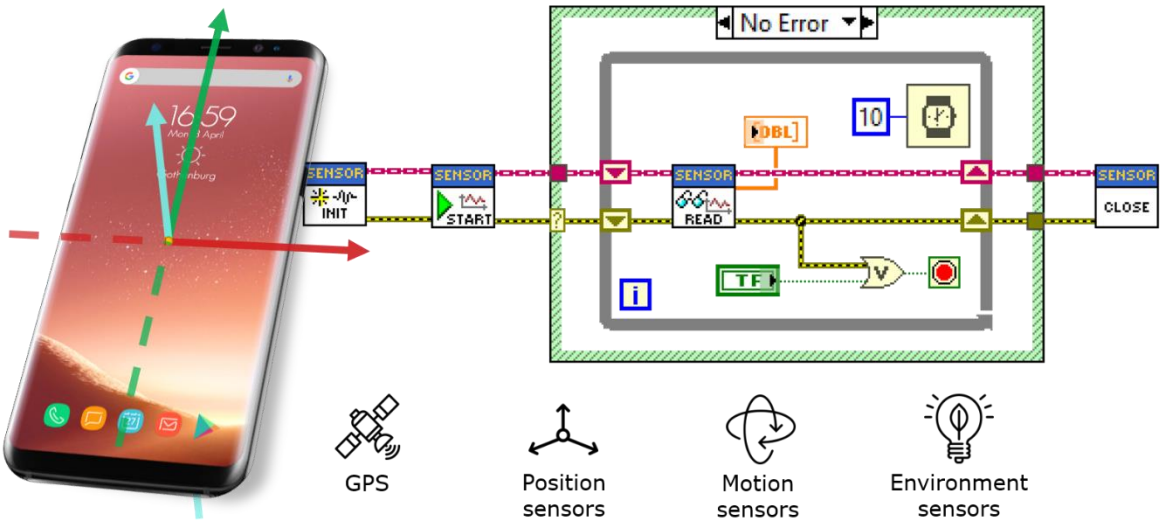
wireflow

AC0091-001, rev A

WF Smartphone Sensor Toolkit User Manual



wireflow

Contents

Support information 2
Technical support and Product information 2
WireFlow headquarters 2
Important information 2
Copyright 2
EULA 3
Introduction 7
Supported sensors 7
Download & Install 10
Requirements 10
Install VI Package 10
Install USB driver 10
Install Android Application 12
Toolkit VIs 13
Init.vi 13
Close.vi 13
GetConnectionStatus.vi 14
StartSensor.vi 14
StartMultipleSensors.vi 14
StopSensor.vi 14
StopMultipleSensors.vi 15
ReadSensor.vi 15
Basic usage/Examples 16
Quickstart 16
Using the toolkit 18
Error codes 20
Technical support and FAQ 20
FAQ 20
Device not listed as a NI-VISA USB Device? 21
No USB Driver for your device? 25
Device not showing at all in Windows Device Manager? 27
References 28

Support information

Technical support and Product information

https://www.wireflow.se/product/wf-smartphone-sensor-toolkit/
support@wireflow.se

WireFlow headquarters

WireFlow AB
Theres Svenssons gata 10
SE-417 55 Göteborg
Sweden

Important information

Copyright

EULA

END-USER LICENSE AGREEMENT FOR
Wireflow Smartphone Sensor Toolkit (AC0091)

IMPORTANT PLEASE READ THE TERMS AND CONDITIONS OF THIS LICENSE AGREEMENT CAREFULLY BEFORE CONTINUING WITH THIS PROGRAM DOWNLOAD/INSTALL: WireFlow's End-User License Agreement ("EULA") is a legal agreement between you (either an individual or a single entity) and WireFlow, for the WireFlow software product(s) identified above which may include associated software components, media, printed materials, and "online" or electronic documentation ("SOFTWARE PRODUCT"). By installing, copying, or otherwise using the SOFTWARE PRODUCT, you agree to be bound by the terms of this EULA. This license agreement represents the entire agreement concerning the program between you and WireFlow, (referred to as "licenser"), and it supersedes any prior proposal, representation, or understanding between the parties. If you do not agree to the terms of this EULA, do not download, install or use the SOFTWARE PRODUCT.

The SOFTWARE PRODUCT is protected by copyright laws and international copyright treaties, as well as other intellectual property laws and treaties. The SOFTWARE PRODUCT is licensed, not sold.

1. GRANT OF LICENSE

The SOFTWARE PRODUCT is licensed as follows:

1.1 Installation and Use

WireFlow grants you a personal, non-transferable and non-exclusive right to use the copy of the Software provided with this EULA on your computer running a validly licensed copy of the operating system for which the SOFTWARE PRODUCT was designed.

1.2 Backup Copies

You may also make copies of the SOFTWARE PRODUCT as may be necessary for backup and archival purposes.

1.3 Evaluation Version

For clarity in the case of Trial Licenses, if You do not pay the applicable license fees prior to the conclusion of any applicable Trial Period, you have no right or license, express or implied, to further use the SOFTWARE PRODUCT in any manner thereafter.

1.4 Academic Version

For clarity in the case of Academic Licenses, if You do not pay the applicable license fees prior to the conclusion of any licensing Period, you have no right or

license, express or implied, to further use the SOFTWARE PRODUCT in any manner thereafter. The SOFTWARE PRODUCT licensed as an Academic License may not be used for any commercial purposes.

1.5 Home Version

The SOFTWARE PRODUCT licensed as a Home License may not be used for any commercial purposes.

2. DESCRIPTION OF OTHER RIGHTS AND LIMITATIONS

2.1 Maintenance of Copyright Notices

You must not remove or alter any copyright notices on any and all copies of the SOFTWARE PRODUCT.

2.2 Distribution

You may not distribute registered copies of the SOFTWARE PRODUCT to third parties. Evaluation versions available for download from WireFlow's websites may be freely distributed.

2.3 Prohibition on Reverse Engineering, Decompilation, and Disassembly

You may not reverse engineer, decompile, or disassemble the SOFTWARE PRODUCT, except and only to the extent that such activity is expressly permitted by applicable law notwithstanding this limitation.

2.4 Rental

You may not rent, lease, or lend the SOFTWARE PRODUCT.

2.5 Support Services

WireFlow may provide you with support services related to the SOFTWARE PRODUCT ("Support Services"). Any supplemental software code provided to you as part of the Support Services shall be considered part of the SOFTWARE PRODUCT and subject to the terms and conditions of this EULA.

2.6 Compliance with Applicable Laws

You must comply with all applicable laws regarding use of the SOFTWARE PRODUCT.

2.7 Export Laws

The export of the SOFTWARE PRODUCT from the country of original purchase may be subject to control or restriction by applicable local law. Licensee is solely responsible for determining the existence and application of any such law to any proposed export and for obtaining any needed authorization. Licensee agrees not to export the SOFTWARE PRODUCT from any country in violation of applicable legal restrictions on such export.

3. TERMINATION

Without prejudice to any other rights, WireFlow may terminate this EULA if you fail to comply with the terms and conditions of this EULA. In such event, you must destroy all copies of the SOFTWARE PRODUCT in your possession.

4. COPYRIGHT

All title, including but not limited to copyrights, in and to the SOFTWARE PRODUCT and any copies thereof are owned by WireFlow or its suppliers. All title and intellectual property rights in and to the content which may be accessed through use of the SOFTWARE PRODUCT is the property of the respective content owner and may be protected by applicable copyright or other intellectual property laws and treaties. This EULA grants you no rights to use such content. All rights not expressly granted are reserved by WireFlow.

4.1 Third party software.

The SOFTWARE PRODUCT may include software under license from third parties (“Third Party Software” and “Third Party License”). Any Third Party Software is licensed to you subject to the terms and conditions of the corresponding Third Party License. Generally, the Third Party License is located in a separate file such as license.txt or a readme file.

5. NO WARRANTIES

WireFlow expressly disclaims any warranty for the SOFTWARE PRODUCT. The SOFTWARE PRODUCT is provided 'As Is' without any express or implied warranty of any kind, including but not limited to any warranties of merchantability, noninfringement, or fitness of a particular purpose. WireFlow does not warrant or assume responsibility for the accuracy or completeness of any information, text, graphics, links or other items contained within the SOFTWARE PRODUCT. WireFlow makes no warranties respecting any harm that may be caused by the transmission of a computer virus, worm, time bomb, logic bomb, or other such computer program. WireFlow further expressly disclaims any warranty or representation to Authorized Users or to any third party.

6. HIGH RISK ACTIVITIES

The SOFTWARE PRODUCT is not fault-tolerant and is not designed, manufactured or intended for use or resale as on-line control equipment in hazardous environments requiring fail-safe performance, such as in the operation of nuclear facilities, aircraft navigation or communication systems, air traffic control, direct life support machines, or weapons systems, in which the failure of the SOFTWARE PRODUCT could lead directly to death, personal injury, or severe physical or environmental damage ("High Risk Activities"). WireFlow and its suppliers specifically disclaim any express or implied warranty of fitness for High Risk Activities.

7. LIMITATION OF LIABILITY

In no event shall WireFlow be liable for any damages (including, without limitation, lost profits, business interruption, or lost information) rising out of

'Authorized Users' use of or inability to use the SOFTWARE PRODUCT, even if WireFlow has been advised of the possibility of such damages. In no event will WireFlow be liable for loss of data or for indirect, special, incidental, consequential (including lost profit), or other damages based in contract, tort or otherwise. WireFlow shall have no liability with respect to the content of the SOFTWARE PRODUCT or any part thereof, including but not limited to errors or omissions contained therein, libel, infringements of rights of publicity, privacy, trademark rights, business interruption, personal injury, loss of privacy, moral rights or the disclosure of confidential information.

8. CONTACT

All questions about this EULA shall be directed to: info@wireflow.se.
WireFlow AB
Theres Svenssons gata 10
SE-417 55 Göteborg
Sweden

Introduction

The WireFlow Smartphone Sensor Toolkit is a LabVIEW add-on that enables Android smartphone sensor data to be acquired and utilized in LabVIEW. The sensor data that can be acquired depends on which smartphone is used, since different smartphone models have different sensors.

Supported sensors

The smartphones coordinate system is defined according to the following image.

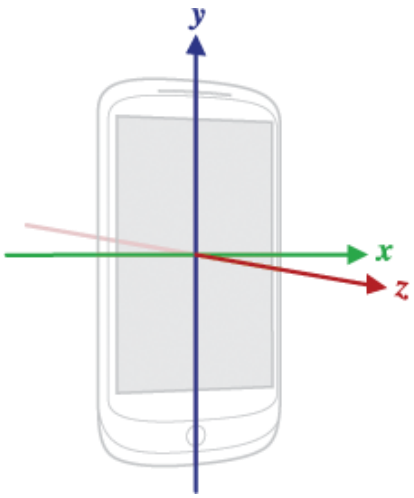


Figure 1. Smartphone coordinate system according to Android. Source: Android [1]

The toolkit currently offers 15 different sensors to choose from. These are listed in the table below.

Table 1. Supported sensors, axis corresponds to the axis in Figure 1.

Sensor	Data out	Description	Units of measure
Accelerometer	ReadData[0]	Acceleration force along X axis	m/s^2
	ReadData[1]	Acceleration force along Y axis	
	ReadData[2]	Acceleration force along Z axis	
Linear acceleration	ReadData[0]	Acceleration force along X axis (excl. gravity)	m/s^2
	ReadData[1]	Acceleration force along Y axis (excl. gravity)	
	ReadData[2]	Acceleration force along X axis (excl. gravity)	
Gyroscope	ReadData[0]	Rate of rotation around X axis	rad/s
	ReadData[1]	Rate of rotation around Y axis	
	ReadData[2]	Rate of rotation around Z axis	
Gyroscope uncalibrated	ReadData[0]	Rate of rotation around X axis (without drift compensation)	rad/s
	ReadData[1]	Rate of rotation around Y axis	

		(without drift compensation)	
	ReadData[2]	Rate of rotation around Z axis (without drift compensation)	
	ReadData[3]	Estimated drift around X axis	
	ReadData[4]	Estimated drift around Y axis	
	ReadData[5]	Estimated drift around Z axis	
Rotation vector	ReadData[0]	Rotation vector component along X axis (X*sin(Θ/2))	unitless
	ReadData[1]	Rotation vector component along Y axis (Y*sin(Θ/2))	
	ReadData[2]	Rotation vector component along Z axis (Z*sin(Θ/2))	
Gravity	ReadData[0]	Force of gravity along X axis	m/s^2
	ReadData[1]	Force of gravity along X axis	
	ReadData[2]	Force of gravity along X axis	
Magnetic field	ReadData[0]	Geomagnetic field strength along X axis	µT
	ReadData[1]	Geomagnetic field strength along Y axis	
	ReadData[2]	Geomagnetic field strength along Z axis	
Magnetic field uncalibrated	ReadData[0]	Geomagnetic field strength (without hard iron calibration) along X axis	µT
	ReadData[1]	Geomagnetic field strength (without hard iron calibration) along Y axis	
	ReadData[2]	Geomagnetic field strength (without hard iron calibration) along Z axis	
	ReadData[3]	Iron bias estimation along X axis	
	ReadData[4]	Iron bias estimation along Y axis	
	ReadData[5]	Iron bias estimation along Z axis	
Game rotation vector[1]	ReadData[0]	Rotation vector component along X axis (X*sin(θ/2))	unitless
	ReadData[1]	Rotation vector component along Y axis (Y*sin(θ/2))	
	ReadData[2]	Rotation vector component along Z axis (Z*sin(θ/2))	
Geomagnetic rotation vector[2]	ReadData[0]	Rotation vector component along X axis (X*sin(θ/2))	unitless
	ReadData[1]	Rotation vector component along Y axis (Y*sin(θ/2))	
	ReadData[2]	Rotation vector component along Z axis (Z*sin(θ/2))	
Proximity[3]	ReadData[0]	Distance from object	cm
Ambient	ReadData[0]	Ambient air temperature	°C

temperature			
Light	ReadData[0]	Illuminance	lx
Pressure	ReadData[0]	Ambient air pressure	hPa or mbar
Relative humidity	ReadData[0]	Ambient relative humidity	%
GPS position	ReadData[0]	Lattitude	° (degrees)
	ReadData[1]	Longitude	° (degrees)
	ReadData[2]	Altitude	m (meters)
GPS time[4]	ReadData[0]	ms since 1:00 AM, 1 Jan 1970	Unix time

[1] The game rotation vector sensor is identical to the rotation vector sensor, except it does not use the geomagnetic field. Therefore the Y axis does not point north, but instead to some other reference. That reference is allowed to drift by the same order of magnitude as the gyroscope drifts around the Z axis.

[2] The geomagnetic rotation vector sensor is similar to the rotation vector sensor, but it uses a magnetometer instead of a gyroscope. The accuracy of this sensor is lower than the normal rotation vector sensor, but the power consumption is reduced.

[3]Some proximity sensors provide only binary values representing near and far.

[4]To convert to LabVIEW time, multiply output value value with 0.001 and add with timestamp from 1:00 AM 1 Jan 1970, then convert calculated value to timestamp. Look in example ReadFromMultipleSensors.vi to see how it's done.

Download & Install

The following items need to be downloaded and installed before using the toolkit.

1. VI Package: At WireFlow webpage.
2. WF Smartphone Sensor Toolkit Android application: Google Play Store
3. Sensor Toolkit USB driver for other phones than the ones included in the build: At WireFlow webpage

Requirements

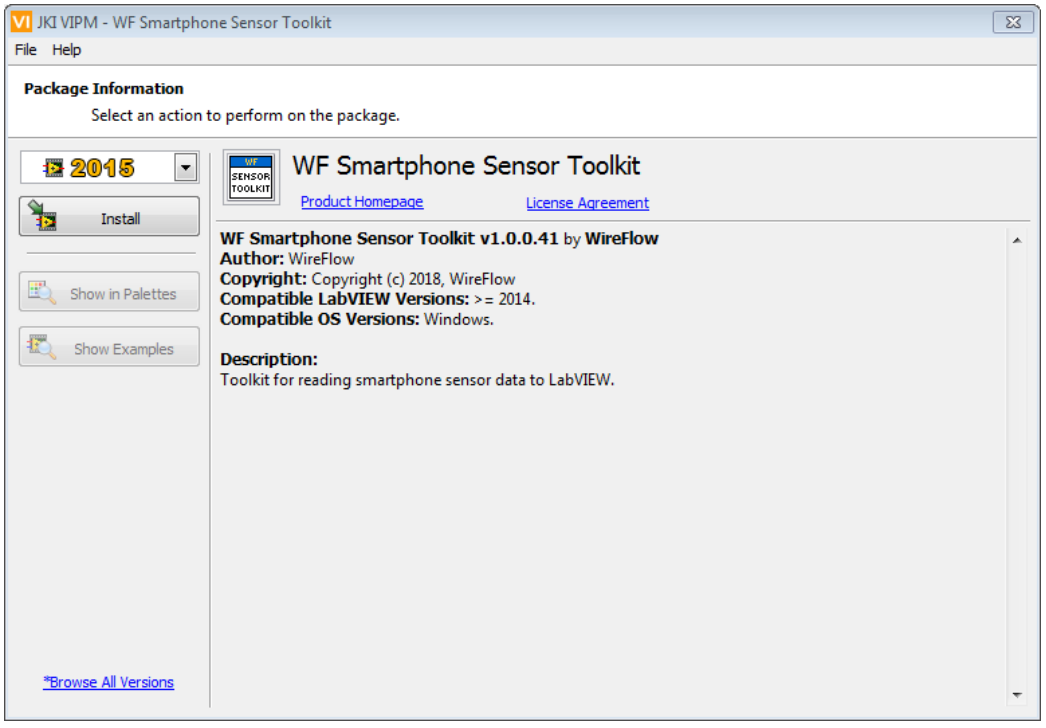
The toolkit requires the following software to be used

- LabVIEW version 2014 or later
- NI-VISA with USB passport support

In addition to the required software an Android based Smartphone is required to get sensor data.

Install VI Package

The VI package requires the program VI Package Manager (VIPM) from JKI to be installed. Once this is installed, just double-click the .vip-file and installation will be opened in VIPM.



Follow the instructions to complete the installation.

Install USB driver

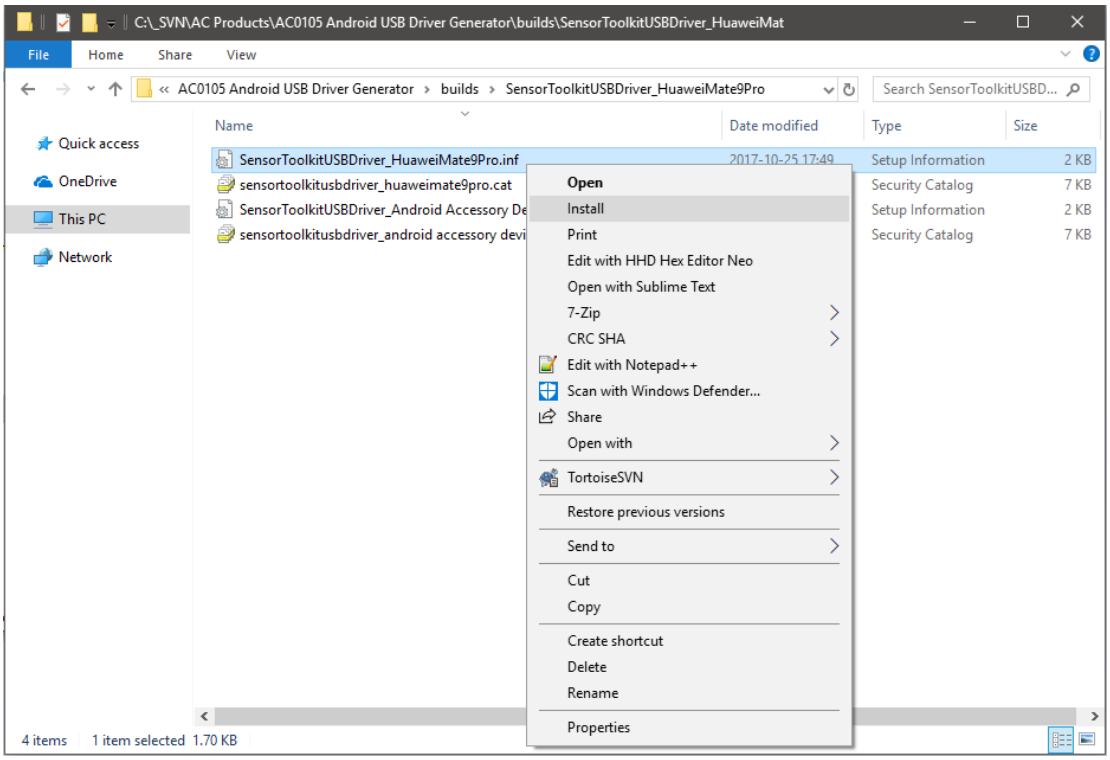
The driver includes a basic set of Smartphone drivers that can be found in the LabVIEW folder

<LabVIEW>\vi.lib\addons\WireFlow_WF Smartphone Sensor Toolkit\Drivers

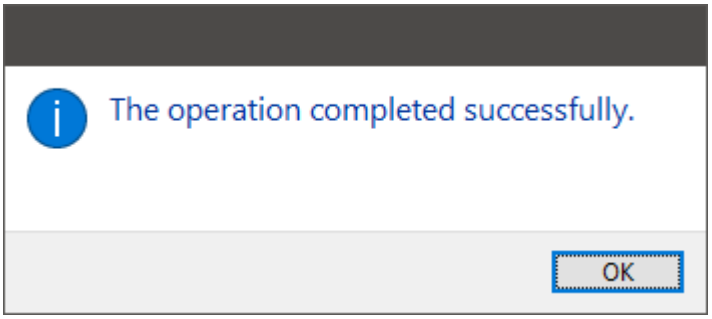
The zipped driver (included or downloaded from the WireFlow webpage) contains multiple vendor/model specific drivers. (If your Android smartphone isn't listed among the drivers, please read and follow the instructions in: No USB Driver for your device?.)
Extract and open the folder and locate the inf-file (vendor + model) corresponding to your smartphone. Also locate the generic Android Accessory device driver.

- SensorToolkitUSBDriver_*Vendor/ModelSpecificName*.inf
- SensorToolkitUSBDriver_AndroidAccessoryDevice.inf

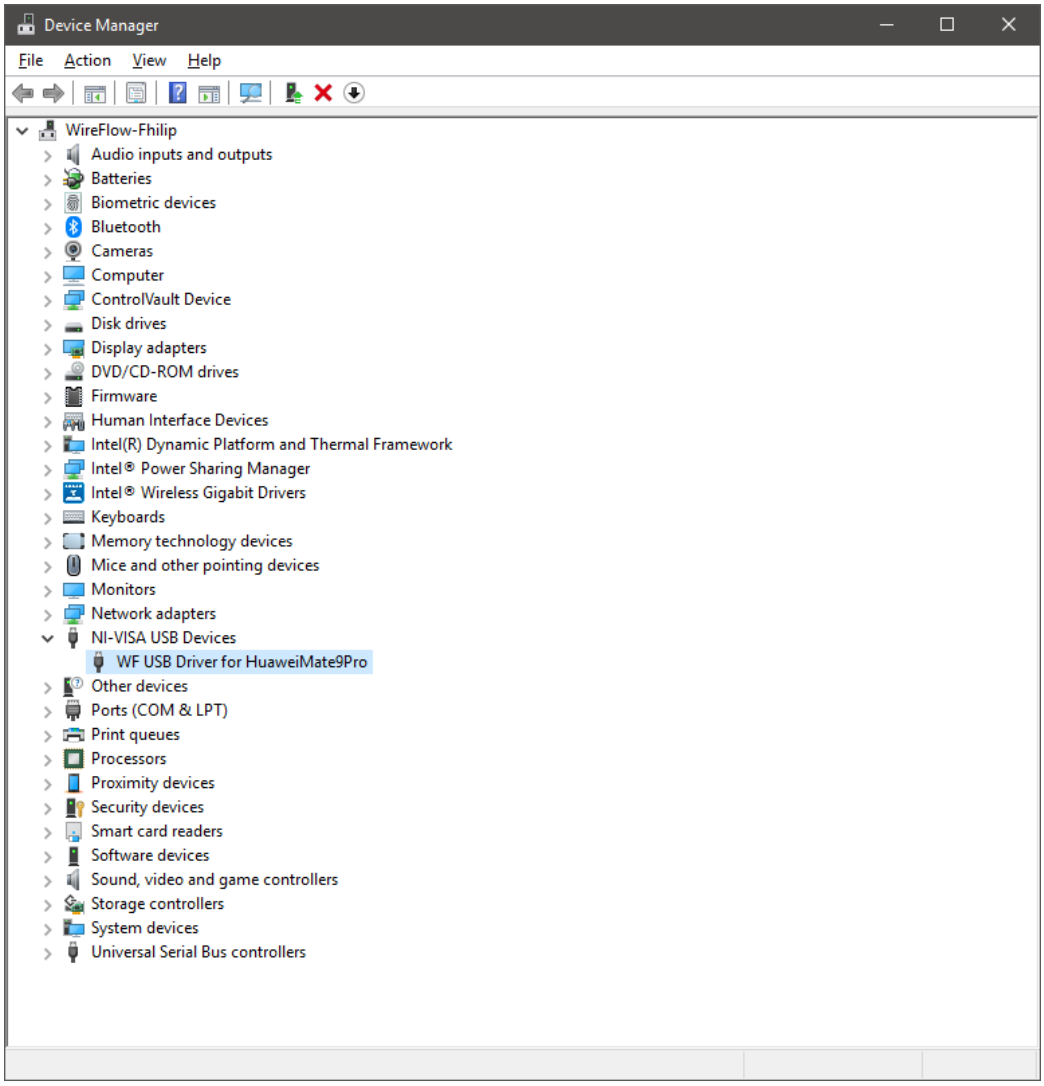
Right-click and choose "Install" for both drivers.



If the installation is successful, the following window will appear:



Try connecting your device now to verify that it works. Choose to connect as an MTP Device from your smartphone. Verify that your device enumerates correctly in Windows by opening Windows Device Manager where your device should be listed under NI-VISA USB Devices, or by opening NI Measurement & Automation Explorer where your device should be listed as an USB device under Devices and Interfaces.



If your device isn't listed as a NI-VISA USB Device after performing these steps, please see Device not listed as a NI-VISA USB Device?.

Install Android Application

The WF Smartphone Sensor Toolkit Android application can be downloaded and installed from within the Google Play Store in your Android smartphone.

Link to app: Google Play Store
Or search in the Google Play Store for "WireFlow Smartphone Sensor Toolkit".

Toolkit VIs

The toolkit consists of 8 VIs which can be seen in the figure below.

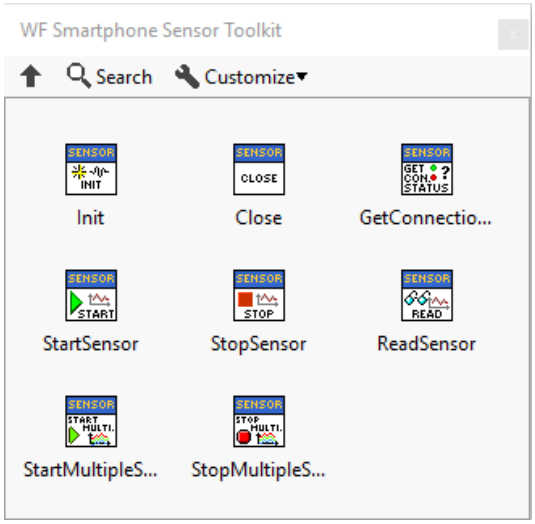
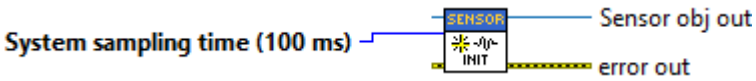


Figure 2. WF Smartphone Sensor Toolkit VIs.

Each VI is described more in detail below.

Init.vi

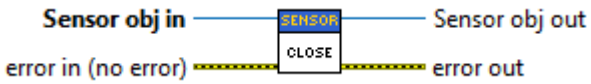
Initializes the class object and starts the asynchronous background process that communicates between the smartphone and the LabVIEW API.



System sampling time (100ms) sets the sampling time for all sensors the user choose to start.

Close.vi

This VI sends a message to the smartphone, telling it to stop all sensors that have been previously started. After sending the message, the VI terminates the background process which closes the connection with the smartphone. Finally it closes all open references, releases all notifiers and destroy all user events.



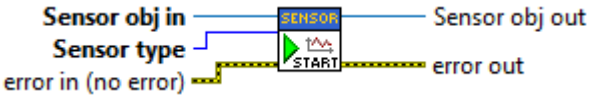
GetConnectionStatus.vi

Returns the connection status between the smartphone and the Sensor Toolkit.

Connection Status states:
INIT - Initializing driver.
NO DEVICE - Not connected to smartphone.
CONNECTED - Ready to send commands to and read data from smartphone.

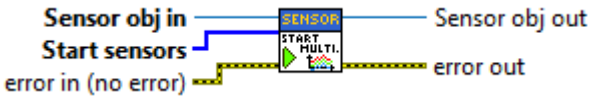
StartSensor.vi

Sends a message to the smartphone to start sending the selected **Sensor type**'s values to LabVIEW. Recieving the values requires the use of the **ReadSensor.vi** with the same **Sensor type** as input parameter.



StartMultipleSensors.vi

Starts multiple sensors, for more details see StartSensor.vi documentation.

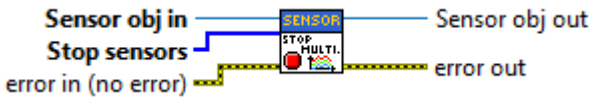


StopSensor.vi

Sends a message to the smartphone that stops the selected **Sensor type**.

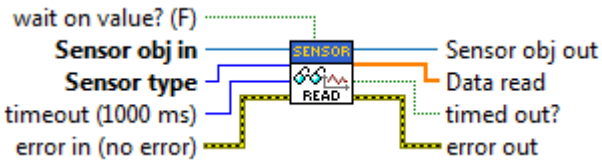
StopMultipleSensors.vi

Stops multiple sensors, for more details see StopSensor.vi documentation.



ReadSensor.vi

Reads the selected sensors value upon notification triggered by the background process. Depending on chosen sensor, the output **Data read** might be represented in array sizes between 1-6.



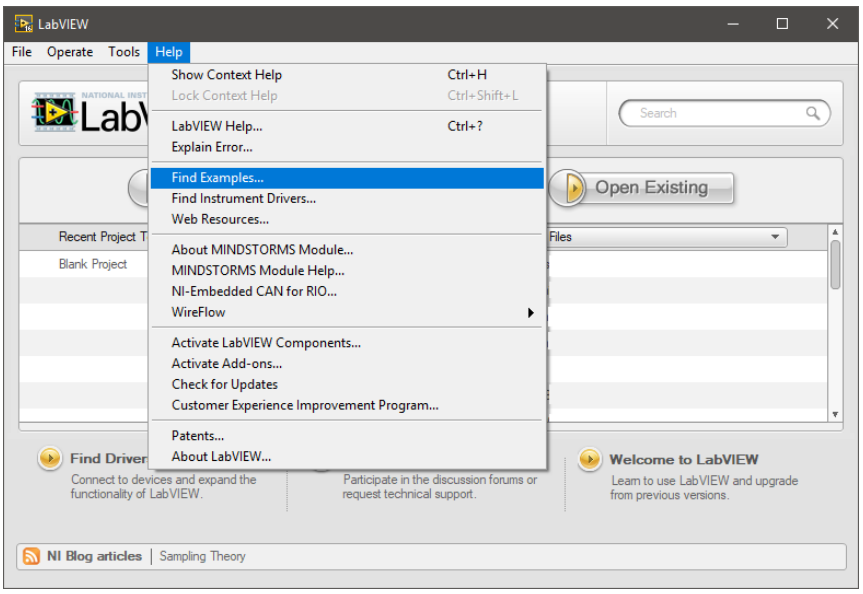
timeout (100 ms) determines how long it's acceptable to wait for a reading to be received.

Basic usage/Examples

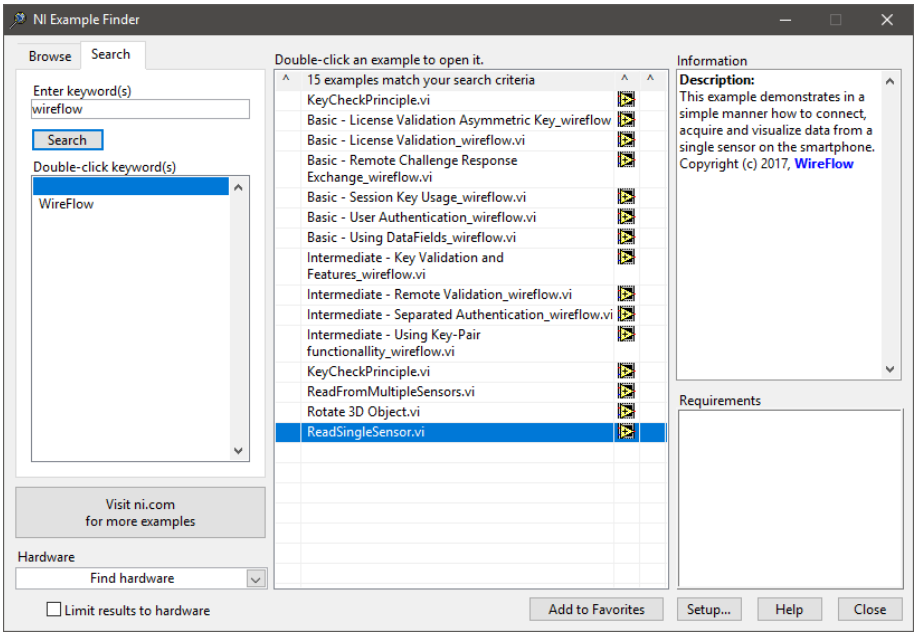
When everything is downloaded and installed (see Download & Install) you can start acquiring sensor data from your smartphone.

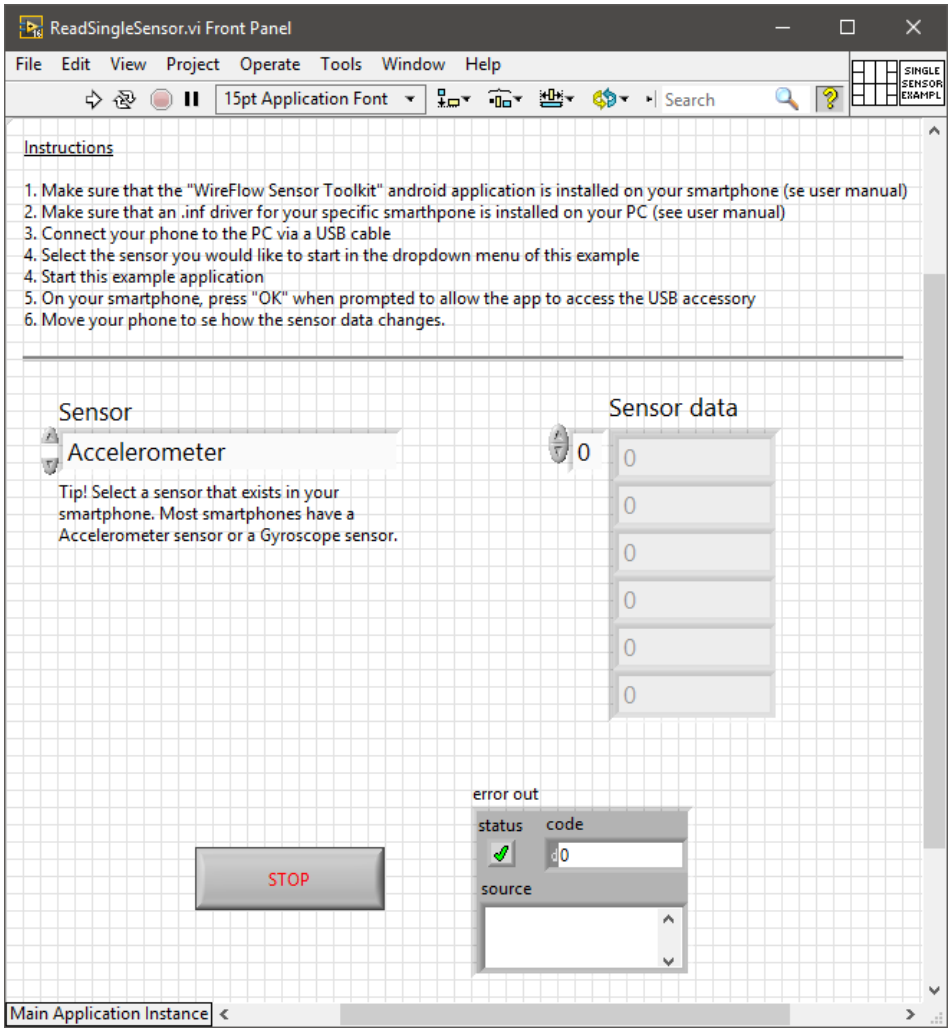
Quickstart

It's recommended that you first try one of our example VI's to see that everything works. Start by opening LabVIEW (≥ 2014) and, in the top menu, choose *Help>>Find Examples…*



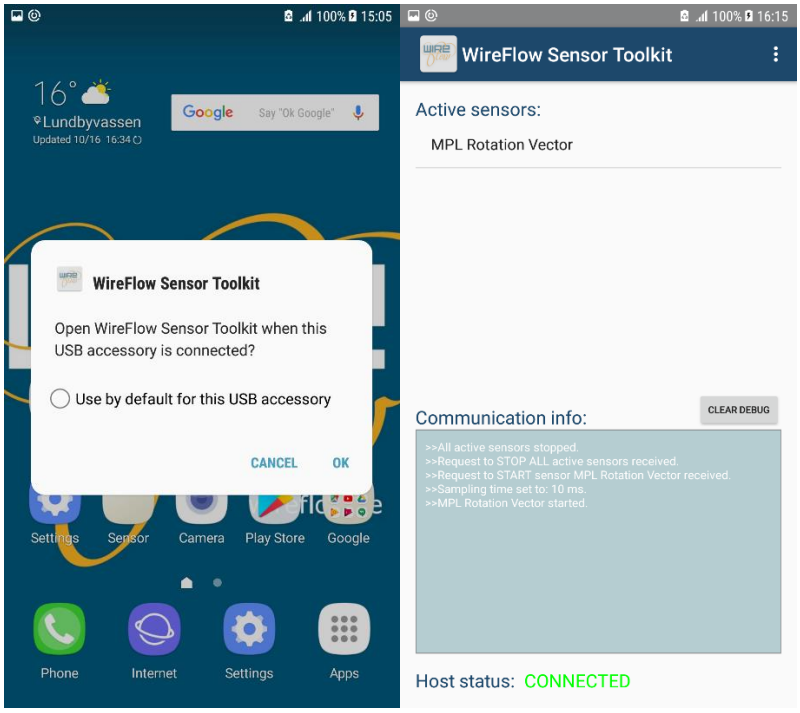
In the new window that appears, type "wireflow", press Search and choose to open the VI named ReadSingleSensor.vi





Now, plug in your smartphone to your PC. At this point, it might be a good idea to see that the device enumerates itself as a NI-VISA USB Device in Windows Device Manager or to find it in NI Measurement & Automation Explorer under Device and Interfaces. If it doesn't, check out Device not listed as a NI-VISA USB Device?.

When you run the VI, the Android application will start and ask you to connect to the device, press OK.



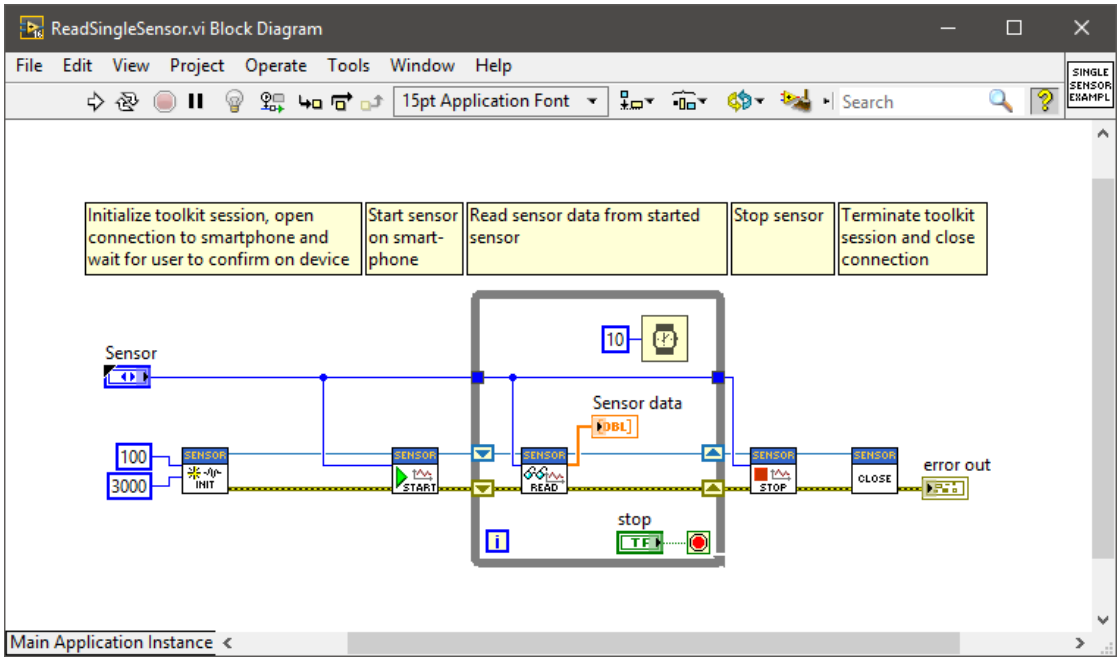
The Android application should now start and activate the selected sensor on your device (should be the Accelerometer if you started the VI without selecting another sensor).

Rotate your device in different ways and look at the Sensor data control on the LabVIEW front panel to see the values change.

Other examples that are available are the *ReadFromMultipleSensors.vi* that starts all the sensors that are available both in the toolkit and in your smartphone, and the *Rotate 3D Object.vi* that shows a 3D object that rotates according to the Rotation Vector in your smartphone.

Using the toolkit

The toolkit consists of a few functions (VIs) that are required to be able to acquire sensor data in LabVIEW. Let's look at a simple example.



The ***Init.vi*** starts the sensor toolkit session. This needs to be executed first for the background service that communicates with the smartphone to start. This VI also requires an input for the desired sampling time, which will be used for all sensors that are started. The second numeric input (in this example 3000) is a timeout constant that tells how long the toolkit session should look for a connection with your smartphone before continuing/exititing.

The ***StartSensor.vi*** starts the selected sensor in the toolkit and in the smartphone.

The ***ReadSensor.vi*** reads the incoming sensor data from the started sensor(s). The output format is always a 1D array of floating point values, but the array size differs depending on what kind of sensor is read (see, Supported sensors).

The ***StopSensor.vi*** stops the selected sensor in both the toolkit and in the smartphone.

The ***Close.vi*** closes the sensor toolkit session. This always needs to be executed lastly for the background process that communicates with the smartphone to shut down correctly.

Error codes

The Sensor Toolkit API has a number of custom error codes that is used to describe the reason for a failure to connect- or read a sensor etc.

Error code	Description
6601	SensorAPI not initialized
6602	SensorAP already initialized
6603	Failed to initialize SensorAPI
6604	No connection to phone
6605	No reply from Android device
6606	Sensor is already started
6607	Specified sensor is not started
6609	Sensor doesn't exist on this Android device
6610	Invalid command
6611	Invalid sampling time

Technical support and FAQ

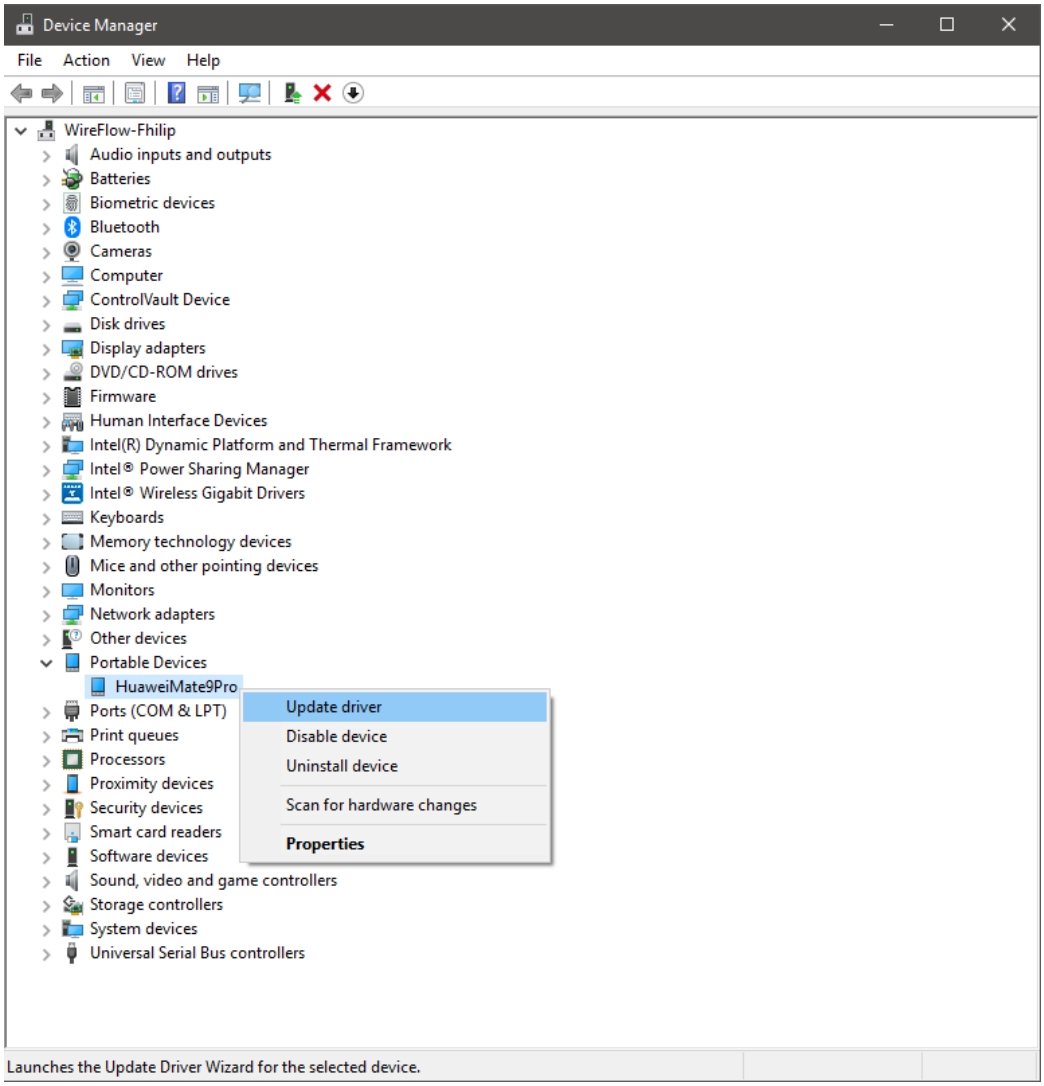
FAQ

Q: Is the toolkit available for iOS devices?
A: No.

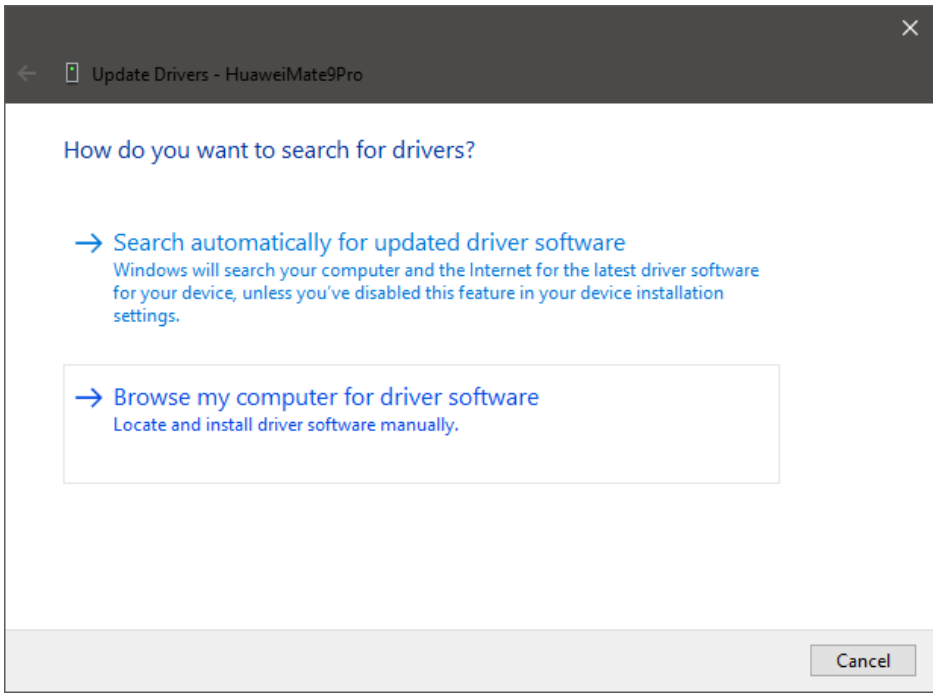
Q: Can I set individual sampling frequencies for individual sensors?
A: No.

Device not listed as a NI-VISA USB Device?

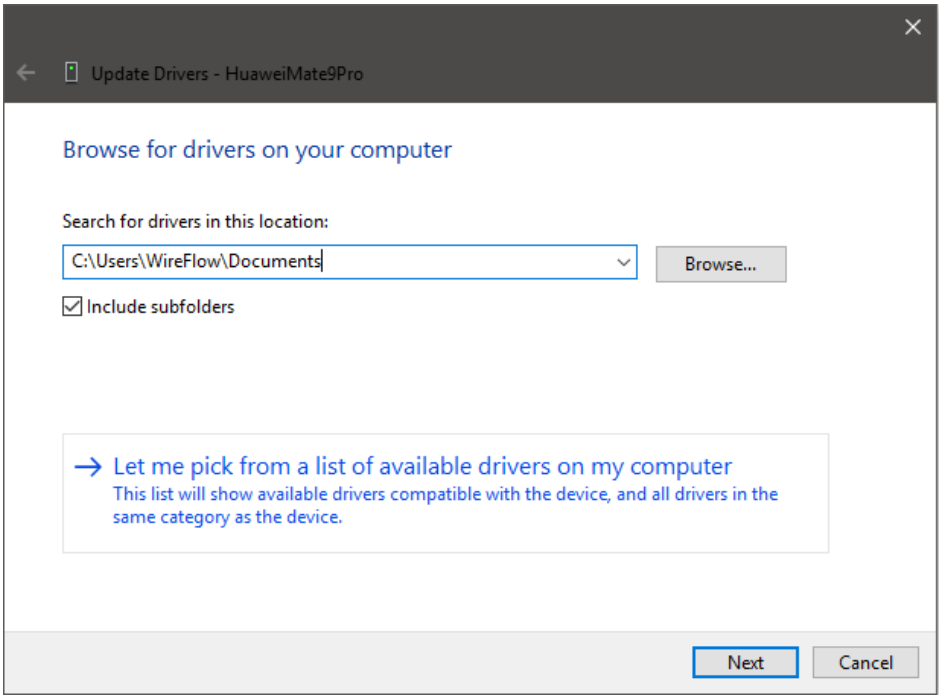
If your device has been connected to the PC before installing this driver, it is highly possible that the device is listed under another category in device manager, e.g. Portable Devices. If that's the case, right click your device in the device manager and choose "Update driver".



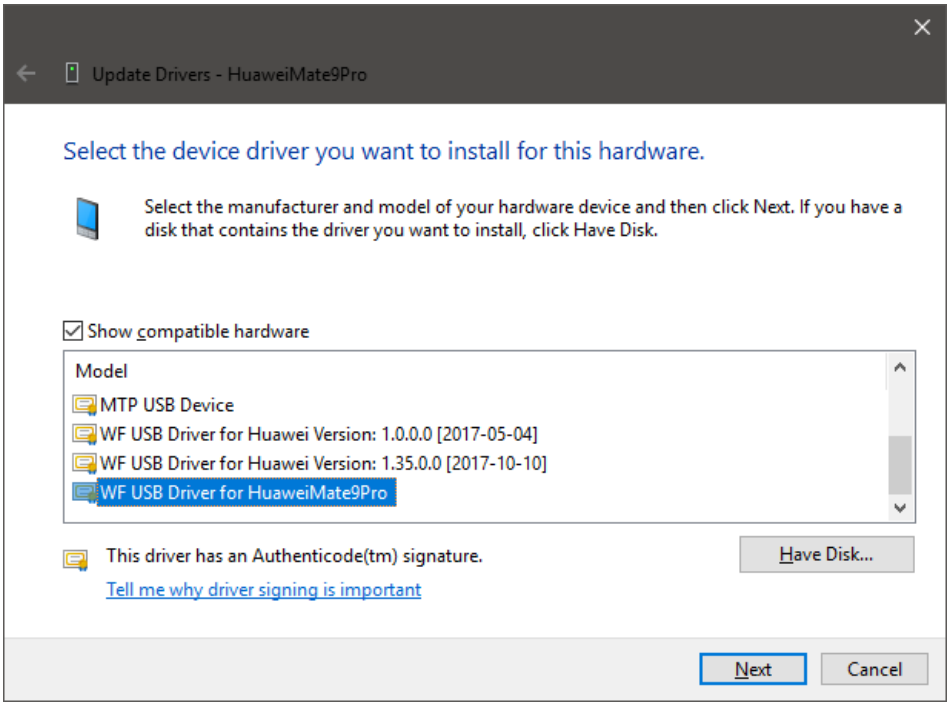
Next, choose “Browse my computer for driver software”.



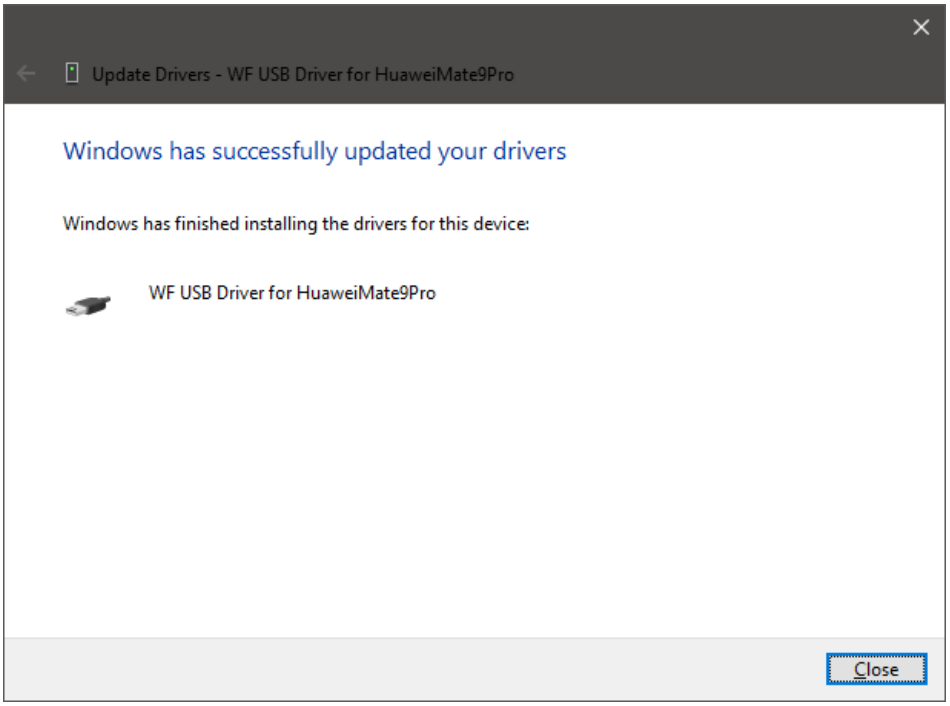
Next, choose “Let me pick from a list of available drivers on my computer.”



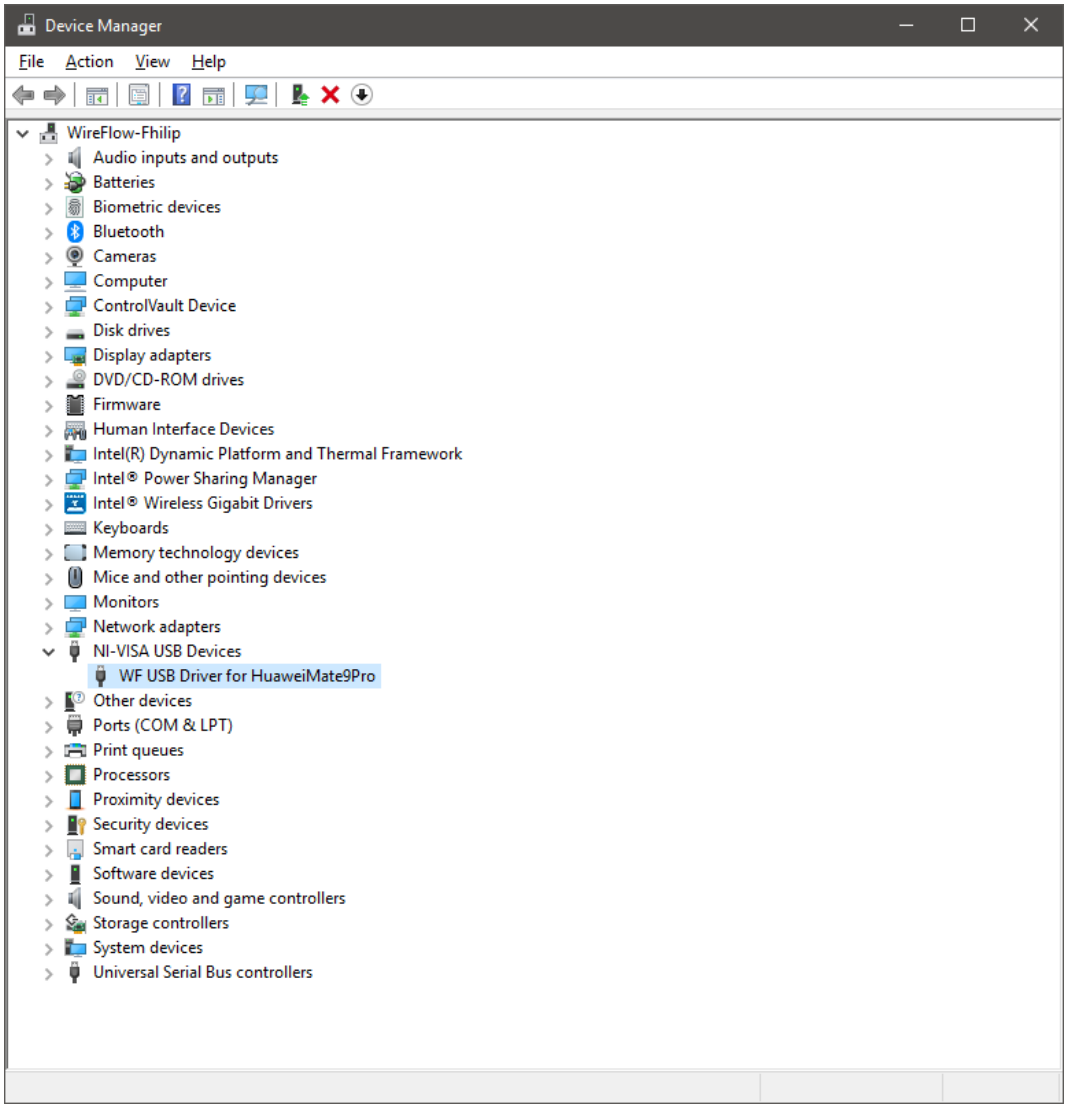
Find the driver named “WF USB Driver For *Vendor/ModelSpecificName*”, choose it and click “Next”.



The driver should now be installed for the device and the following window will appear,



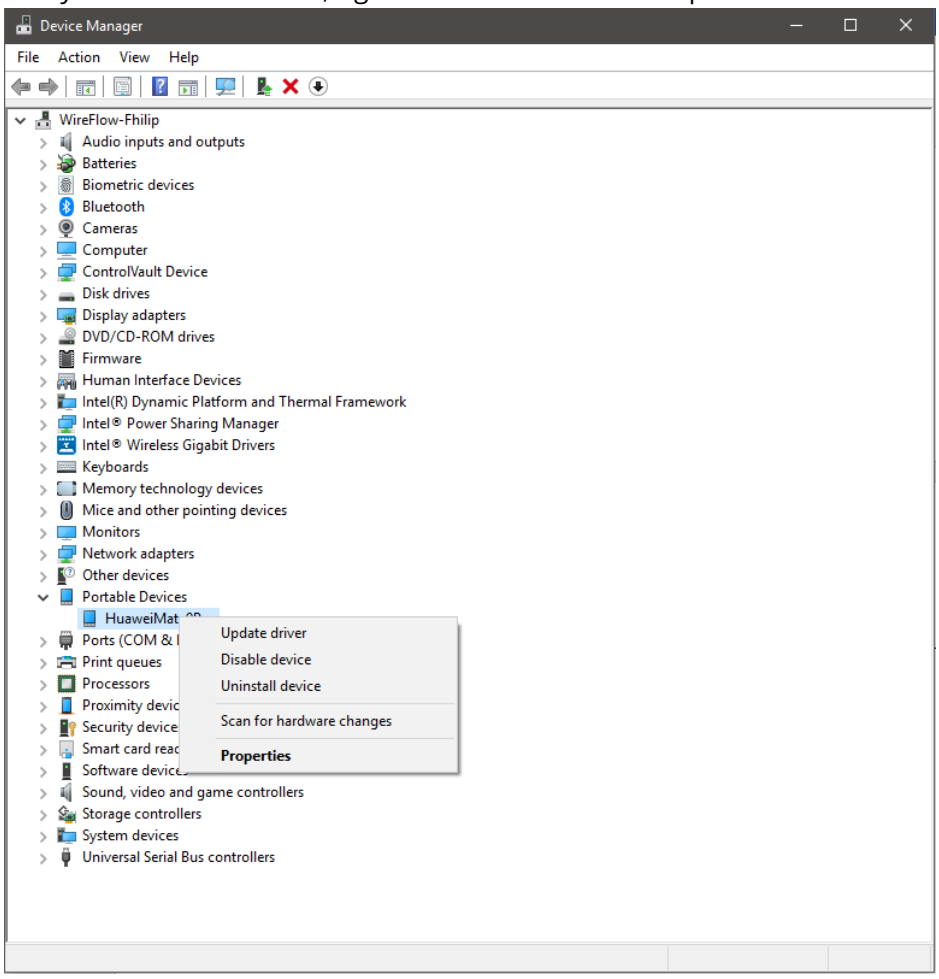
If everything went ok, the device should now show up in the device manager under NI-VISA USB Devices as in the following image,



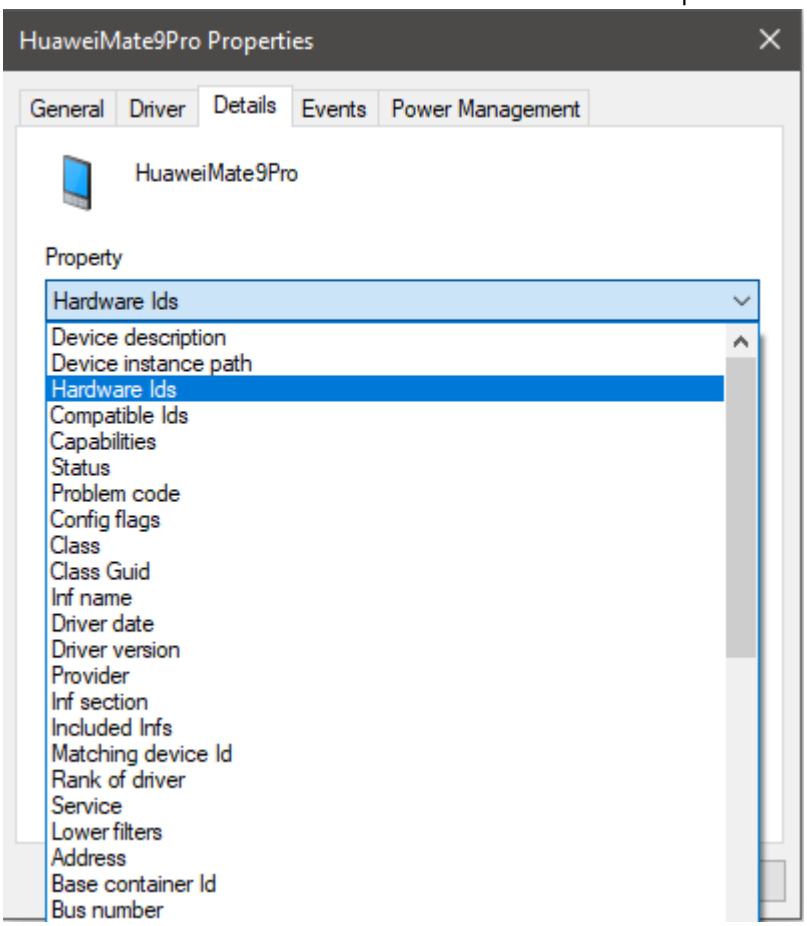
No USB Driver for your device?

If your USB driver isn't listed among the downloaded drivers, please execute the following steps.

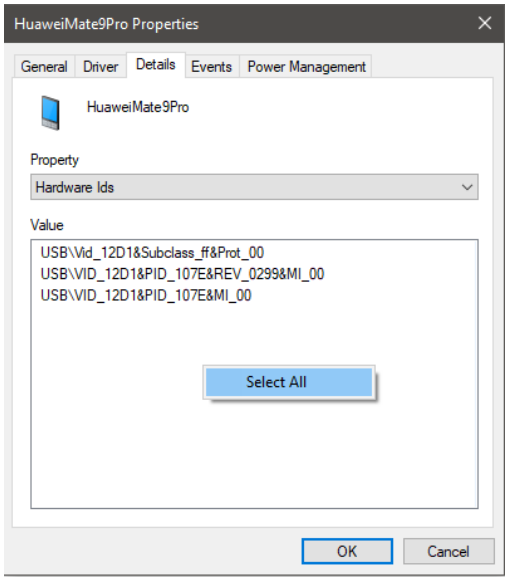
1. Connect your smartphone to your PC.
2. Open up "Device Manager" in Windows.
3. Find your device in the list, right-click it and choose "Properties".

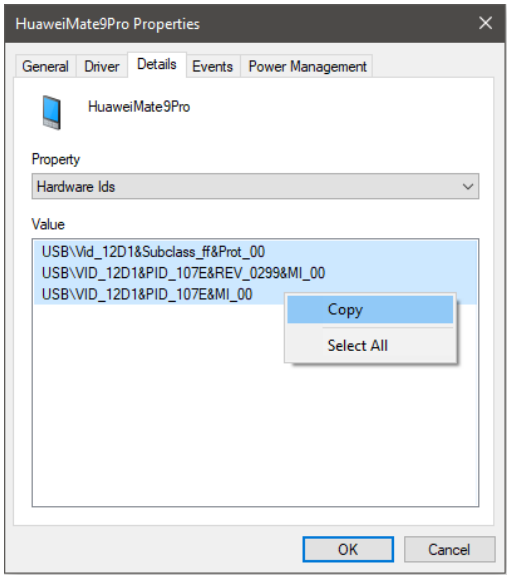


4. Press the “Details”-tab and choose “Hardware IDs” in the drop-down menu.



5. Right-click somewhere in the window and “Select All”, right-click on one of the IDs and select “Copy”.





6. Send an email to support@wireflow.se with the title “Smartphone Sensor Toolkit, New Driver” and paste the Hardware IDs in the email. WireFlow will then reply as soon as possible with a new driver for your device

Device not showing at all in Windows Device Manager?

This might occur for a number of reasons. The most likely one is probably that the smartphone is connected in *charge only-mode*. This can be solved by changing the connection option in your smartphone to MTP (Media Transfer Protocol) mode. If this still doesn’t solve the problem, try connecting the smartphone using a different USB-cord.

References

[1] N/A, "Sensors Overview | Android Developers," [Online]. Available: https://developer.android.com/guide/topics/sensors/sensors_overview.html. [Accessed 25 October 2017].