windows. Aunque otras versiones y plataformas pueden variar en la disposición de los menús y elementos, la funcionalidad es la misma.



- Comprueba que el cable no sea defectuoso simplemente utilizando otro cable. En ocasiones, es posible que el puerto *USB* del ordenador esté deshabilitado.
- Si el cable está conectado pero el dispositivo no aparece listado en el *Arduino IDE* (i.e. en el listado de puertos no aparece (*Arduino Zero (Native USB Port)*)) se puede deber a dos posibles motivos: o bien que hay un problema con el *SCK* (puede simplemente estar apagado), o bien puede que no esté bien instalado el software.

Para encender el dispositivo, pulsa el botón de *ON/OFF* del *SCK* (Figura 4).

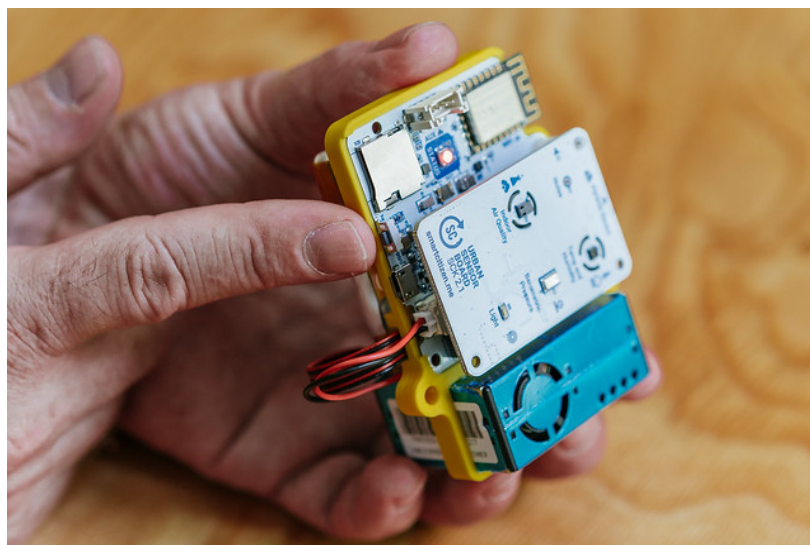


Figura 4. Botón *ON/OFF*

- Prueba a seleccionar el puerto en el *Arduino IDE*. En *Arduino 2.x* se puede seleccionar directamente el puerto desde un selector en la parte superior (Figura 5).

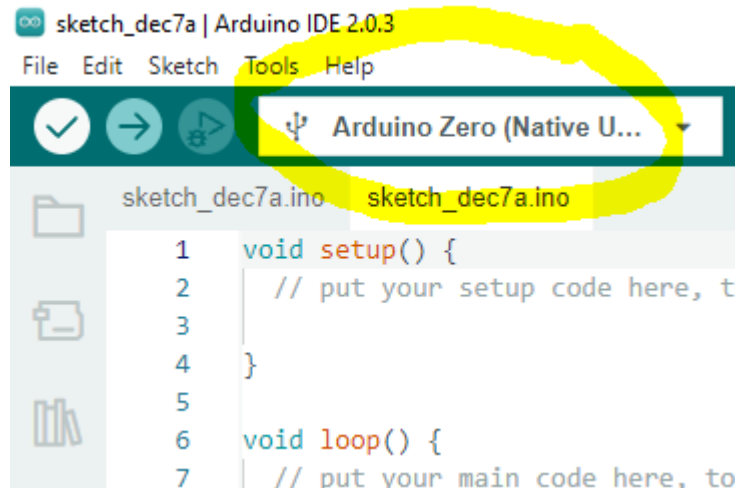


Figura 5. Selección del puerto serial en interfaz de Arduino IDE

Si se hace uso de este menú desplegable, es posible que el programa requiera instalar una placa adicional (se indicará como *Smart Citizen*) en la parte inferior del programa. Adicionalmente, en algunas plataformas, por ejemplo en *Windows* es posible que se requieran permisos adicionales para la instalación (es importante realizar estos pasos con *Permisos de Administrador*). Los mensajes del terminal se pueden ver en la Figura 6.

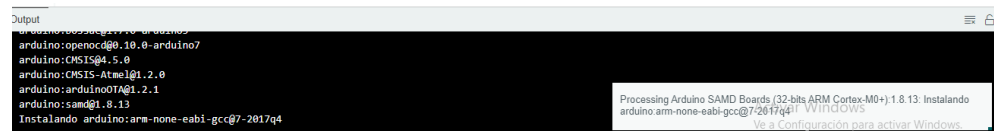


Figura 6. Instalación de la placa de Smart Citizen Kit

- Una vez configurado el *Monitor Serial*, se pueden introducir comandos para que sea interpretado por el *SCK* (en *Herramientas > Monitor serial*). En esta interfaz se pueden cambiar parámetros de lectura y configuración del dispositivo. Por ejemplo, para establecer un cambio en el intervalo de lectura y publicación a 5s, hay que escribir este comando en la línea de comandos y pulsar *ENTER*:

```
config -pubint 5 -readint 5
```

Se debería recibir por la consola un mensaje similar al siguiente:

```
-- New config --
Mode: network
Publish interval (s): 5
Reading interval (s): 5
Wifi credentials: [configuración del wifi]
Token: [token]
```

```
Mac address: [dirección MAC del dispositivo]  
...
```

Nota:

Esta configuración (5 segundos de publicación y lectura) tiene un consumo muy alto de energía, y reduce notablemente la duración de la batería. Es útil si se va a dar una demostración o una sesión didáctica pero, en caso de utilizar el *Smart Citizen* únicamente alimentado a batería por periodos largos de tiempo, es recomendable configurarlo a un intervalo superior. Para restaurar el valor por defecto de actualización, escribe el siguiente comando:

```
config -pubint 180 -readint 60
```

Se debería recibir por la consola un mensaje similar al siguiente:

```
-- New config --  
Mode: network  
Publish interval (s): 180  
Reading interval (s): 60  
...
```

Finalmente, si quieres conocer más sobre comandos disponibles, puedes revisarlos en la documentación en la siguiente guía:

- https://docs.smartcitizen.me/Guides/getting_started/Using_the_Shell/#connectivity-or-configuration-issues

Adicionalmente, existe una serie de *webinars* sobre la línea de comandos (*Shell*) disponibles en *youtube*:

- <https://www.youtube.com/watch?v=ZNStSudpg0A&list=PL33KKs9g8Y1IWstZmDc-46yFuulRZEmi&index=5>

Herramientas web

Acceso a los datos a través de la plataforma web

Introducción

Este tutorial permite revisar los datos de un *Smart Citizen Kit* cualquiera, propio o de otro usuario, para consultar los datos que éste tiene. Para ello se utilizará la página smartcitizen.me/kits/

Es útil para revisar los datos subidos a la plataforma en cualquier momento o si se hace una demostración en *casi-tiempo real*.

Requisitos

- **Software:** Un navegador web.
- **Materiales:** Un ordenador con conexión a internet.

Tiempo estimado

5 minutos

Descripción detallada

1. En un navegador, accede a smartcitizen.me/kits (Figura 1). Si quieres, filtra entre los kits para encontrar el que prefieras. El buscador tiene opciones por nombre y también se puede ampliar el mapa.



Figura 1. Ventana principal del mapa

Una vez seleccionado el *SCK*, se pueden ver los datos en la sección inferior como se indica en la Figura 2.

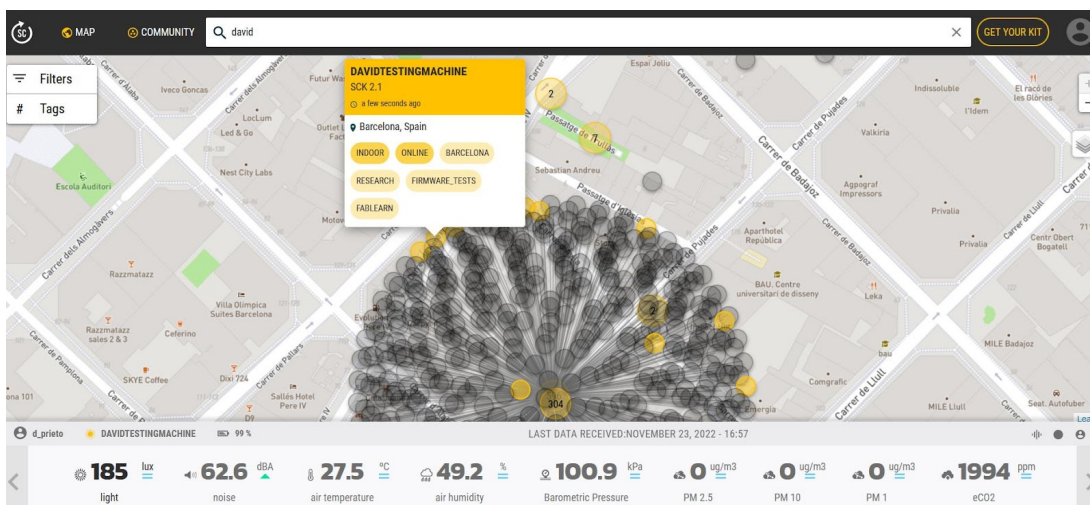


Figura 2. Un kit seleccionado en el mapa y sus datos en la parte inferior.

Los datos mostrados son aquellos que se han recibido más recientemente en la plataforma y se actualizan cada vez que el servidor recibe nuevos datos del dispositivo. Por defecto los datos se envían cada 5 minutos y se leen cada minuto.

En la parte inferior de la página se pueden ver unas gráficas de los datos (Figura 3).

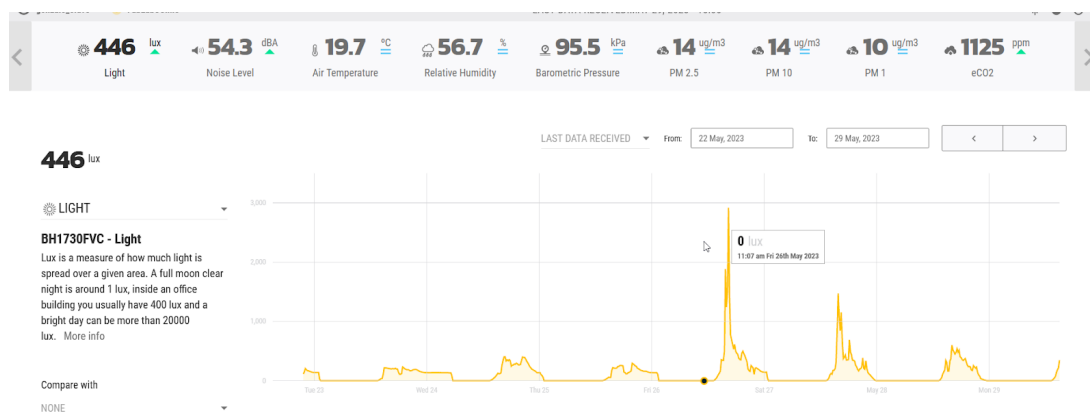


Figura 3. Gráficos de un kit en la plataforma web.

Acceso a los datos a través del dashboard

Introducción

Este tutorial detalla cómo acceder a los datos de un *Smart Citizen Kit* cualquiera, propio o de otro usuario a través de un *dashboard*. Para ello usaremos <https://dashboard.smartcitizen.me>.

Es útil para revisar los datos subidos a la plataforma en cualquier momento o si se hace una demostración en *casi-tiempo real*.

Requisitos

- **Software:** Un navegador web.
- **Materiales:** Un ordenador con conexión a internet.

Tiempo estimado

5 minutos

Descripción detallada

1. Accede a dashboard.smartcitizen.me. Si quieres, puedes filtrar entre los kits en la parte superior (*Figura 1*).

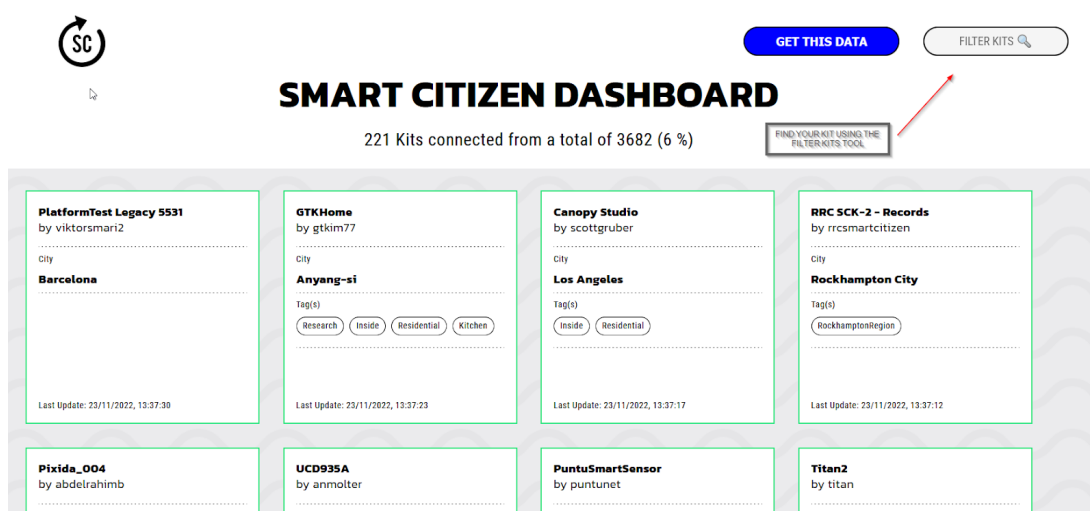


Figura 1. Página principal del dashboard

Se puede utilizar el usuario, el nombre del kit, la ciudad o cualquier otro dato para encontrar el kit deseado.

Para seleccionar el dispositivo, haz *click* en el nombre del mismo. Esto abrirá la pantalla de visualización de datos (*Figura 2*) donde se pueden visualizar gráficas temporales de todos los sensores de un dispositivo. Se puede ampliar la serie temporal:

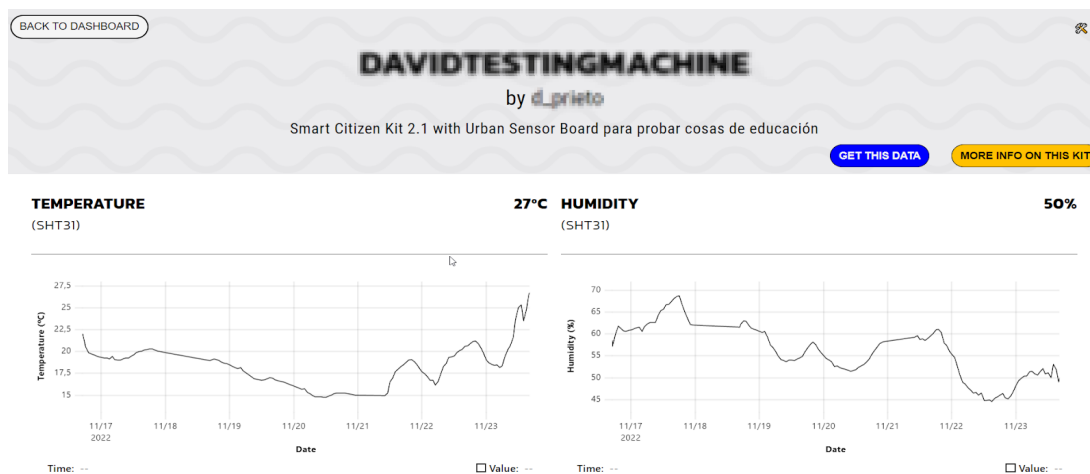



Figura 2. Visualización de datos en el dashboard

En el caso de que desees ver únicamente los valores de los datos recibidos, y no las gráficas, haz click en , el icono de configuración en la parte superior derecha como se ve en la *Figura 3*.

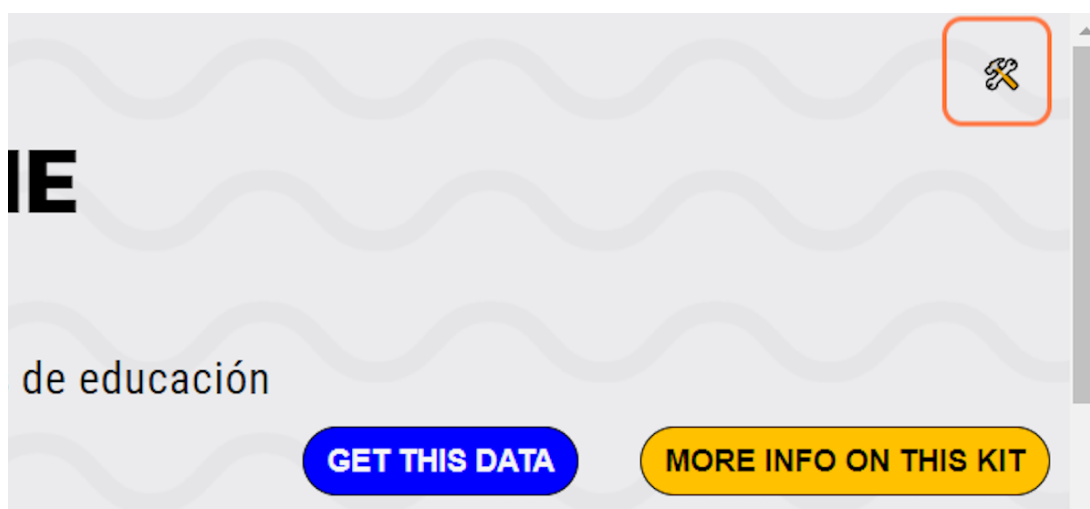


Figura 3. Icono de configuración

Se abrirá una barra lateral en azul (*Figura 4*). Pulsa el botón de mostrar gráficas (*SHOW GRAPHS*) para desactivarlas.



Figura 4. Barra lateral de configuración.

En este caso, se indican únicamente los últimos datos recibidos (Figura 5). Hay que pulsar de nuevo el icono para cerrar el menú de la derecha.

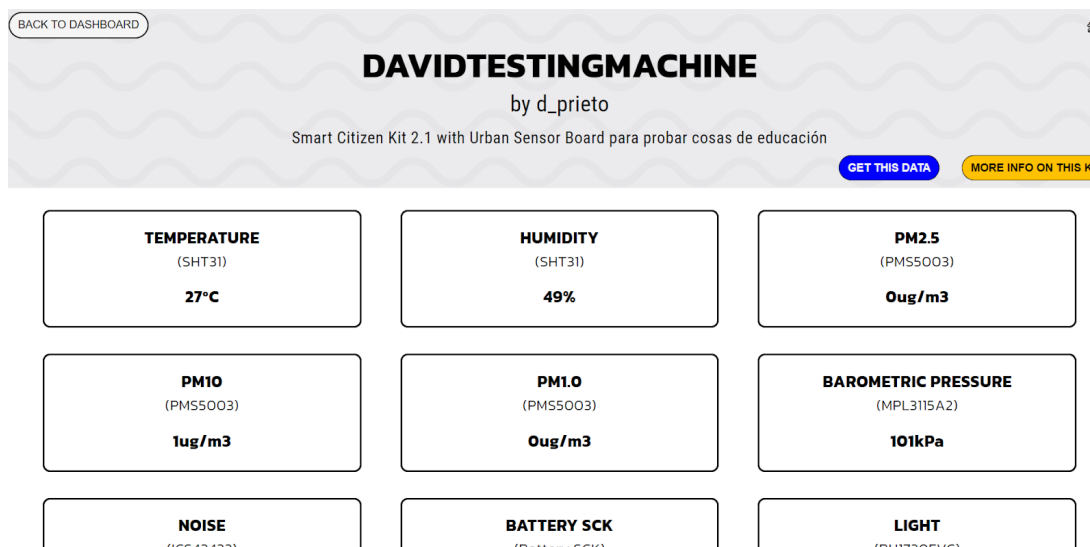


Figura 5. Dashboard sin gráficas.

Nota: Se puede usar el menú lateral también para filtrar los valores o fechas que sean más interesantes.

Orange Data Mining

Configuración de Orange Data Mining

Este tutorial muestra cómo configurar *Orange Data Mining* para descargar, consultar y analizar datos de la plataforma de *Smart Citizen*.

Introducción

Para utilizar *Orange Data Mining* con datos de la plataforma de *Smart Citizen*, es necesario configurar previamente el programa. En concreto, se deben instalar una serie de extensiones (*Add-ons*) que se utilizarán en los siguientes tutoriales.

Este tutorial se puede encontrar en inglés en la documentación oficial⁸.

Requisitos

- **Software:** *Orange Data Mining* instalado. Se puede encontrar en la web del proyecto⁹. Esta instalación es posible en varias plataformas y tiene una duración estimada de *20 minutos*.
- **Materiales:** Un ordenador y conexión a internet.

Tiempo estimado

15 minutos

Descripción detallada

1. Abre *Orange Data Mining*. En los menús superiores selecciona *Options* -> *Add-ons*. Se desplegará una ventana como la *Figura 1* (la apariencia puede variar en función de la plataforma utilizada¹⁰):

⁸ Documentación oficial:

<https://docs.smartcitizen.me/Resources/Tutorials/Configure%20Orange%20Data%20Analysis/>

⁹ Web del proyecto Orange Data Mining: <https://orangedatamining.com/>

¹⁰ Las figuras se han realizado a través de la versión 3.34 de Windows pero otras versiones tendrán un mismo funcionamiento.

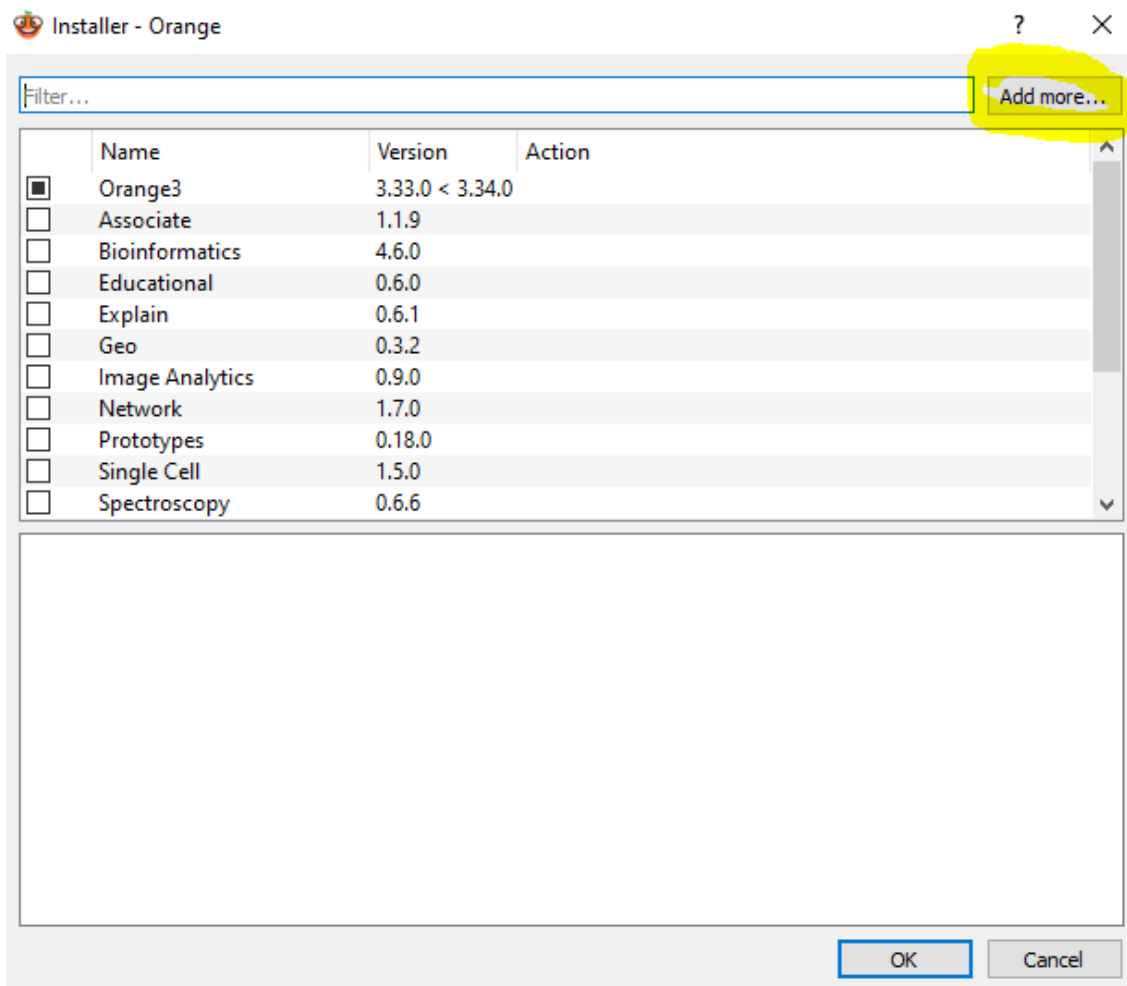
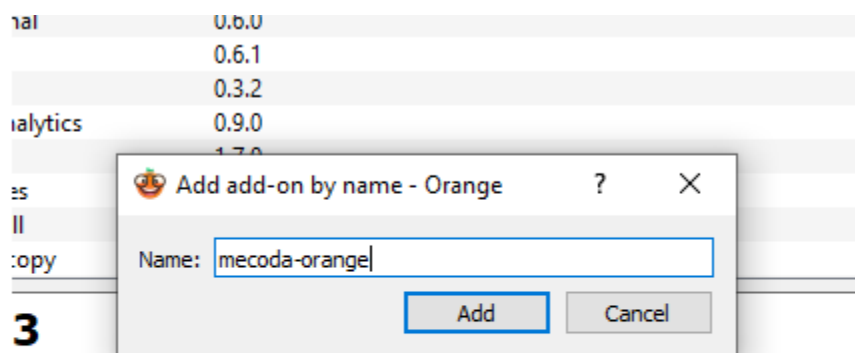


Figura 1. Menú de Add-ons (extensiones) de Orange Data Mining

Selecciona *Add more...* en la parte superior izquierda (Figura 1) y escribe exactamente *mecoda-orange* en la ventana emergente. Haz *click* en *Add* como se indica en la Figura 2.



ment-based data mining software. It includes a range of data visualization, exploration, modeling techniques. It can be used through a nice and intuitive user interface or, for me is a module for the Python programming language.

ersion of Orange (for Python 3). The deprecated version of Orange 2.7 (for Python 2.7) and [sources](#)).

Figura 2. Menú de búsqueda de add-ons

Tras esto, verás que en la lista aparece *Mecoda Orange* al final.

Selecciona *Mecoda Orange*, *Geo* y *Timeseries* y haz *click* en *Ok* para instalarlas. Si se cierra la página sin dar a *Ok*, no se instalará nada. En la *Figura 3* se muestran las extensiones a instalar.

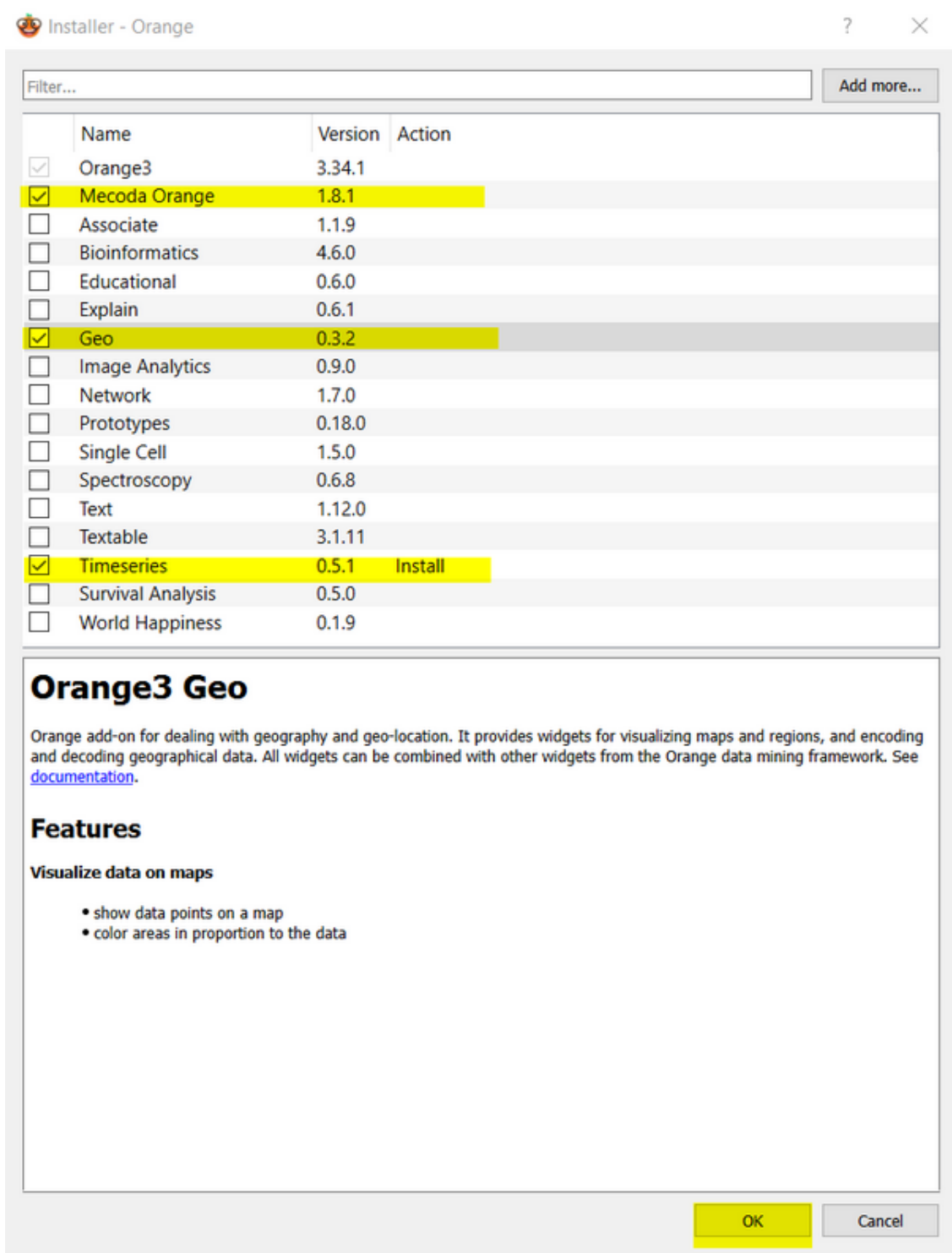


Figura 3. Extensiones para trabajar con Smart Citizen seleccionadas

Nota: El proceso de instalación se iniciará en este momento, y aparecerá una ventana de instalación que muestra el progreso del proceso. Este proceso puede durar entre **5 y 10 minutos**, por lo que es importante realizar esta acción previamente la sesión de trabajo con el alumnado. Tras la instalación, Orange deberá reiniciarse antes de continuar.

Tras el reinicio se puede revisar que está todo instalado observando los menús de la izquierda donde se pueden ver las nuevas incorporaciones: *MECODA*, *Timeseries* y *Geo*. La columna izquierda debería ser similar a la mostrada en la *Figura 4*.

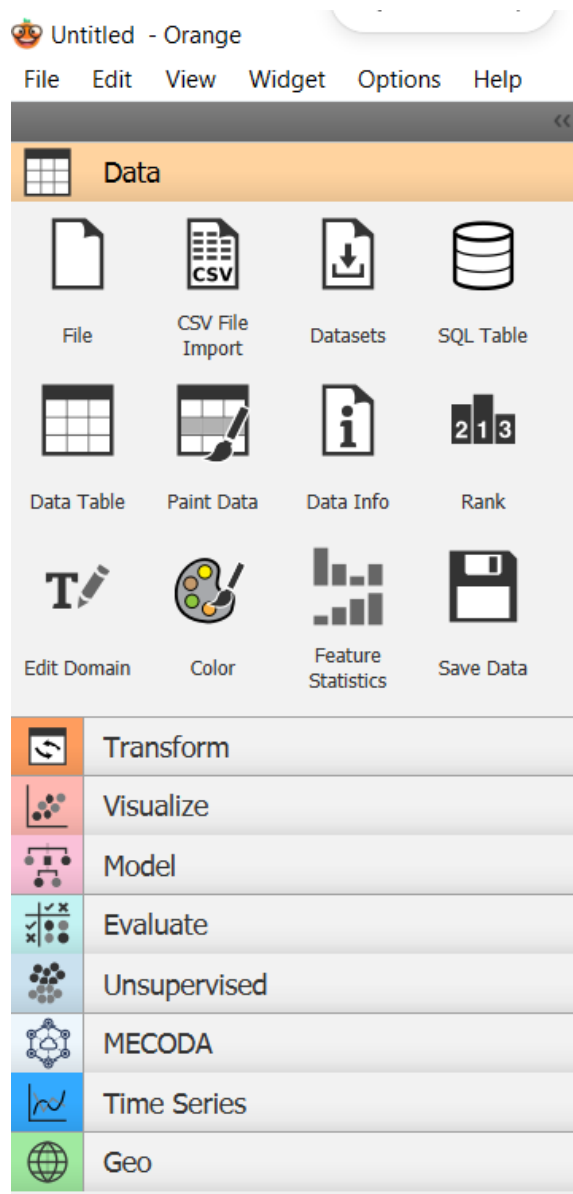


Figura 4. Extensiones instaladas en panel lateral.

Acceso a datos de Smart Citizen a través de Orange

Introducción

Este tutorial sirve como introducción a la forma de trabajar con *Orange Data Mining*. Se muestra cómo se pueden descargar los datos de la plataforma de *Smart Citizen*. Entender cómo hacer este tutorial puede posteriormente usarse para trabajar conceptos más avanzados como derivadas y tendencias en apartados posteriores.

Se parte de unos datos de ejemplo, pero está pensado este tutorial para que sean sustituibles para cualquier medición.

Este tutorial se puede encontrar en inglés en la documentación oficial¹¹:

Requisitos

- **Software:** *Orange Data Mining* instalado. Se puede encontrar en la web del proyecto¹². Esta instalación es posible en varias plataformas y tiene una duración estimada de *20 minutos*.
- **Materiales:** Un ordenador y conexión a internet.

Tiempo estimado

15 minutos

Descripción detallada

1. Abre *Orange Data Mining* abierto, desde el menú de la izquierda, selecciona el *widget Smart Citizen Search*. Creará un *nodo* como el de la *Figura 1* en la parte central de *Orange Data Mining*:



Figura 1. Widget de Smart Citizen Search en el lienzo

Al hacer *click* en el *widget* se abre una ventana como la de la *Figura 2* con varias opciones disponibles.

¹¹ Documentación oficial: <https://docs.smartcitizen.me/Resources/Tutorials/Simple%20Orange%20Workflow/>

¹² Web del proyecto Orange Data Mining: <https://orangedatamining.com/>

Smart Citizen Search - Orange

Info

No data fetched yet.

Search filters

City:

Tags (comma separated):

User name:

Kit ID:

Or directly find device

Device ID:

Search devices

?

Figura 2. Opciones del widget de Smart Citizen Search

Aquí, busca el kit que quieras o usa los campos de la parte superior para buscar por otros parámetros, por ejemplo, usando el nombre de usuario. Puedes buscarlo con el mapa visitando <https://smartcitizen.me/kits/> en un navegador web y entonces, buscar el kit en el mapa y seleccionarlo. (Figura 3)

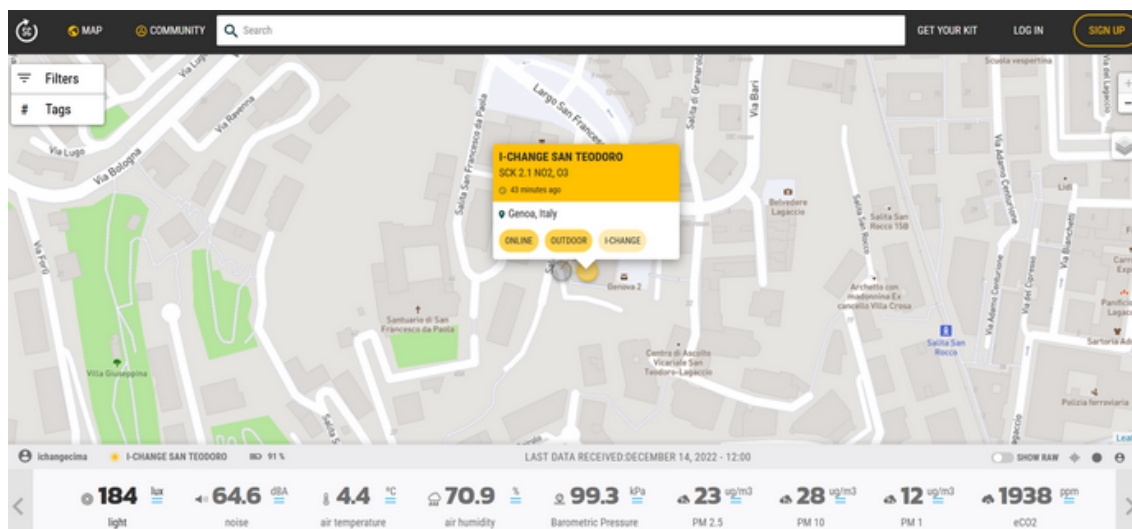


Figura 3. Selección de un Kit en la plataforma

Al seleccionarlo la URL cambiará a otra, por ejemplo: <https://smartcitizen.me/kits/15618>.

En caso de no tener un *kit*, se puede usar un *kit* de ejemplo, usando en *Device ID* el número 15618. Tras esto, pulsa *Search Devices*.

Una vez encontrado el *kit*, pasamos a descargar los datos del mismo. Para ello, usa otro *widget*: *Smart Citizen Data* (Figura 4).

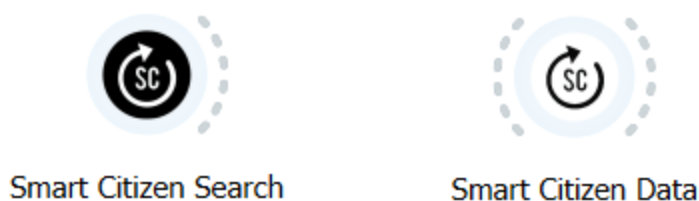


Figura 4. Widgets de Smart Citizen Search junto al de Smart Citizen Data.

Ahora, prueba a conectar los nodos. Esto permite que los datos de uno pasen al otro (Figura 5).

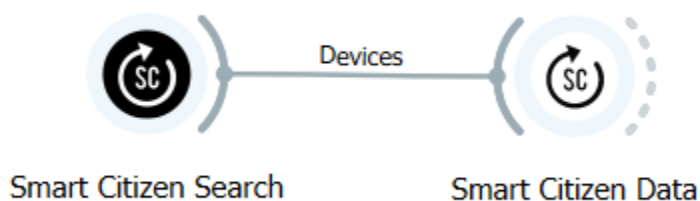


Figura 5. Unión de widgets.

Ahora, haz click sobre el *widget Smart Citizen Data*. En el caso de los datos de ejemplo debería mostrar algo similar a la *Figura 6*.

The screenshot shows a configuration window for 'Smart C...'. It has a title bar with a refresh icon, a minimize button, a maximize button, and a close button. The main content area is divided into several sections:

- Info:** Device: 15618, Name: I-CHANGE SAN TEODORO, City: Genoa (IT), By: ichangecima
- Get data at a specific frequency:** Rollup: 10, Rollup units: m
- Filter by date (YYYY-MM-DD):** Initial Date: [empty], End Date: [empty]
- Resample data
- Get data** button
- A help icon (?) at the bottom left.

Figura 6. Panel de configuración de Smart Citizen Data

En el panel de información superior se ve la información del kit que hemos seleccionado. Si no aparece información, es posible que haya un problema en la configuración del *widget Smart Citizen Search*.

Si aparece información en la parte superior, es el momento de seleccionar qué datos queremos descargar. En este ejemplo, usaremos una frecuencia de datos cada 10 minutos por un periodo de un mes: desde el 1 de octubre de 2022 hasta el último día de ese mes.

Para establecer la frecuencia de los datos, indica en "Rollup:" 10 y en "Rollup units:" en m (*m* para *minutos*, *s* para *segundos*, *ms* para *milisegundos*, *h* para *horas*, *d* para *días*).

Después, hay que indicar el periodo de fechas en caso de buscar una fecha en particular. Los datos con el ejemplo se pueden ver en la *Figura 7*.

The screenshot shows a configuration panel for 'Smart Citizen Data'. It is divided into two main sections. The first section is titled 'Get data at a specific frequency' and contains three input fields: 'Rollup:' with the value '10', 'Rollup units:' with a dropdown menu showing 'm', and a note below stating 'Rollup needs to be an integer'. The second section is titled 'Filter by date (YYYY-MM-DD)' and contains two date input fields: 'Initial Date:' with the value '2022-10-01' and 'End Date:' with the value '2022-11-01'. At the bottom of the panel, there is a checkbox labeled 'Resample data' which is currently unchecked.

Figura 7. Panel de configuración de Smart Citizen Data

La opción de *Resample data* hará un promedio en caso de que haya datos incompletos. En este caso no usaremos esta opción. Por último, selecciona *Get data* para obtener los datos. Tras unos segundos, en la parte inferior aparecerá un mensaje como: *Device XXXXX data downloaded!* donde XXXXX es el número del *kit* que se ha insertado (en el caso del ejemplo, 15618).

2. A continuación mostraremos los datos en formato tabular. Selecciona el *widget Data table* en el panel lateral izquierdo, en el apartado *Data* como se ve en la *Figura 8*.

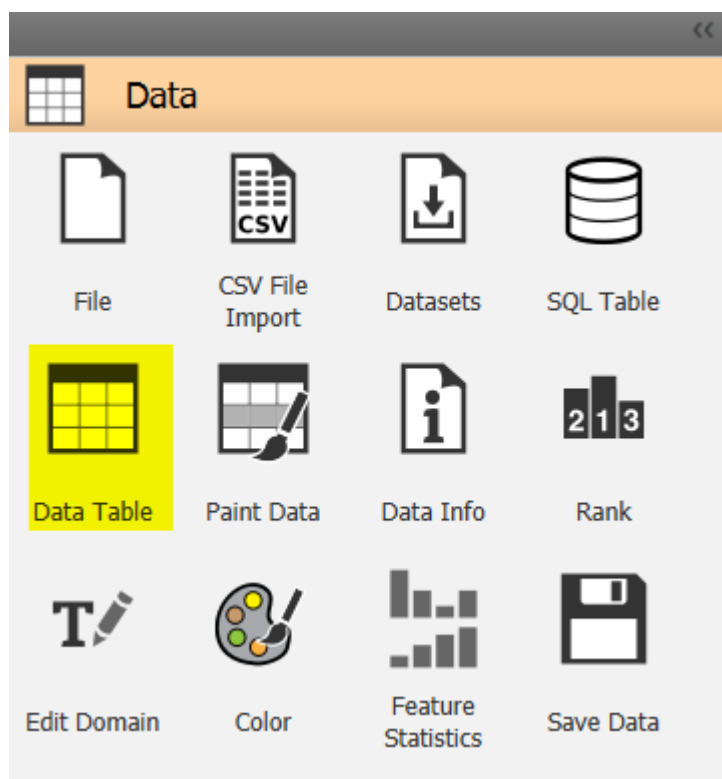


Figura 8. Panel Data

Haz click en *Data Table*, y aparecerá en el lienzo tal y como se muestra en la Figura 9.

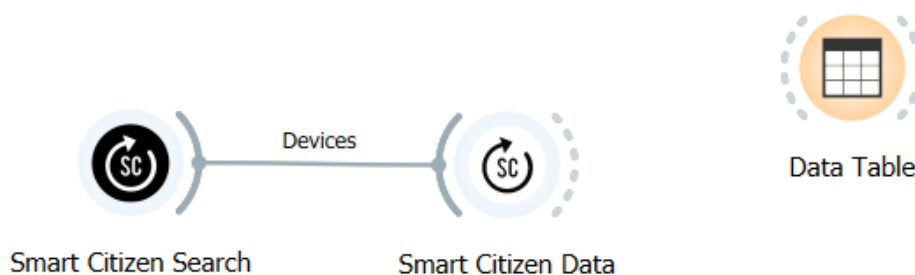


Figura 9. Lienzo con los tres widgets.

Conecta este *widget* con el de Smart Citizen Data como se hizo anteriormente entre los otros dos widgets (Figura 10).

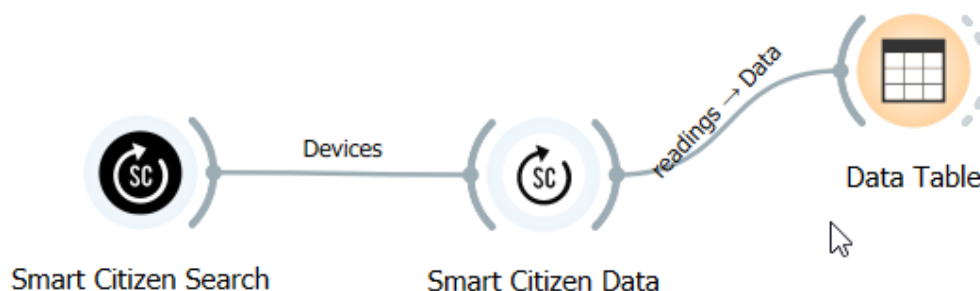


Figura 10. Conexión de los 3 widgets

Después de conectar el *widget*, haz *click* en el mismo y aparecerán los datos descargados en la tabla¹³. Al abrir el *widget* debe aparecer una ventana con los datos similar a la de la Figura 11.

	TIME	ADC_48_0	ADC_48_1	ADC_48_2
52	2022-10-02 03:00:21+0200	0.250625	0.251656	0.40
53	2022-10-02 04:00:00+0200	?	?	
54	2022-10-02 04:00:21+0200	0.250219	0.251156	0.40
55	2022-10-02 05:00:00+0200	?	?	
56	2022-10-02 05:00:20+0200	0.249375	0.249781	0.40
57	2022-10-02 06:00:00+0200	?	?	
58	2022-10-02 06:00:20+0200	0.250125	0.248969	0.40
59	2022-10-02 07:00:00+0200	?	?	
60	2022-10-02 07:00:20+0200	0.249781	0.249844	0.40
61	2022-10-02 08:00:00+0200	?	?	
62	2022-10-02 08:00:20+0200	0.249687	0.249875	0.40
63	2022-10-02 09:00:00+0200	?	?	
64	2022-10-02 09:00:20+0200	0.250094	0.250125	0.40
65	2022-10-02 10:00:00+0200	?	?	
66	2022-10-02 10:00:20+0200	0.246312	0.247156	0.40
67	2022-10-02 11:00:00+0200	?	?	
68	2022-10-02 11:00:20+0200	0.245906	0.246156	0.40
69	2022-10-02 12:00:00+0200	?	?	
70	2022-10-02 12:00:20+0200	0.245375	0.2455	0.40
71	2022-10-02 13:00:00+0200	?	?	
72	2022-10-02 13:00:20+0200	0.245906	0.245937	0.40

Figura 11. Datos descargados mostrados como tabla.

¹³ Para obtener más ayuda de cómo funcionan las tablas de datos, visita esta sección de la documentación de Orange:

<https://orange3.readthedocs.io/projects/orange-visual-programming/en/latest/widgets/data/datatable.htm>

3. Para representar los datos en una gráfica temporal, es necesario primero convertirlos a *serie temporal* (*Timeseries*). Para ello, en el menú lateral, en el apartado *Time Series* (*add-on* instalado previamente en *Orange*) seccionales el *widget* con el nombre *As Timeseries*. Su localización se puede ver en la *Figura 12*.

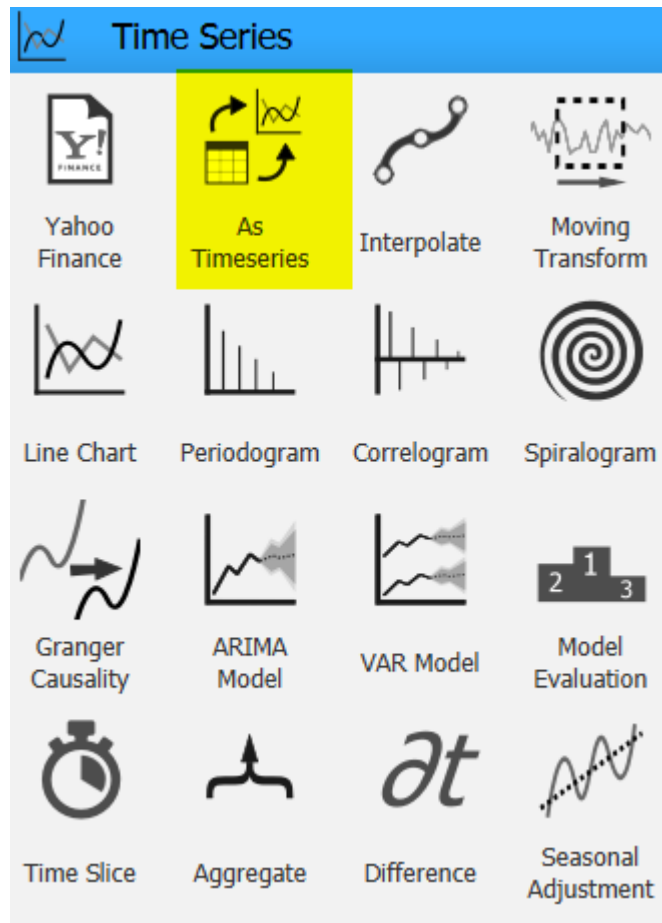


Figura 12. Menú de Time Series

Añade al lienzo y conéctalo al resto de *widgets*: puedes conectarlo al de la tabla de datos o al *Smart Citizen Data* como en la figura 13. Un mismo *widget* puede tener varias salidas.

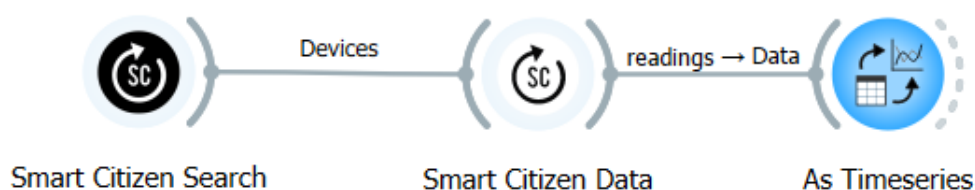


Figura 13. Conexión de widgets

Por último, añade *Line chart*, el cual se encuentra en el menú *Time Series* de la figura 12. Conéctalo al *widget As Timeseries*, y marca las propiedades como se indican en la Figura 14.

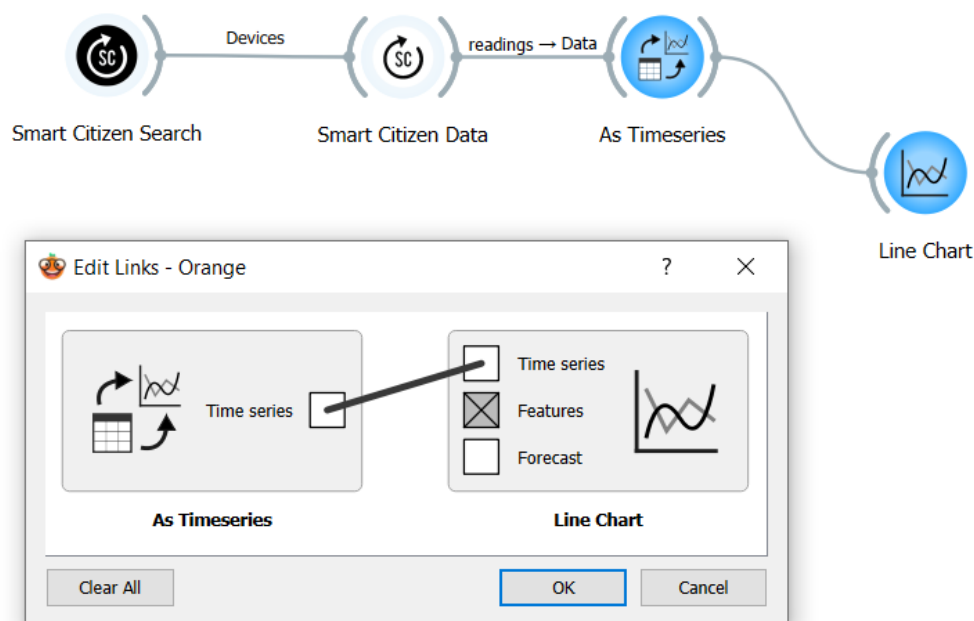


Figura 14. Configuración del widget de Line Chart

A continuación, haz click en el widget de Line Chart (Figura 15).

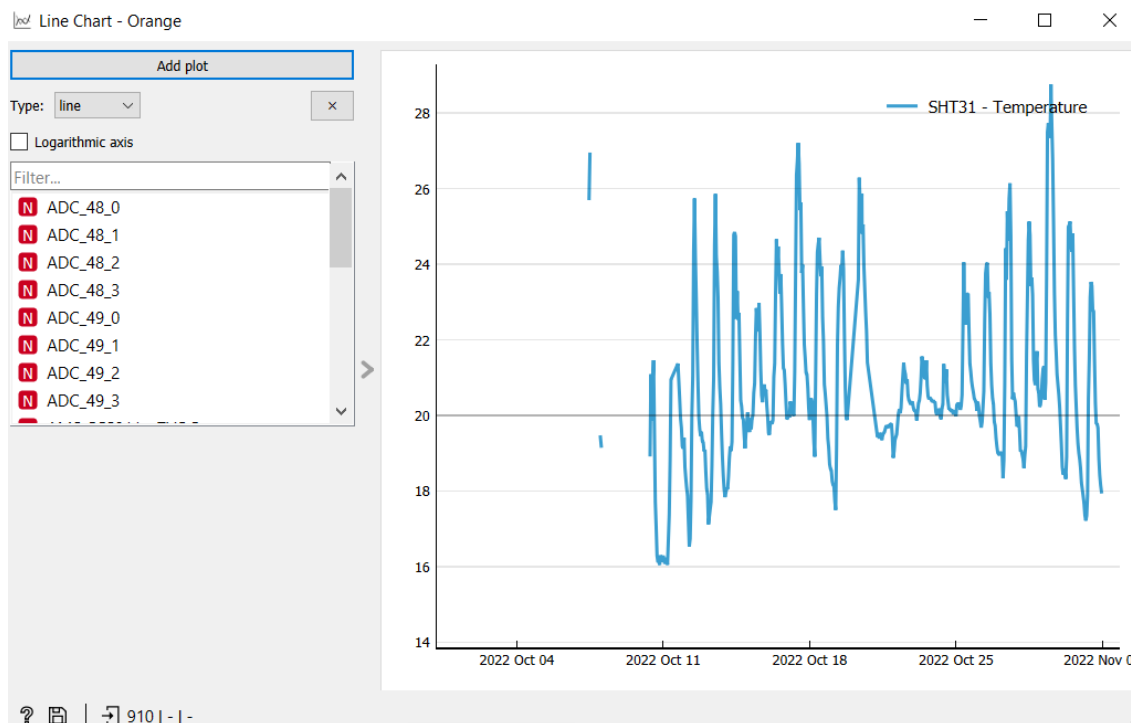


Figura 15. Interior de Line Chart

Por defecto, este nodo mostrará una de las columnas. En estas gráficas, podemos ver los cambios intradiarios, las variaciones semanales, y comenzar a analizar los datos capturados. Se pueden añadir más visualizaciones para comparar las lecturas de un sensor con otro. Por ejemplo, es posible comparar temperatura y humedad como se indica en la *Figura 16*.

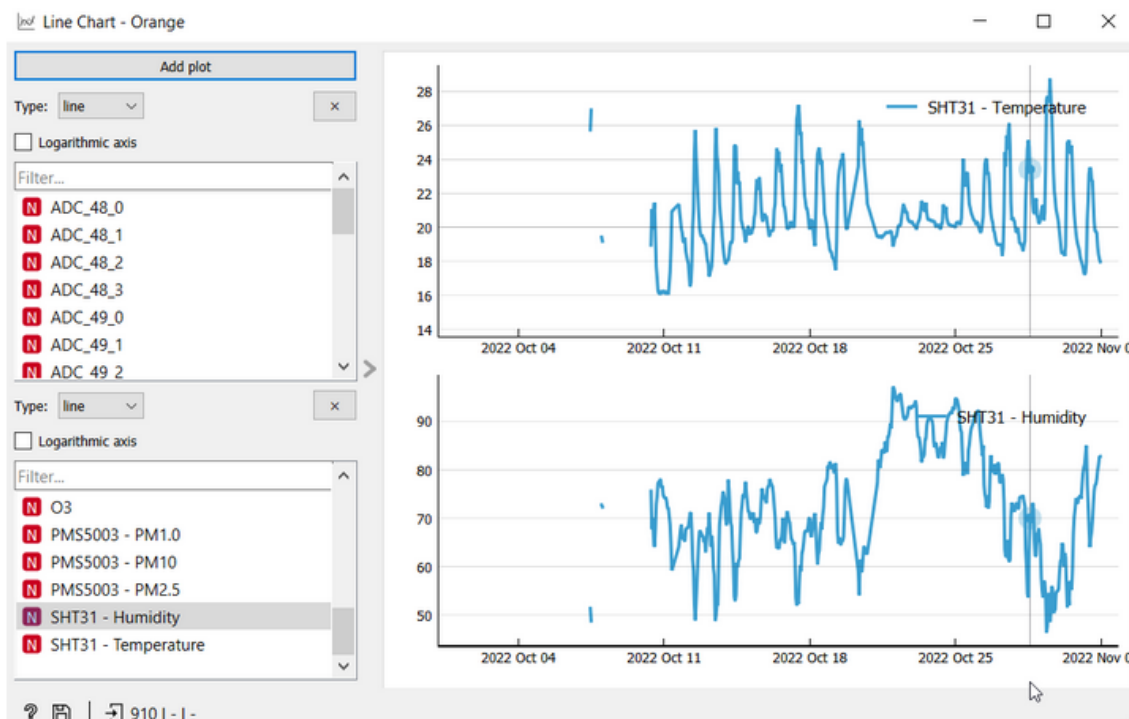


Figura 16. Comparación de temperatura y humedad.

Por último, es posible descargar datos de más de un dispositivo simultáneamente. Para ello, descarga los datos por separado (cada uno en su nodo de descarga, siguiendo los pasos anteriores). Después, puedes usar varias técnicas, por ejemplo, el *widget Merge Data* que se encuentra en el menú lateral, en el apartado *Transform* (Figura 17). Es importante tener en cuenta que los datos duplicados serán renombrados (añadiendo (1) al final de cada columna).

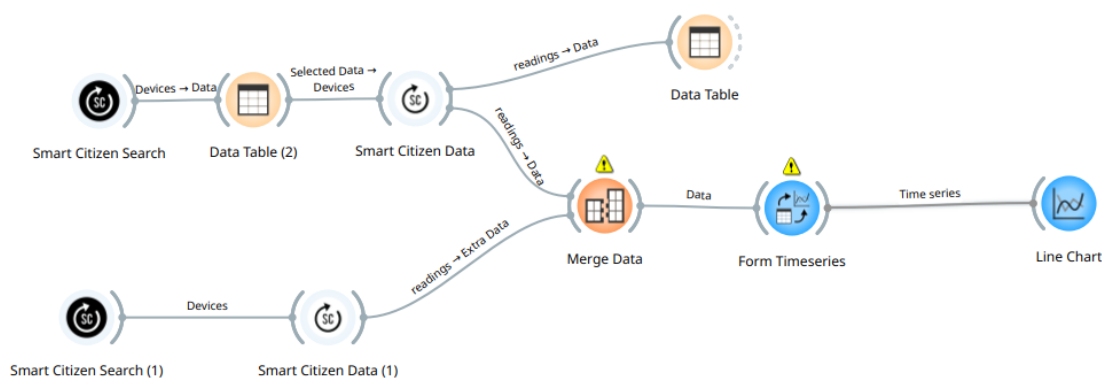


Figura 17. Comparación de temperatura y humedad.

Representación de datos geolocalizados con Orange

Introducción

Este tutorial sirve como introducción a la forma de trabajar de *Orange Data Mining* con datos geolocalizados. presentando cómo se pueden recoger los datos de la plataforma de *Smart Citizen*.

Se parte de unos datos de ejemplo, pero está pensado este tutorial para que sean sustituibles para cualquier medición.

Este tutorial se puede encontrar en inglés en la documentación oficial¹⁴:

Requisitos

- **Software:** *Orange Data Mining* instalado. Se puede encontrar en la web del proyecto¹⁵. Esta instalación es posible en varias plataformas y tiene una duración estimada de *20 minutos*.
- **Materiales:** Un ordenador y conexión a internet.

Tiempo estimado

15 minutos

Descripción detallada

1. Abre *Orange Data Mining* abierto, desde el menú de la izquierda, selecciona el widget *Smart Citizen Search*. Creará un *nodo* como el de la *Figura 1* en la parte central de *Orange Data Mining*:



Figura 1. Widget de Smart Citizen Search en el lienzo

Al hacer *click* en el *widget* se abre una ventana como la de la *Figura 2* con varias opciones disponibles.

¹⁴ Documentación oficial: [https://docs.smartcitizen.me/Resources/Tutorials/Orange and geolocated data/](https://docs.smartcitizen.me/Resources/Tutorials/Orange%20and%20geolocated%20data/)

¹⁵ Web del proyecto Orange Data Mining: <https://orangedatamining.com/>

Figura 2. Opciones del widget de Smart Citizen Search

Aquí, busca el kit que quieras o usa los campos de la parte superior para buscar por otros parámetros, por ejemplo, usando el nombre de usuario. Puedes buscar un *kit* con *GPS* en el apartado de *Kit ID* como se muestra en la Figura 2.

Al buscar por *Kit ID*, obtendrás una tabla de resultados con ese ID. Puedes utilizar una *Data Table* y seleccionar en la misma para enviar una selección al nodo Smart Citizen Data. El diagrama completo se muestra en la Figura 3, y la selección en la tabla (*Data Table* en la Figura 3) se muestra en la Figura 4.

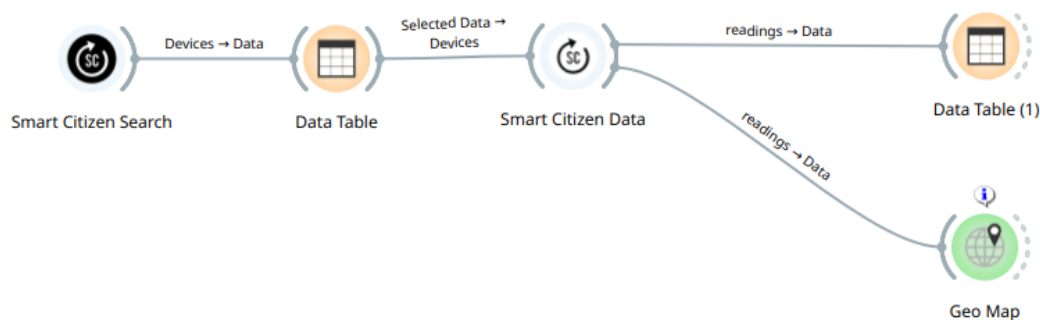


Figura 3. Flujo para datos geolocalizados

Asegúrate de marcar la línea que quieras, en este caso usaremos

Info	name	id	description	added_at	updated_at	last_reading_at	owner_id	owner_username	user_tags	system_tags	state	kit_id	latitude	longitude	city	country_code	
58 instances (no missing data)	23 31F1	14422	Smart Citizen	2021-07-15 08:...	2022-04-19 11:...	2021-07-15 08:39:39	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
15 features	37 9E8B	14423	Smart Citizen	2021-07-15 08:...	2022-04-19 11:...	2021-07-15 08:45:21	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
No target variable	28 0C47	14424	Smart Citizen	2021-07-15 08:...	2022-04-19 11:...	2021-07-15 08:49:45	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
1 meta attribute	27 0114	14425	Smart Citizen	2021-07-15 08:...	2022-04-19 11:...	2021-07-15 08:34:41	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
Variables	25 4DC9	14427	Smart Citizen	2021-07-15 09:...	2022-04-19 11:...	2021-07-15 09:14:32	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
✓ Show variable labels (if present)	24 0F83	14428	Smart Citizen	2021-07-15 09:...	2022-04-19 11:...	2021-07-15 09:38:21	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
Visualize numeric values	23 0E5F	14430	Smart Citizen	2021-07-15 09:...	2022-04-19 11:...	2021-07-15 09:46:54	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
Color by instance classes	22 8BF9	14431	Smart Citizen	2021-07-15 09:...	2022-04-19 11:...	2021-07-15 09:49:31	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
Selection	21 8B5A	14432	Smart Citizen	2021-07-15 09:...	2022-04-19 11:...	2021-07-15 09:53:19	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
✓ Select full rows	20 61D0	14434	Smart Citizen	2021-07-15 10:...	2022-04-19 11:...	2021-07-15 10:15:03	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	42 7F1E	14433	Smart Citizen	2021-07-15 10:...	2022-04-19 11:...	2021-07-15 10:15:39	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	20 9478	14436	Smart Citizen	2021-07-15 10:...	2022-04-19 11:...	2021-07-15 10:24:31	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	19 1273	14413	Smart Citizen	2021-07-15 07:...	2022-04-19 11:...	2021-07-15 12:30:39	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	41 63E9	14435	Smart Citizen	2021-07-15 10:...	2022-04-19 11:...	2021-07-16 02:50:53	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	18 95CD	14449	Smart Citizen	2021-07-21 14:...	2022-04-19 11:...	2021-07-21 14:36:00	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	17 8887	14454	Smart Citizen	2021-07-21 17:...	2022-04-19 11:...	2021-07-21 17:25:59	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	16 0B58	14455	Smart Citizen	2021-07-21 17:...	2022-04-19 11:...	2021-07-21 17:58:09	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	15 17FA	14456	Smart Citizen	2021-07-22 07:...	2022-04-19 11:...	2021-07-22 07:43:04	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	14 8E76	14457	Smart Citizen	2021-07-22 07:...	2022-04-19 11:...	2021-07-22 07:49:30	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	13 8B46	14458	Smart Citizen	2021-07-22 07:...	2022-04-19 11:...	2021-07-22 07:55:35	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	12 98FD	14459	Smart Citizen	2021-07-22 08:...	2022-04-19 11:...	2021-07-22 08:02:19	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	11 8EAC	14462	Smart Citizen	2021-07-22 10:...	2022-04-19 11:...	2021-07-22 10:17:09	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	10 90DA	14463	Smart Citizen	2021-07-22 10:...	2022-04-19 11:...	2021-07-22 10:48:44	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	9 4D5F	14451	Smart Citizen	2021-07-21 14:...	2022-04-19 11:...	2021-07-22 10:56:06	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	8 5081	14450	Smart Citizen	2021-07-21 14:...	2022-04-19 11:...	2021-07-22 10:56:19	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	39 8496	14460	Smart Citizen	2021-07-22 08:...	2022-04-19 11:...	2021-07-22 10:56:47	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	38 3E34	14461	Smart Citizen	2021-07-22 10:...	2022-04-19 11:...	2021-07-22 10:58:27	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	43 0B83	14466	Smart Citizen	2021-07-22 13:...	2022-04-19 11:...	2021-07-22 13:18:33	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	7 409D	14467	Smart Citizen	2021-07-22 13:...	2022-04-19 11:...	2021-07-22 13:28:18	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	6 58A3	14468	Smart Citizen	2021-07-22 13:...	2022-04-19 11:...	2021-07-22 13:40:24	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	44 6681	14473	Smart Citizen	2021-07-22 15:...	2022-04-19 11:...	2021-07-22 15:05:19	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	40 490C	14470	Smart Citizen	2021-07-22 14:...	2022-04-19 11:...	2021-10-20 20:14:43	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	5 692B	14464	Smart Citizen	2021-07-22 11:...	2022-04-19 11:...	2021-11-02 23:36:54	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	3 3AAC	14469	Smart Citizen	2021-07-22 13:...	2022-04-19 11:...	2021-11-03 10:23:52	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	2 81C_sensor2.4	14461	Smart Citizen	2021-06-06 11:...	2021-12-30-09:...	2021-12-30 09:31:28	7837	oguzcan.kemreli		[offline, toutd...	has_published	32	43.2711	2.8361	Ritbio	ES	
	4 8B7E	14474	Smart Citizen	2021-07-22 15:...	2022-04-19 11:...	2022-03-17 14:53:47	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	49 499F	14472	Smart Citizen	2021-07-22 14:...	2022-04-19 11:...	2022-04-07 21:16:52	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	50 7F1E	15283	Smart Citizen	2022-04-19 07:...	2022-04-19 11:...	2022-04-19 07:28:16	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	47 77CD	14426	Smart Citizen	2021-07-15 09:...	2022-05-02 17:...	2022-04-20 20:43:31	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	53 413F	14429	Smart Citizen	2021-07-15 09:...	2022-06-30 13:...	2022-06-17 05:35:46	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	51 811P	15282	Smart Citizen	2022-04-19 07:...	2022-07-01 06:...	2022-07-01 06:15:46	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	52 8114	15282	Smart Citizen	2022-04-19 07:...	2022-07-01 06:...	2022-07-01 06:15:46	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	34 81C_MOBILE...	15695	Smart Citizen	2022-10-06 07:...	2022-10-06 12:24:59	6098	roberto	[MakingSense...		[offline, toutd...	has_published	32	56.1555	10.1873	Aarhus	DK	
	56 81C_MOBILE...	15652	Smart Citizen	2022-09-17 06:...	2022-10-04 12:...	2022-10-10 10:02:50	6098	roberto		[offline, toutd...	has_published	32	56.1561	10.1876	Aarhus	DK	
	55 0CND0546V0R...	16003	Smart Citizen	2023-01-24 16:...	2023-01-24 16:...	2023-01-24 17:18:45	8565	joanna747	[Research, 'Ex...		[offline, toutd...	has_published	32	59.9087	10.9259	Oslo	NO
	58 DAVIDEESTIN...	15822	Smart Citizen	2022-11-16 15:...	2023-02-09 15:...	2023-02-09 15:21:08	8512	d_prieto	[Barcelona, 'R...	[indoor, 'offli...	has_published	32	41.3988	2.15427	Barcelona	ES	
	57 525F	14465	Smart Citizen	2021-07-22 13:...	2023-03-09 07:...	2023-03-09 14:28:54	7761	Battisrob		[offline, toutd...	has_published	32	44.495	11.3525	Bologna	IT	
	54 8480V02V0CHU...	16161	Smart Citizen	2023-03-27 13:...	2023-03-29 13:...	2023-06-05 09:20:17	8636	Michael Lalam		[offline, toutd...	has_published	32	50.0775	14.4449	Prague	CZ	

Figura 4. Data Table y selección

En caso de no tener un *kit*, se puede usar un *kit* de ejemplo, usando en *Device ID* el número 14471. Tras esto, pulsa *Search Devices*.

Una vez encontrado el *kit*, pasamos a descargar los datos del mismo. Para ello, usa otro *widget*: *Smart Citizen Data*. Ahora, haz click sobre el *widget Smart Citizen Data*. En el caso de los datos de ejemplo debería mostrar algo similar a la *Figura 5*.

Info

Device 14471 data downloaded!

Get data at a specific frequency

Rollup:

Rollup units:

Filter by date (YYYY-MM-DD)

Initial Date:

End Date:

Resample data

? | 1

Figura 5. Panel de configuración de Smart Citizen Data

Si aparece información en la parte superior, es el momento de seleccionar qué datos queremos descargar. En este ejemplo, no introduciremos ninguna fecha, pero estableceremos la frecuencia de los datos. Para ello, indica en “Rollup:” 1 y en “Rollup units:” en m (*m* para *minutos*, *s* para *segundos*, *ms* para *milisegundos*, *h* para *horas*, *d* para *días*).

La opción de *Resample data* hará un promedio en caso de que haya datos incompletos. En este caso no usaremos esta opción. Por último, selecciona *Get data* para obtener los datos. Tras unos segundos, en la parte inferior aparecerá un mensaje como: *Device XXXXX data downloaded!* donde XXXXX es el número del kit que se ha insertado (en el caso del ejemplo, 14417).

2. En el panel de “Geo” hay que hacer clic en *Geo Map* (Figura 6) para crear nodo *Geo Map* donde representar los datos geolocalizados.

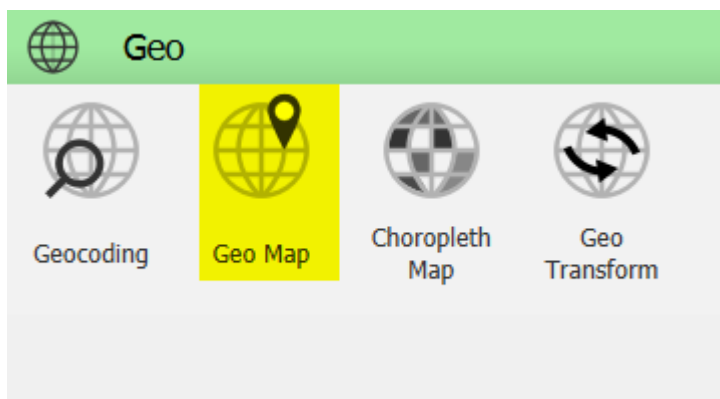


Figura 6. Panel de Geo

Como se ha hecho antes, hay que conectar la tabla de datos con el *Geo Map* arrastrando desde la parte derecha de Smart citizen data o de la tabla (output) a la parte izquierda del *Geo Map* (el input). Hay que clicar en el *Geo Map* y se verá algo como la *Figura 7* en el caso de que el set de datos tenga datos geolocalizados.

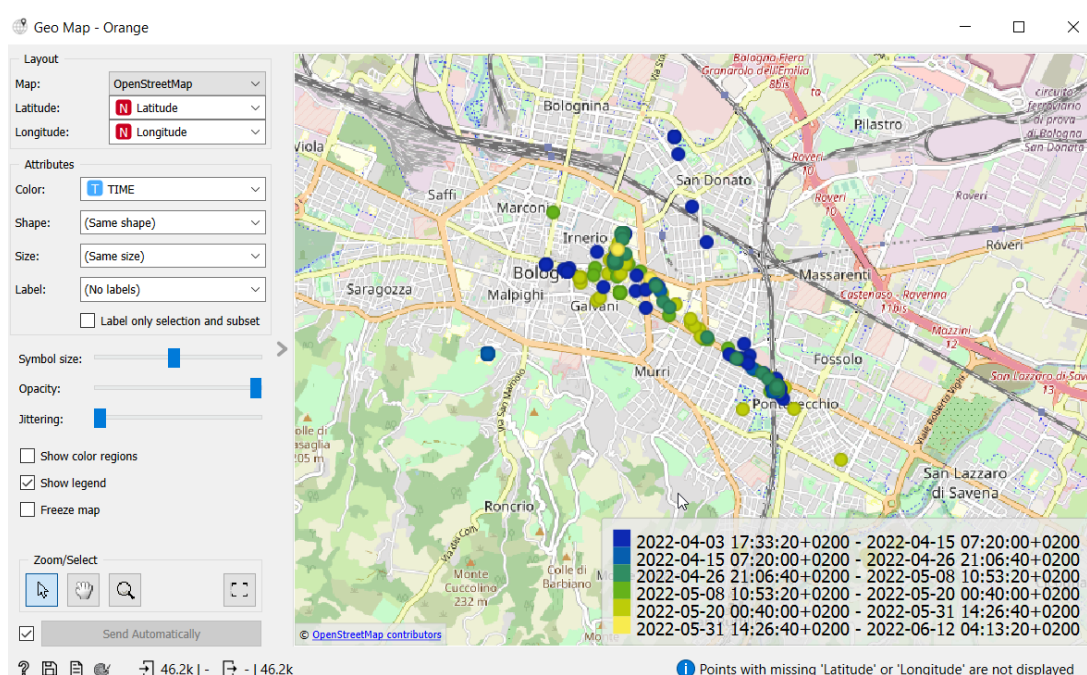


Figura 7. Interior de un Geo Map

Se pueden ajustar colores, tamaños y parámetros utilizando los atributos en la parte izquierda de la interfaz. Para cambiar las paletas de colores sencillamente se usa el nodo “color” como intermediario entre *Geo Map* y la tabla de datos.

Nota: Los datos que no tengan latitud o longitud son ignorados por defecto.

Processing

Configuración de Processing para su uso con el Smart Citizen Kit

Introducción

Este tutorial sirve para configurar *Processing* para utilizar el kit. Es el mismo procedimiento que hay que realizar para cualquier interacción de *Processing* con serial, con lo que sirve también para hablar con *Arduino*.

Este tutorial se puede encontrar en inglés en la documentación oficial¹⁶:

Requisitos

- **Software:** los tutoriales indicados a continuación requieren de la instalación previa de *Processing*, el cual se puede encontrar en la web del proyecto¹⁷. Esta instalación es posible en varias plataformas y tiene una duración estimada de 15 minutos.
- **Materiales:** los tutoriales indicados a continuación requieren de un ordenador con puerto USB, un cable USB con terminación USB-micro y un Smart Citizen Kit.

Tiempo estimado

10 minutos

Descripción detallada

1. Abre un nuevo sketch en blanco y escribe el siguiente código:

```
// Example by Tom Igoe
import processing.serial.*;

// The serial port
Serial myPort;

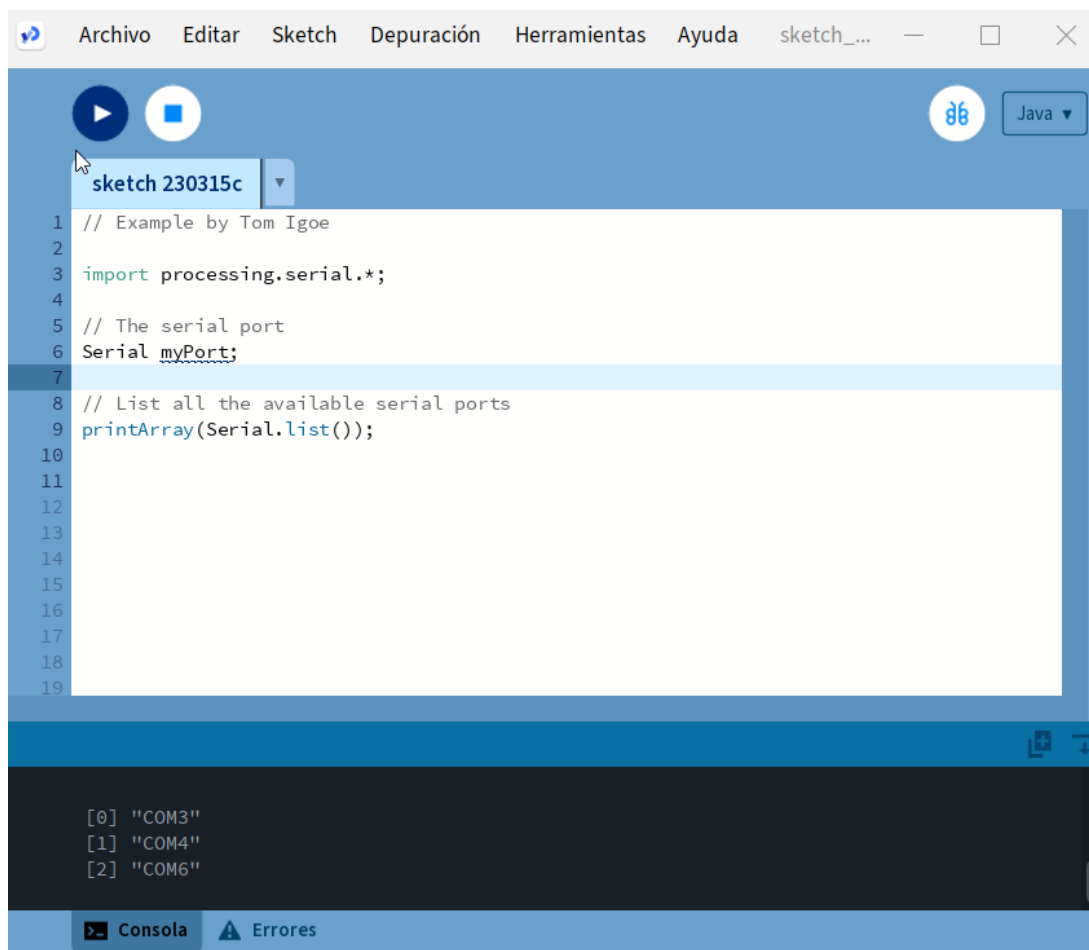
// List all the available serial ports
printArray(Serial.list());
```

Al ejecutarlo aparecerán en la consola los puertos disponibles. En *Windows* estos puertos tendrán nombres como “COM2” y en *MAC* o *Linux* esta nomenclatura es diferente (*Figura 1*).

¹⁶ Documentación oficial:

<https://docs.smartcitizen.me/Resources/Tutorials/Setting-up-Processing-and-using-Serial/>

¹⁷ Web del proyecto Processing <https://processing.org/download>



```
1 // Example by Tom Igoe
2
3 import processing.serial.*;
4
5 // The serial port
6 Serial myPort;
7
8 // List all the available serial ports
9 printArray(Serial.list());
10
11
12
13
14
15
16
17
18
19
```

```
[0] "COM3"
[1] "COM4"
[2] "COM6"
```

Figura 1. Sketch de processing 1

Por ejemplo:

```
[0] "COM3"
[1] "COM4"
[2] "COM6"
```

En los próximos ejemplos, usaremos uno de estos para conectar con el *Smart Citizen Kit*. En los diferentes códigos de ejemplo, hay que revisar cual es el índice correcto (el número entre corchetes) que se refiere al puerto al que está conectado el *Smart Citizen Kit*. Para comprobar cuál es, basta comparar los puertos que salen con el kit conectado al ordenador con respecto a los que salen sin haberlo conectado.

Animación de Processing con datos en tiempo real

Introducción

Este tutorial explica cómo crear una animación para visualizar la cantidad de luz que recibe el *Smart Citizen Kit* en tiempo real.

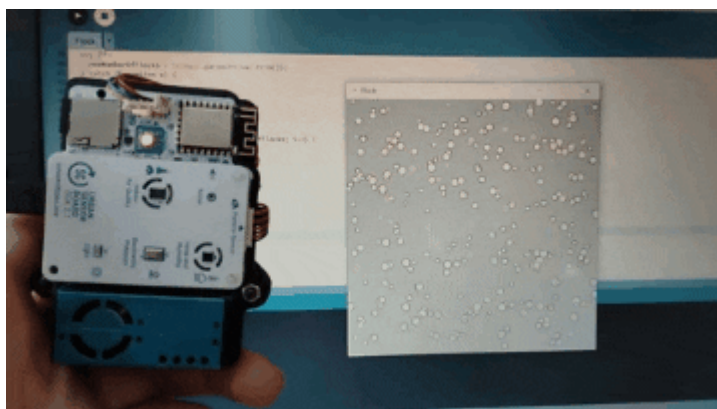


Figura 1. Sketch de processing 2

Este tutorial es fácilmente adaptable a otras métricas.

Este tutorial se puede encontrar en inglés en la documentación oficial¹⁸.

Requisitos

- **Software:** los tutoriales indicados a continuación requieren de la instalación previa de *Processing*, el cual se puede encontrar en la web del proyecto¹⁹. Esta instalación es posible en varias plataformas y tiene una duración estimada de 15 minutos. Descargar el código del repositorio https://github.com/fablabbcn/smartcitizen-docs/blob/master/docs/assets/pde/Flocks_example_serial/Flocks_example_serial.pde
- **Materiales:** los tutoriales indicados a continuación requieren de un ordenador con puerto USB, un cable USB con terminación USB-micro y un Smart Citizen Kit.

Descripción detallada

1. Una vez averiguado el número del puerto siguiendo el tutorial anterior, se usará ese valor en la variable `PORT_INDEX` (*Figura 2*).

¹⁸ Documentación oficial:

<https://docs.smartcitizen.me/Resources/Tutorials/Setting-up-Processing-and-using-Serial/#example-of-use-the-flock-of-light>

¹⁹ Web del proyecto Processing <https://processing.org/download>

```

Flock ▼
1 import processing.serial.*;
2
3 Flocker[] flock = new Flocker[10000];
4 int numberOfFlocks = 3;
5
6 //Serial stuff
7 Serial smartCitizenKitSerial; // Create object from Serial class
8 String val; // Data received from the serial port
9 int PORT_INDEX = 2; // change for the number of the port
10
11 void setup() {

```

Figura 2

Nota: Si al ejecutar el sketc aparece un error, es probable que el número del puerto no sea correcto. Si el error indica “port busy”, es debido a que otra aplicación (como *Arduino IDE*) está utilizando el puerto en este momento.

Si todo es correcto, al ejecutar el código debería aparecer ventana con una animación representando unos círculos, cuyo número es proporcional al número de *luxes* que recibe el sensor de luz del Smart Citizen Kit. Se puede comprobar su funcionamiento pasando la mano por delante del sensor y observando cómo disminuyen los círculos.

Comentarios sobre el funcionamiento del código

La primera función importante es *setup*, que se ejecuta solo una vez cada vez que se ejecuta el programa y que se ve cómo se inicia el serial con estas instrucciones:

```

void setup() {
  String portName = Serial.list()[PORT_INDEX];
  smartCitizenKitSerial = new Serial(this, portName, 115200);
}

```

Aquí se establece el nombre del puerto y se introduce como variable dentro de la lista de puertos disponibles. Se instancia un objeto de la clase *Serial* en la variable *smartCitizenSerial*. El constructor necesita el objeto en sí mismo (escrito usando la palabra reservada “*this*”), el nombre del puerto y la frecuencia en baudios en número (115200).

Para más información existen otros tutoriales que van en más detalle sobre la relación de *Processing* y *serial* (ver notas al pie²⁰).

Posteriormente, enviamos un comando a través de la consola (*shell*) del Smart Citizen Kit, el cual nos devolverá datos en tiempo real. El comando es: `monitor light -notime -noms`

Se puede visitar más referencia en el tutorial de uso de la consola en arduino en esta documentación y también en este enlace: <https://docs.smartcitizen.me/Guides/getting%20started/Using%20the%20Shell/>

Después el código revisa los datos en este fragmento:

```
//Serial reading
if ( smartCitizenKitSerial.available() > 0){
  // If data is available,
  val = smartCitizenKitSerial.readStringUntil('\n');
  // read it and store it in val
  println(val);
}
```

En este caso, se revisa si el objeto tiene disponible información y se almacena en una variable (de tipo *string*). Finalmente, se integra con la lectura del sensor con la animación, cambiando el número de círculos (Flocks):

```
Integer newNumberOfFlocks = numberOfFlocks;

try {
  newNumberOfFlocks = Integer.parseInt(val.trim());
  if (newNumberOfFlocks < 0){
    newNumberOfFlocks = 1;
  }
} catch (Exception e) {
  //do nothing
}
```

Este sistema de *try/catch* permite procesar elementos que podrían resultar en error. Por ejemplo, en caso de que a través del serial recibiéramos un valor no numérico, el número de círculos quedaría inalterado.

²⁰ Tutoriales adicionales para processing (Sparkfun)

<https://learn.sparkfun.com/tutorials/connecting-arduino-to-processing/all> y oficial

<https://processing.org/reference/libraries/serial/index.html>

Recursos adicionales

Este listado se puede consultar en el siguiente enlace:

<https://docs.smartcitizen.me/Resources/Education/CLAB/>

Configuración del kit

- Cómo comenzar con el kit (onboarding):
<https://docs.smartcitizen.me/Guides/getting%20started/Onboarding%20Sensors/>
- Cómo usar el terminal del Smart citizen kit::
<https://docs.smartcitizen.me/Guides/getting%20started/Using%20the%20Shell/>
- Cómo hacer debugging del Smart Citizen kit:
<https://docs.smartcitizen.me/Guides/getting%20started/Debugging%20your%20sensors/>
- Obtención de datos en formato CSV :
<https://docs.smartcitizen.me/Guides/getting%20started/Downloading%20the%20Data/>
- Subir información de la tarjeta SD:
<https://docs.smartcitizen.me/Guides/getting%20started/Uploading%20SD%20Card%20Data/>
- Actualizar configuración de red:
<https://docs.smartcitizen.me/Guides/getting%20started/Updating%20the%20Wi-Fi/>

Datos

- Widgets de Orange:
<https://docs.smartcitizen.me/Guides/data/Orange%20Data%20Widgets/>
- Llamada de API a través de Processing:
<https://docs.smartcitizen.me/Resources/Tutorials/How-to-call-the-Smart-Citizen-API-using-Processing/>

Otras referencias de educación

- Proyecto DOIT. Midiendo nuestro entorno:
https://toolboxadmin.doit-europe.net/sites/default/files/2020-05/10_Measuring%20Our%20Environment_DOIT-Manual_espanol.pdf
- Aulamar: técnicas de muestreo de agua: [AULAMAR - Técnicas de Muestreo de agua \(ES\)](#)

Proyectos externos

- Proyecto atenció:
<https://sites.google.com/view/projecteatencio/material-did%C3%A0ctic-19-20/m3-com-%C3%A9s-la-qualitat-de-laire-de-linstitut?authuser=0>
- XTEC:
<https://sites.google.com/view/projecteatencio/material-did%C3%A0ctic-19-20/m3-com-%C3%A9s-la-qualitat-de-laire-de-linstitut?authuser=0>
- Smart Citizen User Guide Bibliolab Science and Healthy City:
https://docs.smartcitizen.me/assets/publications/Smart_Citizen_Kit_User_Guide_Bibliolab_Science_and_Healthy_City.pdf
- A Citizen's Guide To Understanding and Monitoring Lakes and Streams:
<https://apps.ecology.wa.gov/publications/documents/94149.pdf>
- Air quality vs motorway traffic: https://github.com/cooscoos/air_qual