CRVS technical guide
SmartVA Auto-Analyse:
User guide

March 2020

Resources available from the University of Melbourne, Bloomberg Philanthropies Data for Health Initiative

CRVS course prospectuses
These resources outline the context, training approach, course content and course objectives for the suite of CRVS trainings delivered through the Bloomberg Philanthropies Data for Health Initiative. Each course focuses on a specific CRVS intervention or concept, and is designed to support countries to strengthen their CRVS systems and data.

CRVS Fellowship reports and profiles
The CRVS Fellowship Program aims to build technical capacity in both individuals and institutions to enhance the quality, sustainability and health policy utility of CRVS systems in Fellows' home countries. *Fellowship reports* are written by Fellows as a component of the program, and document, in detail, the research outcomes of their Fellowship. *Fellowship profiles* provide a summary of Fellows' country context in relation to CRVS, an overview of the Fellowship experiences, the research topic and the projected impact of findings.

CRVS analyses and evaluations
These analytical and evaluative resources, generated through the Initiative, form a concise and accessible knowledge-base of outcomes and lessons learnt from CRVS initiatives and interventions. They report on works in progress, particularly for large or complex technical initiatives, and on specific components of projects that may be of more immediate relevance to stakeholders. These resources have a strong empirical focus, and are intended to provide evidence to assist planning and monitoring of in-country CRVS technical initiatives and other projects

CRVS best-practice and advocacy
Generated through the Initiative, CRVS best-practice and advocacy resources are based on a combination of technical knowledge, country experiences and scientific literature. These resources are intended to stimulate debate and ideas for in-country CRVS policy, planning, and capacity building, and promote the adoption of best-practice to strengthen CRVS systems worldwide.

CRVS country reports
CRVS country reports describe the capacity-building experiences and successes of strengthening CRVS systems in partner countries. These resources describe the state of CRVS systems-improvement and lessons learnt, and provide a baseline for comparison over time and between countries.

CRVS technical guides
Specific, technical and instructive resources in the form of *quick reference guides, user guides* and *action guides*. These guides provide a succinct overview and/or instructions for the implementation or operation of a specific CRVS-related intervention or tool.

CRVS tools
Interactive and practical resources designed to influence and align CRVS processes with established international or best-practice standards. These resources, which are used extensively in the Initiative's training courses, aim to change practice and ensure countries benefit from such changes by developing critical CRVS capacity among technical officers and ministries.

Made possible through funding from Bloomberg Philanthropies www.bloomberg.org

Suggested citation

Richards, B, Klammer, M, Joshi, R, Chowdhury, H, Firth, S. *SmartVA Auto-Analyse: User guide.* CRVS technical guides. Melbourne, Australia: Bloomberg Philanthropies Data for Health Initiative, Civil Registration and Vital Statistics Improvement, University of Melbourne; 2020.

Contents

SmartVA Auto-Analyse: User guide

Welcome to the *SmartVA Auto-Analyse user guide*. This user guide provides physicians with instructions for installing and running version 2.15 of the program.

Introduction

SmartVA Auto-Analyse ('Auto-Analyse') is a diagnostic program for physicians that runs on Tariff 2.0 to analyse verbal autopsy (VA) data and produce a computer certification of cause of death (COD). Auto-Analyse produces real-time COD estimates by analysing data through Open Data Kit (ODK) Briefcase and the SmartVA application. Reducing the five-step process used by ODK Briefcase to one single step, Auto-Analyse produces a list of the top three most likely causes of death, as well as the full list of endorsed diagnostic symptoms (symptoms reported by the decedent's family member/s during the VA interview) used for the prediction.

Auto-Analyse differs from SmartVA-Analyze in a number of important ways. Firstly, Auto-Analyse changes the sequence in which the different VA sections are presented to certifying physicians, commencing with the open-ended narrative and past history, followed by the system-based structured questions. Alongside reporting the top three causes of death predicted by SmartVA-Analyze as determined from the Tariff score (including the likelihood of dying from each cause: possible, somewhat likely, likely, very likely), SmartVA Auto-Analyse produces a summary of *all* endorsed symptoms, providing greater insight for the certifying physician to make an informed, final decision on the COD for each decedent. The likelihood of each COD listed for the top three causes is based on the relative ranking of the Tariff scores that met the necessary cause-specific and absolute thresholds.

The download link for Auto-Analyse can be found under section 1.2 (Installation) of this guide. Auto-Analyse can be used on either a combination of an Android tablet and Windows laptop/desktop, or through an Android emulator (such as Droid4x, which ships with version 2.15 of the program) for Windows on a laptop/desktop.

Section 1: Downloading and installing SmartVA Auto-Analyse

1.1 Software pre-requisites

For the successful running of SmartVA Auto-Analyse, a number of software pre-requisites must be met:

- Windows 7 or newer operating system (Mac not supported);

- Windows PowerShell 2.0 or greater must be installed. While this should already come installed with Windows, it has been noted in this user guide should any issues be experienced with running SmartVA Auto-Analyse;

- Java Runtime Environment (JRE) is required to run the ODK Briefcase component. If JRE is not already installed, a notification will appear the first time Auto-Analyse is launched. The latest version of JRE can be downloaded here: https://www.java.com/en/download/.

1.2 Installation

SmartVA Auto-Analyse download link
To begin, download the setup file from the link below:

- https://cloud.mdhs.unimelb.edu.au/index.php/s/AcvrIYIbhVv7CZM

1.3 Run setup

Once the download has completed, close any open programs and run the setup file. To do this, **right-click** on the program and **select '[Run as administrator]'**.
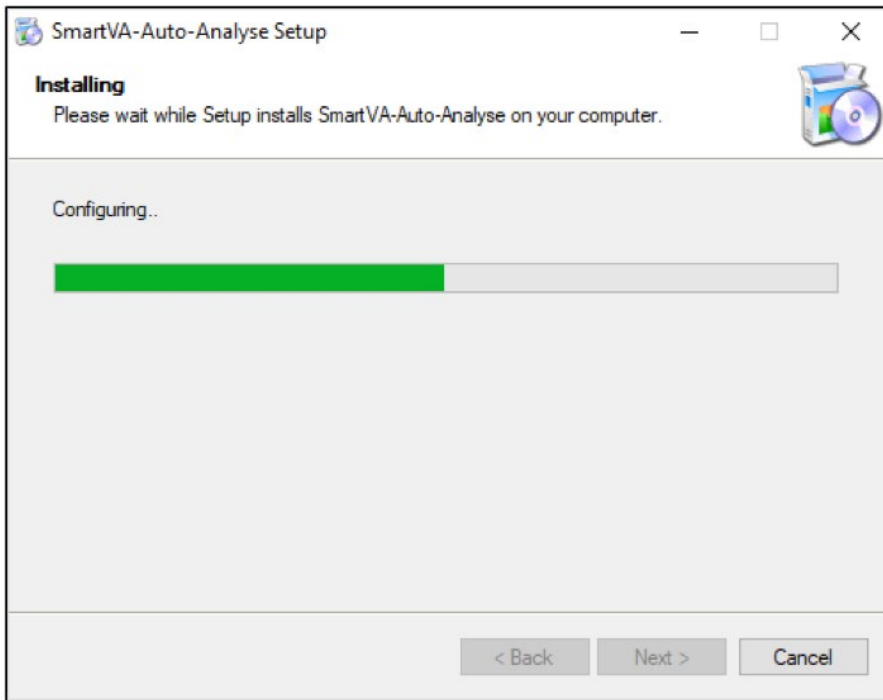
Click [Next] to continue
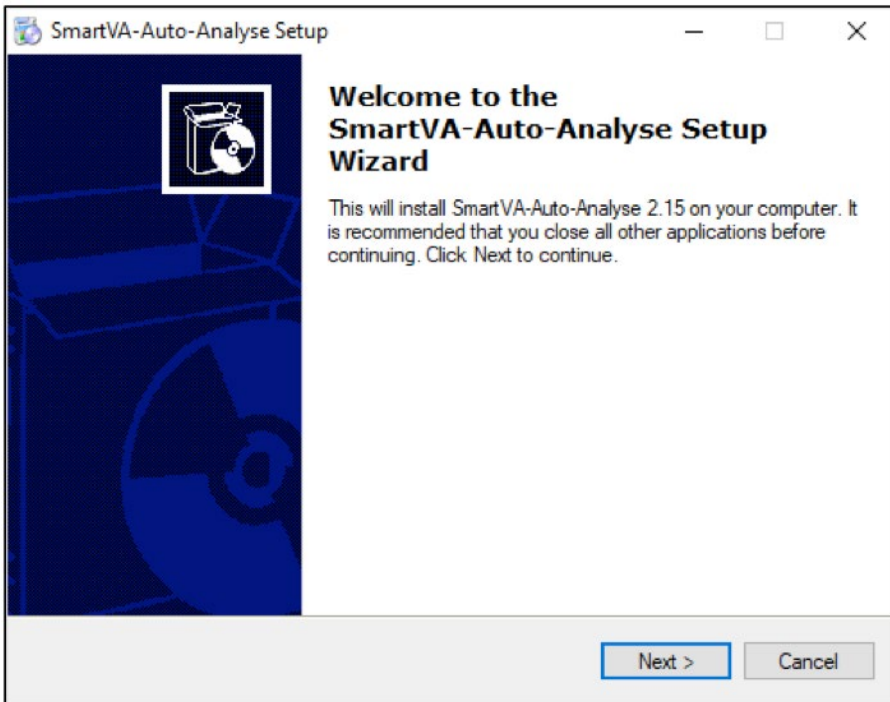
Read the License Agreement and **click** on **'[I accept the agreement]'** if you agree to the license and wish to continue installing. **Click [Next].**
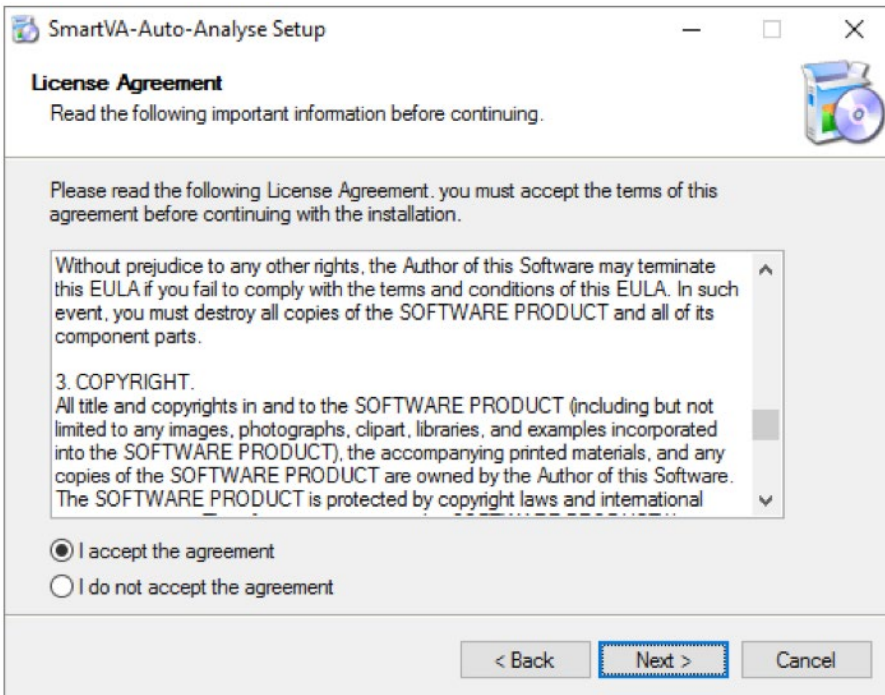

The installer will now install the program files, set up your desktop icon and check for an existing version of **C:\SmartVA**.

If, during configuration, an existing **C:\SmartVA** folder is detected then you will be advised that this will be renamed. This is due to differences in the versions of software in use and is to ensure that old data is kept safe. **Click** **[OK]**.
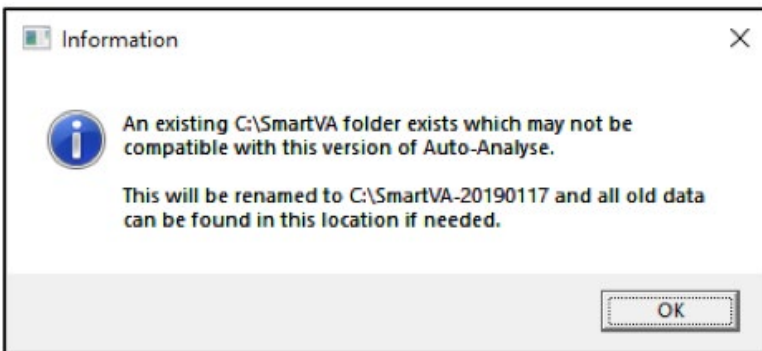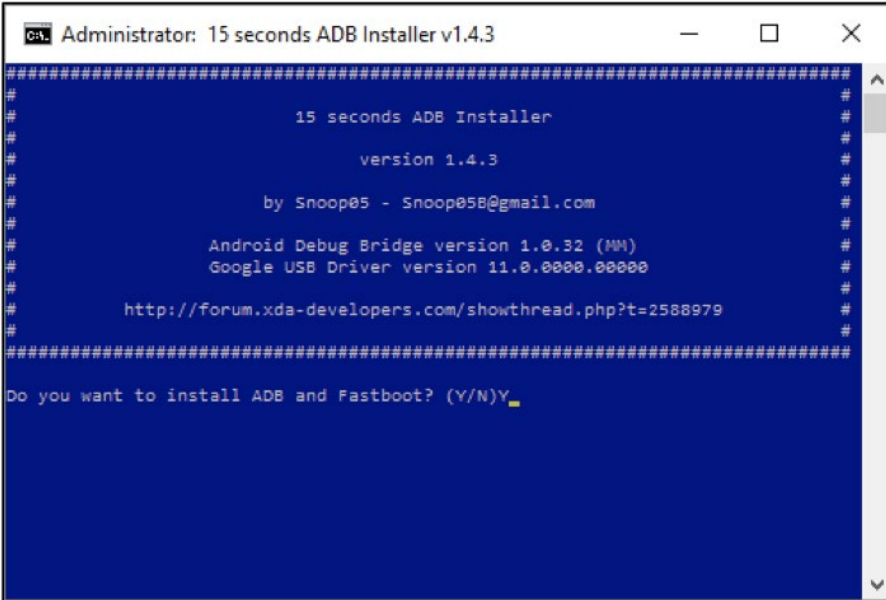
Important: If the installer is unable to rename this folder then you will be advised there was an issue. This folder should be renamed manually to ensure that data does not become corrupted when using the new version.
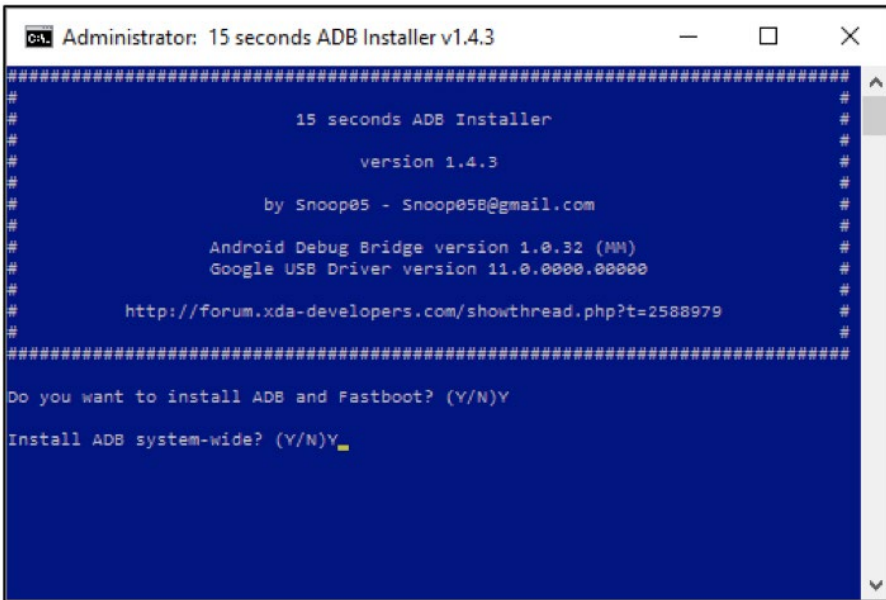
CRVS technical guide
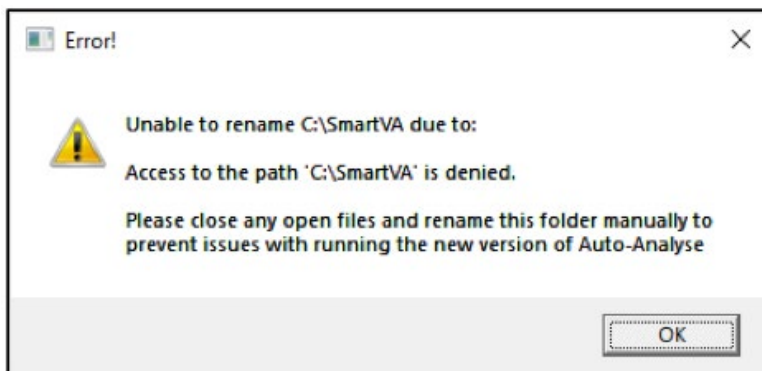
Once this has been performed, the Android Debug Bridge (ADB) install will commence.

The ADB (required to communicate with the tablet) installer will launch. **Answer '[Y]'** to each question to have the ADB component and driver installed. If you have previously installed this, then **answer '[N]'** to each question.
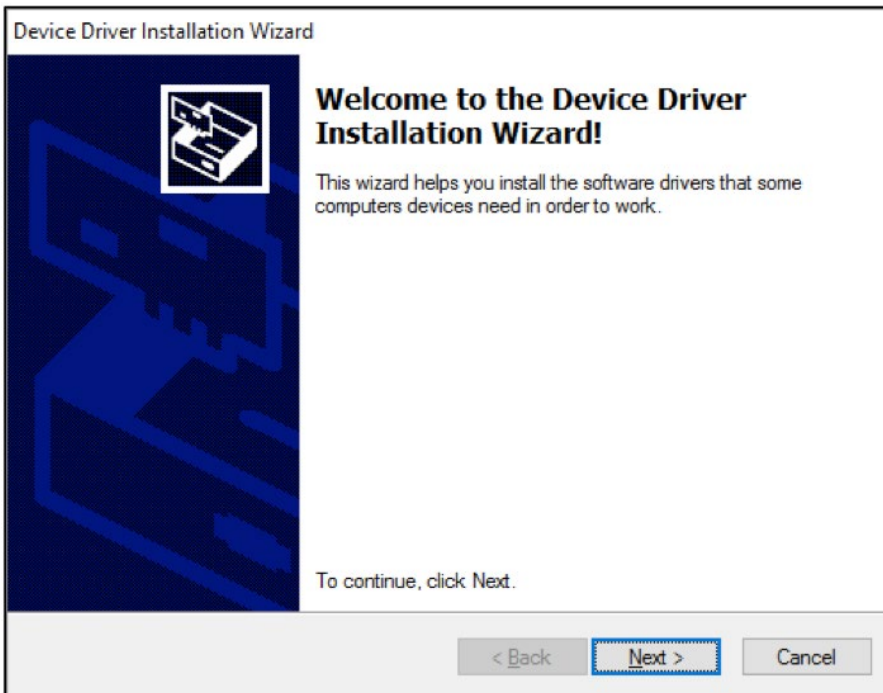
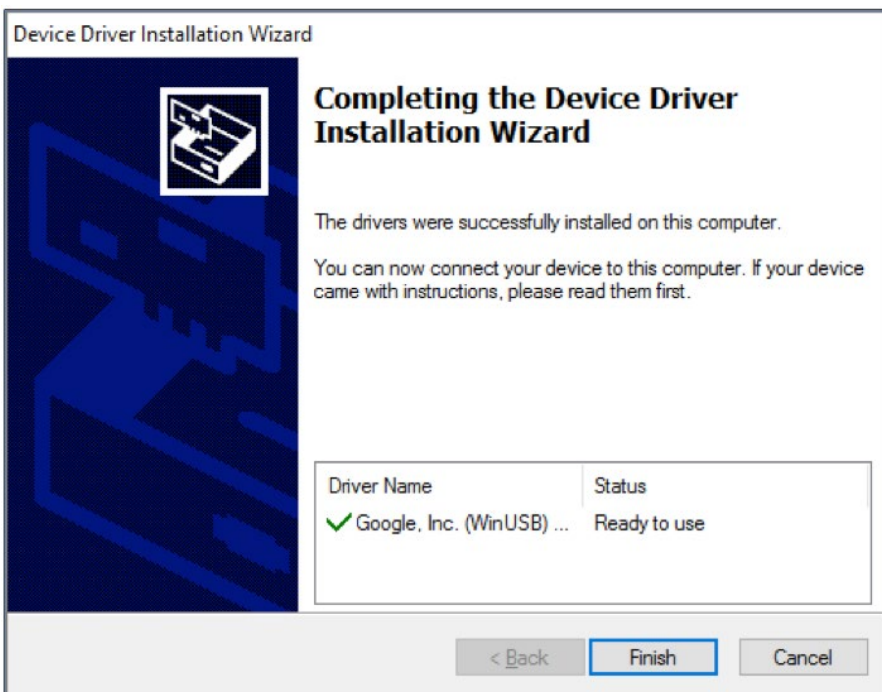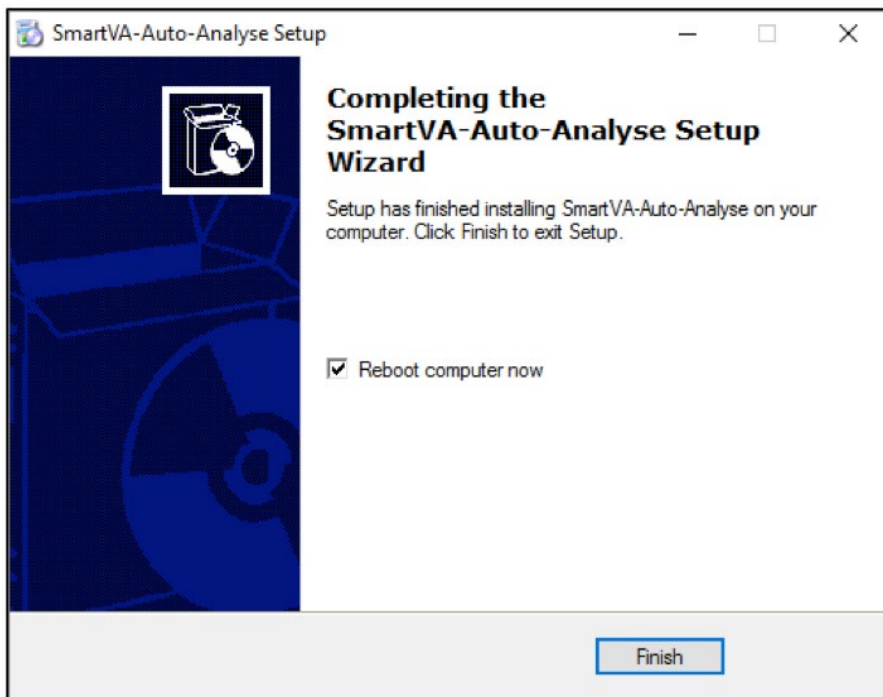The device driver installation wizard will now run. **Click** [**Next**].

You may see a warning like the one below. **Click** [**Install**].
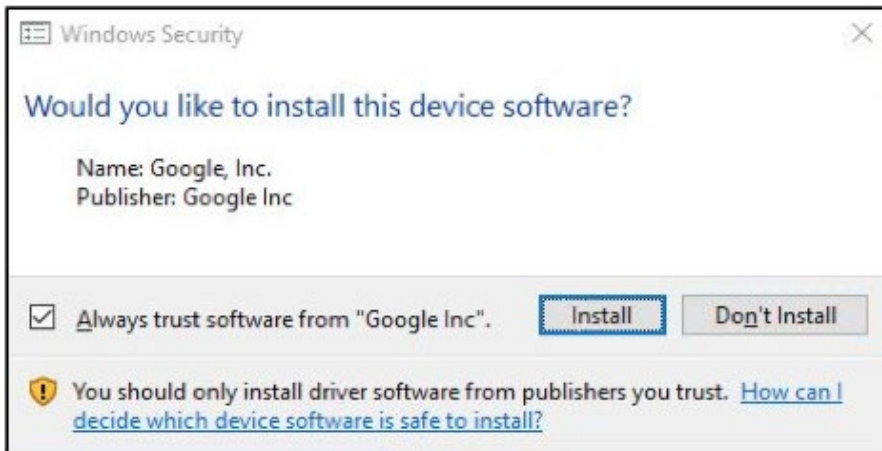
Click [Finish] ato complete the ADB driver installation and return to the program installation.

Finish the program installation by **clicking [Finish]**. This will reboot your computer and finalise the installation of drivers and other components.

Section 2: Configuration file – configuring for your environment

2.1 Opening the file

SmartVA Auto-Analyse uses a configuration file to allow certain settings to be changed. These will need to be reviewed prior to running for the first time and set appropriately for your implementation. Any changes will be reflected when the program is next run. To view and edit this file, you need to open the file in Notepad. To do this, perform the following:

- **Open** Windows Explorer;

- **Navigate** to **C:\University of Melbourne\SmartVA-Auto-Analyse**. **Open** the configuration file **'config.txt'**.

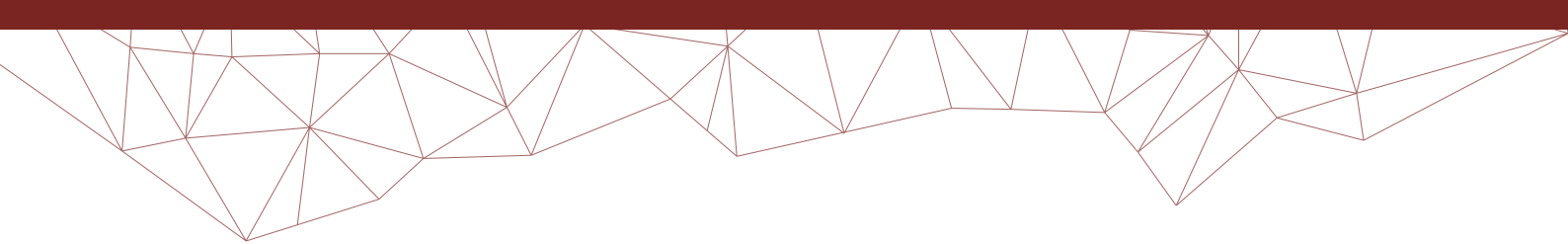The file should now be open for editing and look similar to the below.

2.2 Settings

There are a number of settings in this file, each grouped under specific headings. Each of the headings and configuration options are explained in the table below.

Note that 0 = Off/No and 1= On/Yes for applicable settings.

Setting	Configuration option	Configuration option details	Default value
General The general section used to define general program settings.	**AndroidOld**	If you have an older Android device running version 2.x, or if you are accessing data stored on a USB drive, then set this value to '1'. This will prevent the program from trying to access the device via ADB and will use the USB Mass Storage method of obtain files. If this setting is set to '0', then the program will assume a newer Android device is connected and will use the installed ADB drivers (with tablet in Developer Mode) to pull data directly from the tablet.	0
	DeviceType	This setting allows you to specify if you are running a physical device or an emulator (such as Droid4X, MEmu or Bluestacks). For a physical device, set to '-d'. If using an emulator, leave it as '-e'. Note: the AndroidOld setting should be set to '0' for this to work.	-e
	DisplayResults	This setting defines whether results will be displayed to screen. Setting this to '0' will prevent display to screen and results will be logged in the archive **'results.csv'** file only. Setting this to '1' will display results and prompt for a Physician COD to be entered for each record processed.	1
	RetainResults	This setting changes the way that results are stored in the archive **'results.csv'** file. Setting this to '0' will mean that only results from the current run will be retained. Setting this to '1' will allow the accumulation of all results processed overtime to be stored/maintained.	1

Setting	Configuration option	Configuration option details	Default value
Aggregate Server Information This section contains settings related to your aggregate server if used.	**Aggregate**	Set to '0' if you are processing data from a tablet; Set to '1' if you have an aggregate server that you wish to process data from.	0
	AggregateURL	The URL of your aggregate server should be entered here.	\<blank\>
	AggregateUser	A valid aggregate server username should be entered here.	\<blank\>
	AggregatePass	The password for the above-specified user should be entered here.	\<blank\>
Briefcase Runtime Options This section is used to define the runtime options for ODK Briefcase.	**FormID**	The name of the form in use is required by ODK Briefcase in order to export the correct information. Once a form has been uploaded to an Android device for use with ODK Collect, the form name should also be updated to allow the automated processing of any entries created. **Note: This setting is case-sensitive and needs to match the exact name of the form as seen in the device.**	countryname_ v9
Tariff Runtime Options As with ODK Briefcase, the SmartVA-CLI application requires certain parameters in order to analyse and produce results. These parameters are set in this section.	**country**	A three-letter abbreviation representing the country that the data originated from. Note that all valid country abbreviations are listed at the bottom of the configuration file for easy reference.	PHL
	hiv	This setting is used to indicate if this is a HIV region. Set to '0' for no, and '1' for yes.	0
	malaria	The user must determine whether malaria is a possible COD in the population from which the VAs were collected. If the box next to "Malaria region" is not selected, the Tariff Method will not assign malaria as a COD. Set to '0' for no, and '1' for yes.	0
	hce	This setting relates to health care experience (HCE) variables. The user should determine whether, as part of the survey, questions regarding the HCE of the deceased or his/her family are asked.	1
	freetext	If your data has an open response component and you would like this to be analysed by the Tariff Method, make sure this setting is set to '1'.	1
	figures	This setting allows you to specify if Tariff figures (graphs generated by the SmartVA-Analyze software) are to be generated or not at run-time. As this requires more processing time and overheads, this is turned off by default. Set this value to 'true' to have figures generated, or 'false' to omit these figures when running the program.	false
	language	This setting allows you to specify a supported language. **Note: this does not relate to the language in the SmartVA form, but to the Tariff output files which currently support English, Chinese and Spanish. Leaving this setting blank will default to English. Alternatively specify either 'Chinese' or 'Spanish'.**	\<blank\> (English)
Locations This section contains settings related to file locations for use with the program	**ProcessDir**	This is the 'working directory' for the program and is where the program will store many of the files used to process your results. Note that the **'results.csv'** output file and all archived information will be stored in the 'archive directory' (see ArchiveDir setting below). This directory also contains the Tariff output files that you would usually find in the TariffFiles sub-folder. These are cleared each time the program runs and only the predictions and cause specific mortality fraction Tariff outputs are kept (also stored in the archive folder).	C:\SmartVA

| Setting	Configuration option	Configuration option details	Default value
Locations This section contains settings related to file locations for use with the program	**ArchiveDir**	This is the 'archive directory' for the program. This directory is used to store a copy of files found on your tablet or emulator after processing. If there are any errors with Tariff processing, the program will copy the tablet files to an 'Errors' subfolder which can then be reviewed at a later date. Any files copied to the archive can also be used to process results manually, using ODK Briefcase and SmartVA software. If SmartVA Auto-Analyse is successful in processing the tablet files, a copy of these will be placed under the **'Processed\Forms'** subfolder and **'Processed\ Instances'** subfolder. Only the latest version of any given form or instance is maintained in the archive to prevent duplicates of raw data, and to ensure any changes in VA data are captured in the archive (such as a form/VA that is edited *after* having already run this program). It is possible to set this value to either a drive location or a Universal Naming Convention (UNC) path, and therefore can be used to store the archive on a network drive or location that can then be backed up and retained over time (recommended). If this network location is used for all computers running Auto-Analyse, then the data will be aggregated into the archive, creating one full dataset with associated form(s). Refer to your IT support team to determine if this option is available.	C:\SmartVA
Tablet ODK Collect Directories This section contains settings related to the ODK Collect application directory.	**ODKDir FormsDir / InstancesDir**	This setting should reflect the name of the odk folder as used by ODK Collect on the device. The current version of ODK Collect uses the directory **'odk'**. Should the directory name change with future ODK Collect versions, this setting can be updated to ensure the program will continue to function correctly. FormsDir and InstancesDir are the directory names as they appear in the CollectFiles directory, contained within the ProcessDir as set above. These values should not need to be changed unless ODK Collect software changes the names of these default locations as part of future updates.	odk FormsDir = forms InstancesDir = instances

Section 3: Emulator installation and configuration

This version of Auto-Analyse ships with optional emulator software (Droid4x) to allow the collection and processing of VAs on one device (Windows 7 – 10).
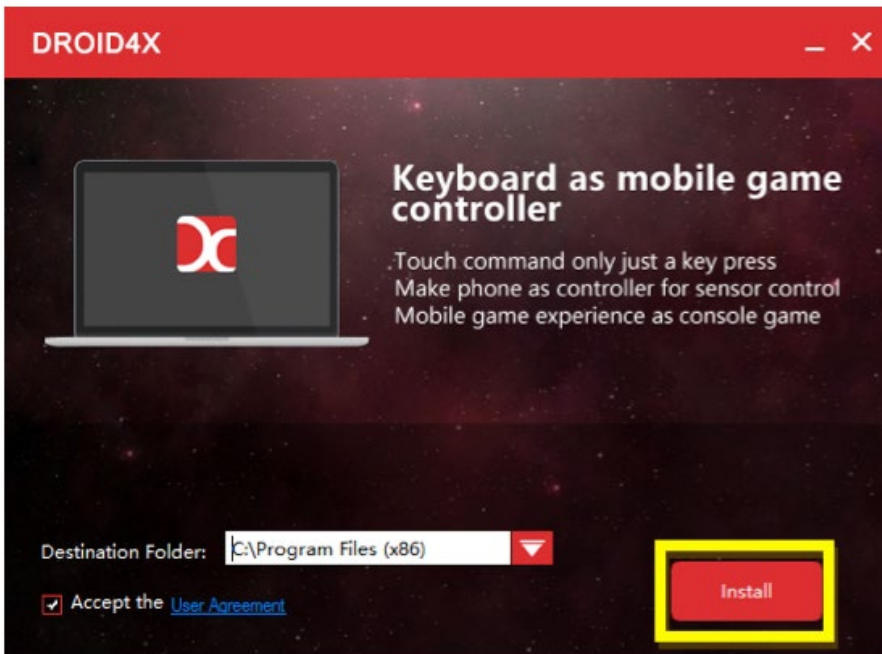
3.1 Installing Droid4X

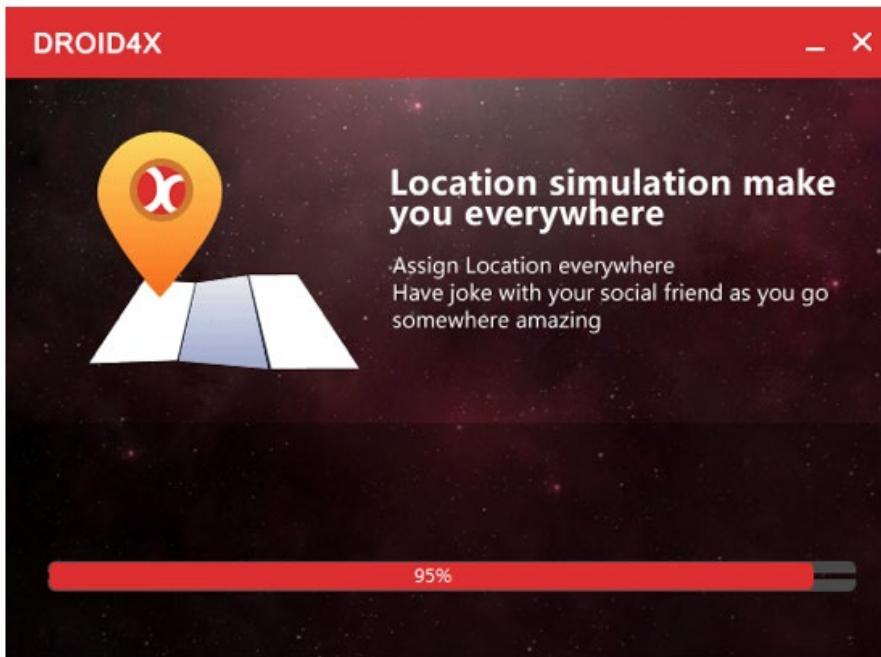To install the emulator, close all open programs and open Windows Explorer.

Navigate to **C:\University of Melbourne\SmartVA-Auto-Analyse** and **right-click** the **Droid4xInstaller.exe** file and select **'[Run as administrator]'**.
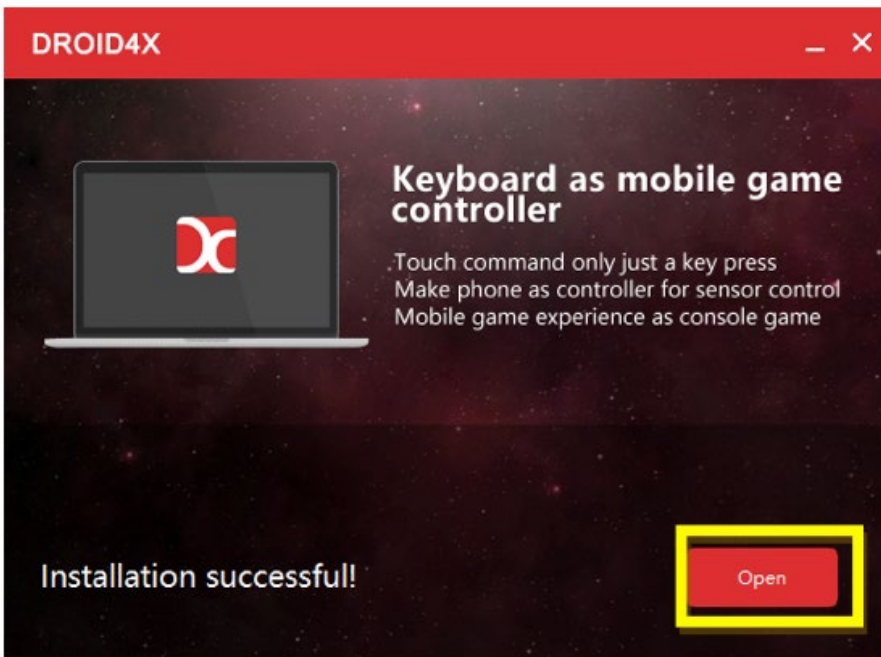
CRVS technical guide
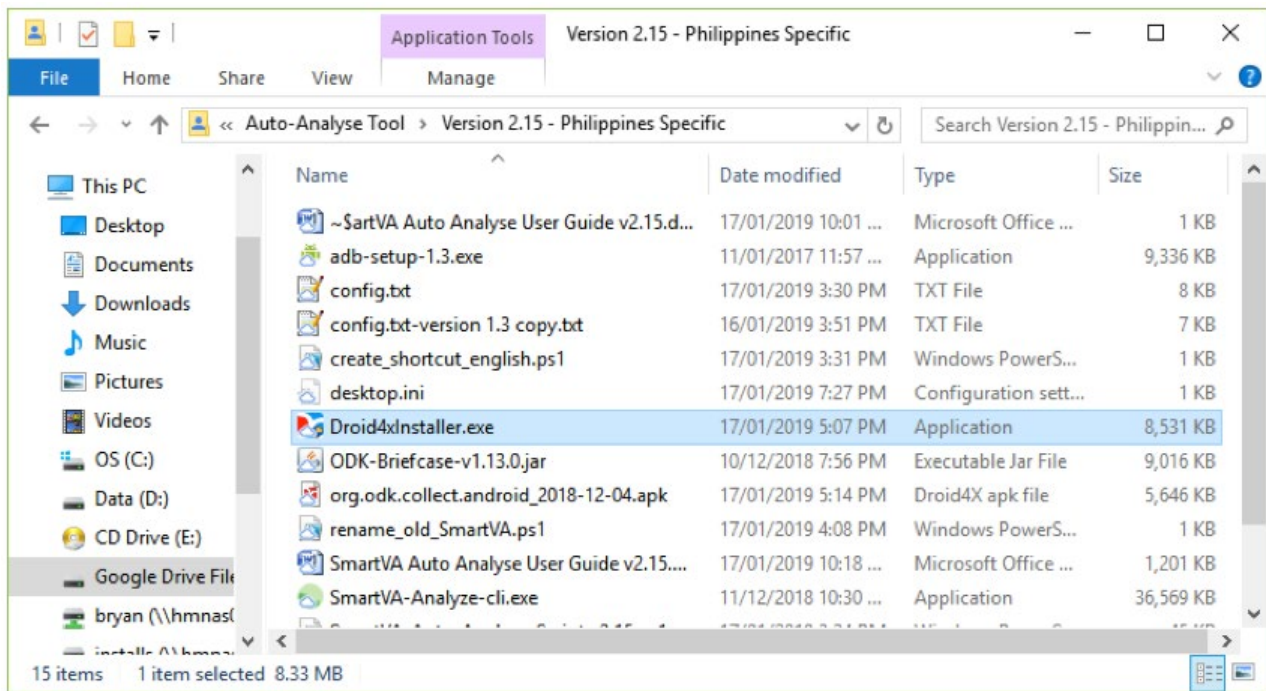
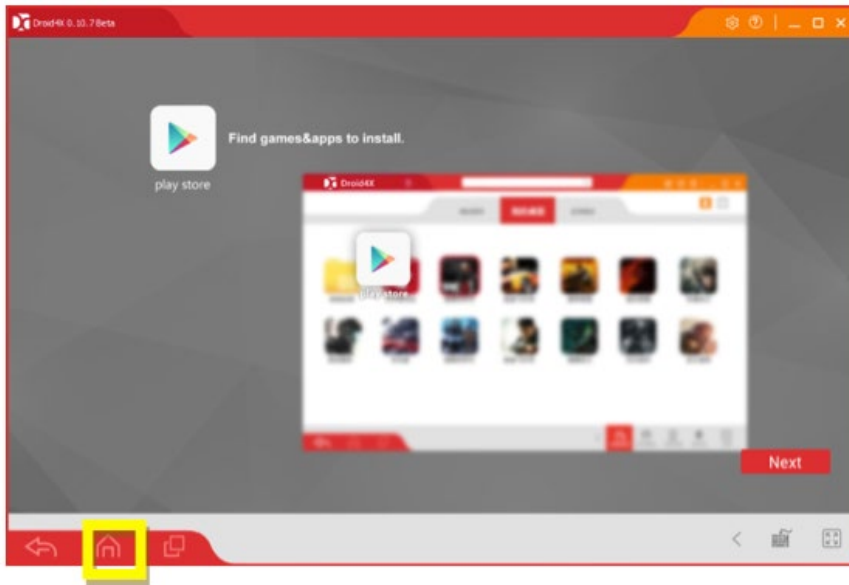Once the installer loads, **click** **[Install]**.

The installer will install the application.

Once successfully installed, **click** **[Open]**.

The emulator will load, and once complete, the screen below will show. **Click** on the **[Home]** button.
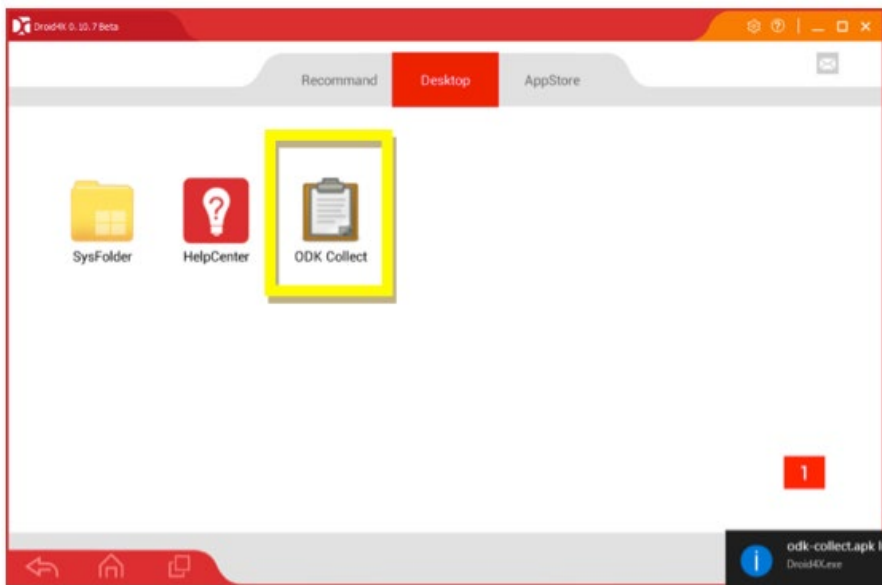
3.2 Installing ODK Collect to emulator

With the emulator open and running, **navigate** to **C:\University of Melbourne\SmartVA-Auto-Analyse** and **double-click** the **'odk-collect.apk'** file. This will install ODK Collect automatically, ready for the next step (adding the VA form).
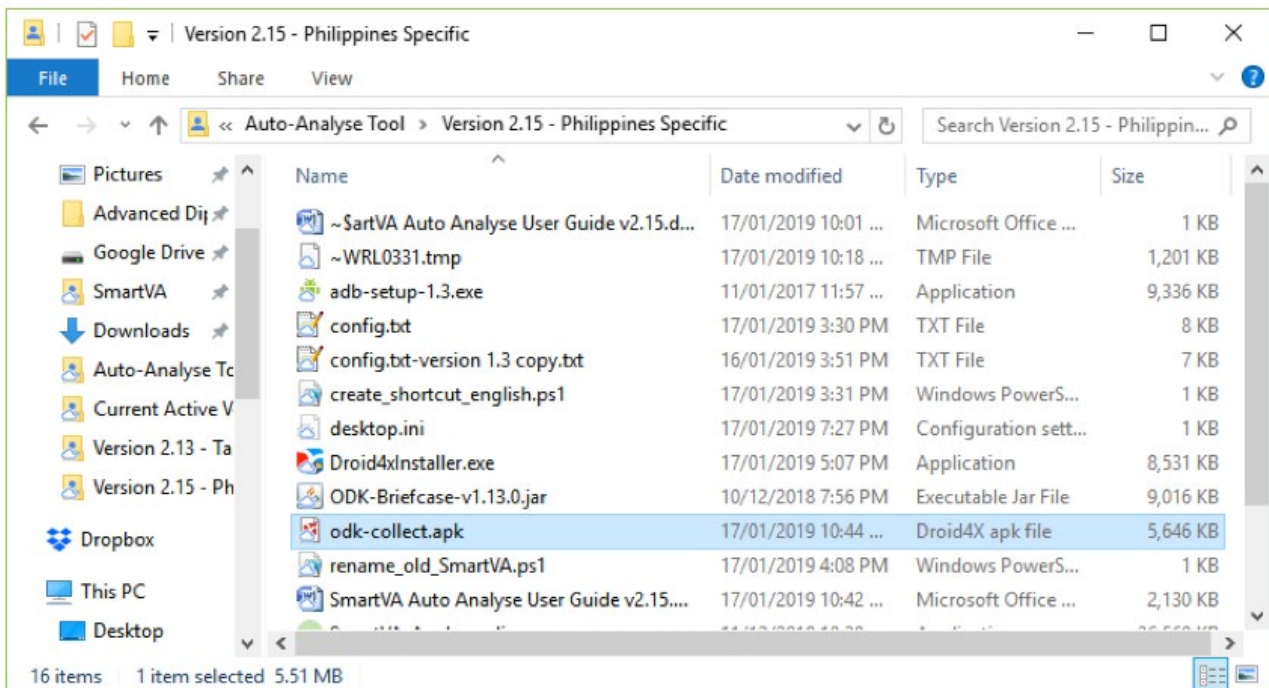
Once installed, you should see ODK Collect appear in the emulator as follows:

**Important: Launch ODK Collect by clicking on it, to confirm it has installed correctly (this is also important for the steps that follow).**

3.2 Installing the VA form to the emulator

To install the form for use with VA, you will need to obtain a copy of the latest form file (**countryname_v9.xml** at time of writing) and place it on your desktop in preparation for the next steps.
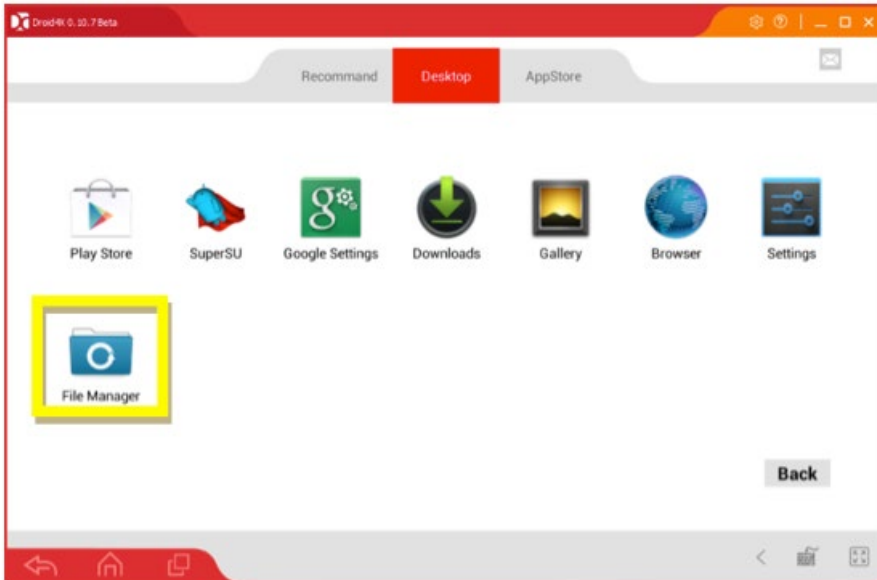
Configure Root Mode

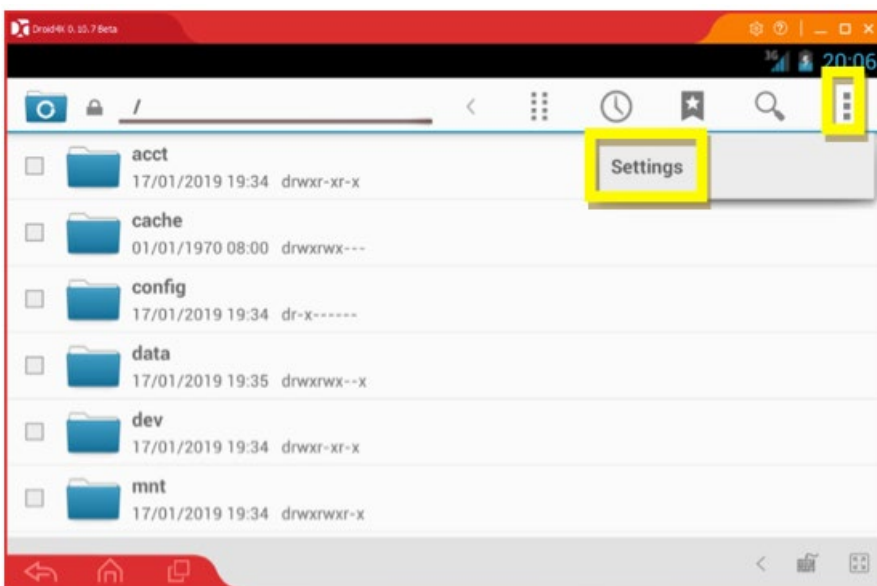In order to copy the form file, Root Access Mode should be enabled.
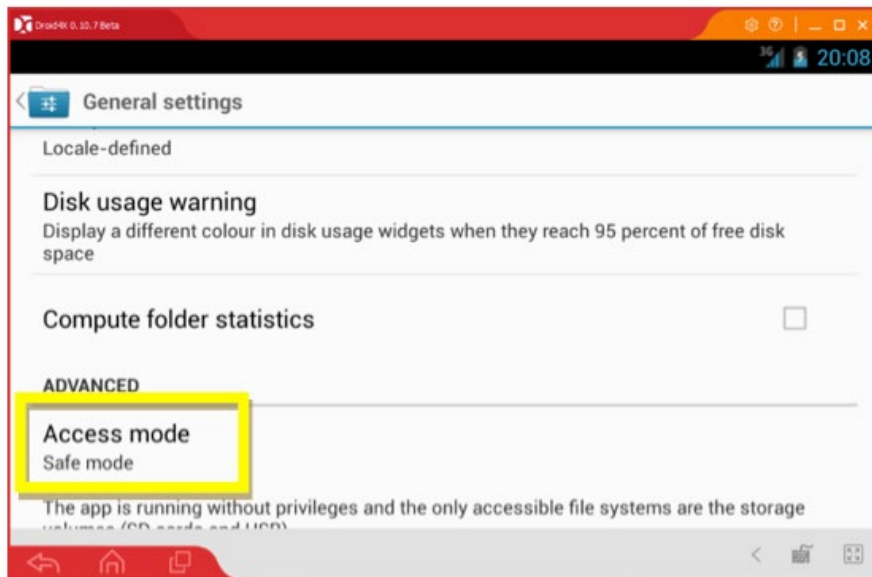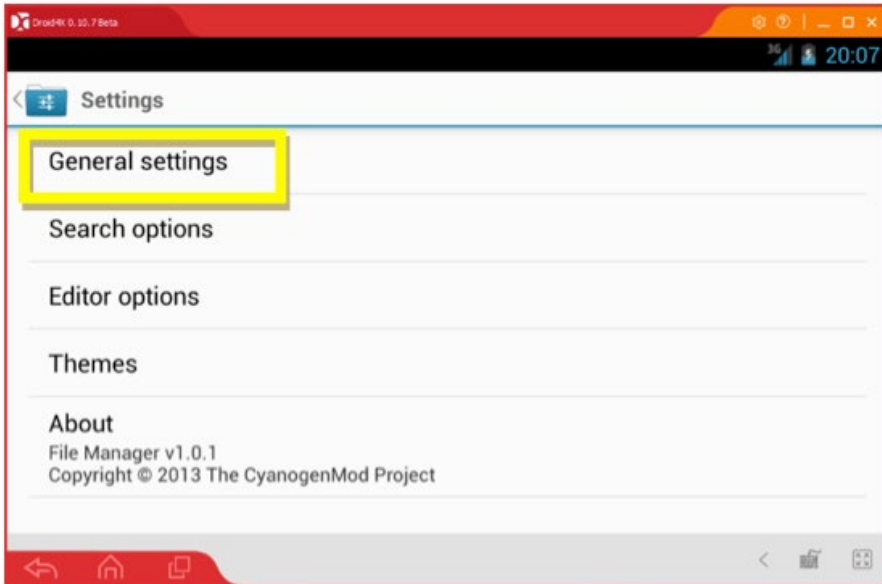
From the emulator home page, **click** on **[SysFolder]**.
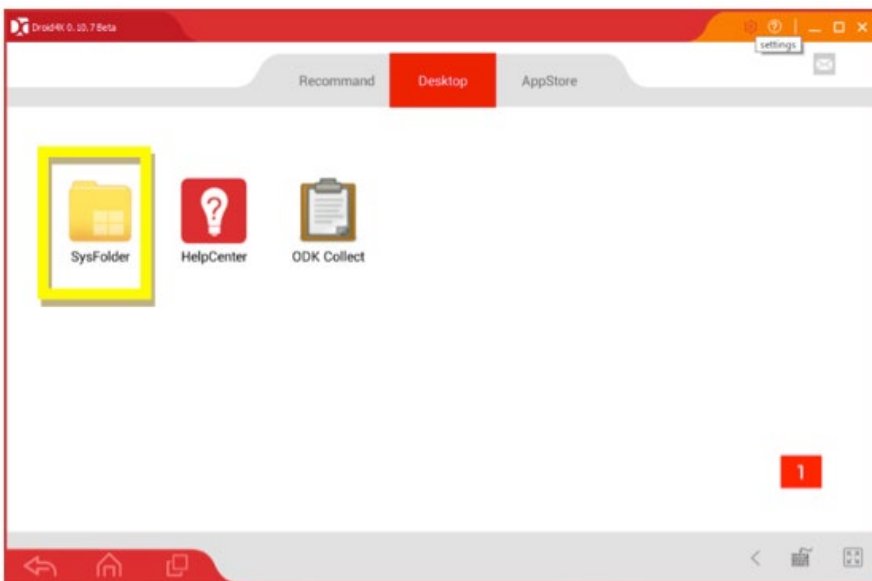
Click on **[File Manager]**.

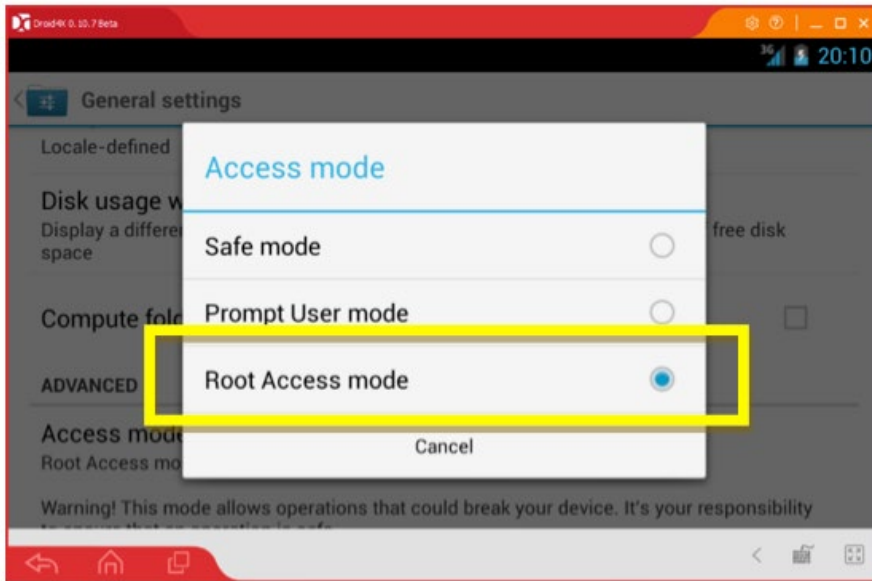**Click** on the **three dots** and **click [Settings]**.

Click on **[General Settings]**, scroll down and **select** **[Access Mode]**.
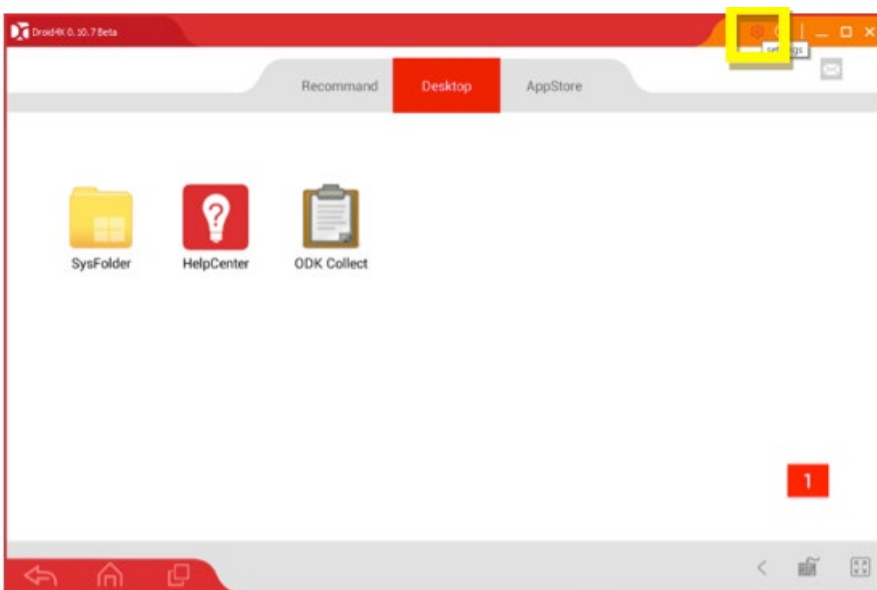

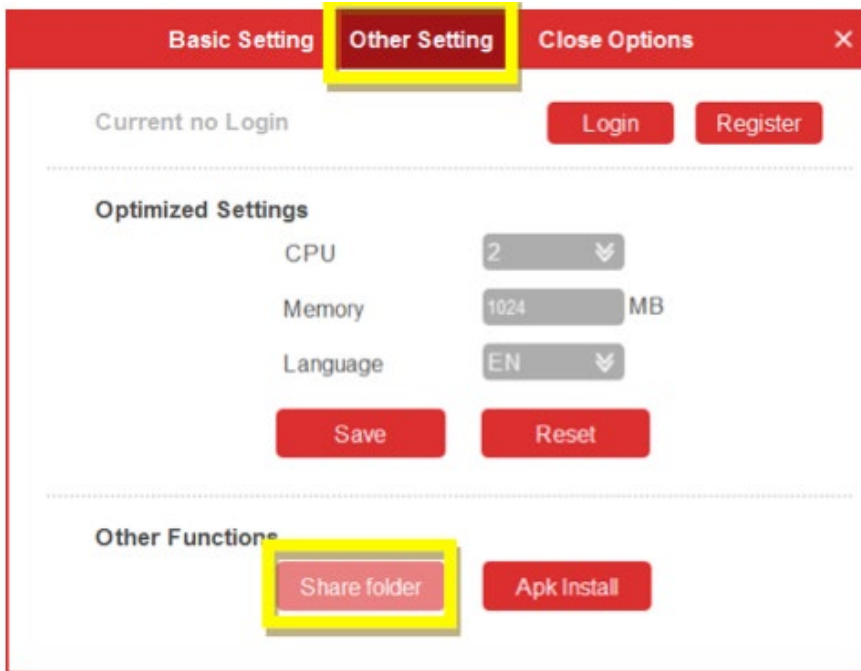**Select** **[Root Access mode]** from the options available:

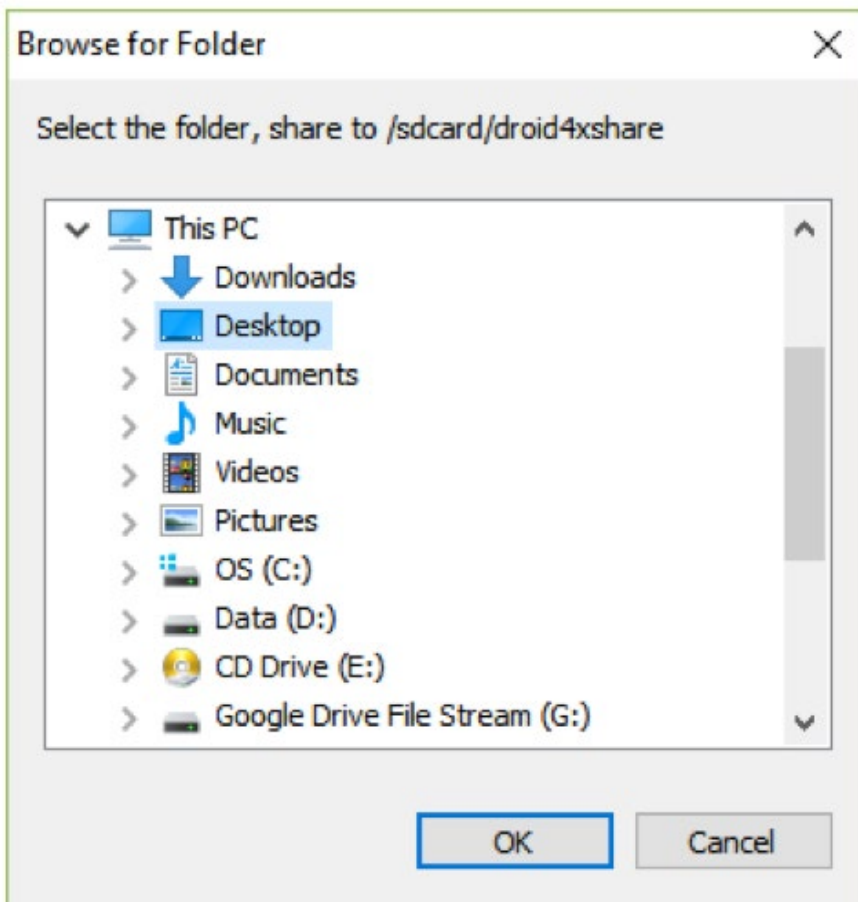**Important: Close the emulator and re-open for the settings to take effect.**

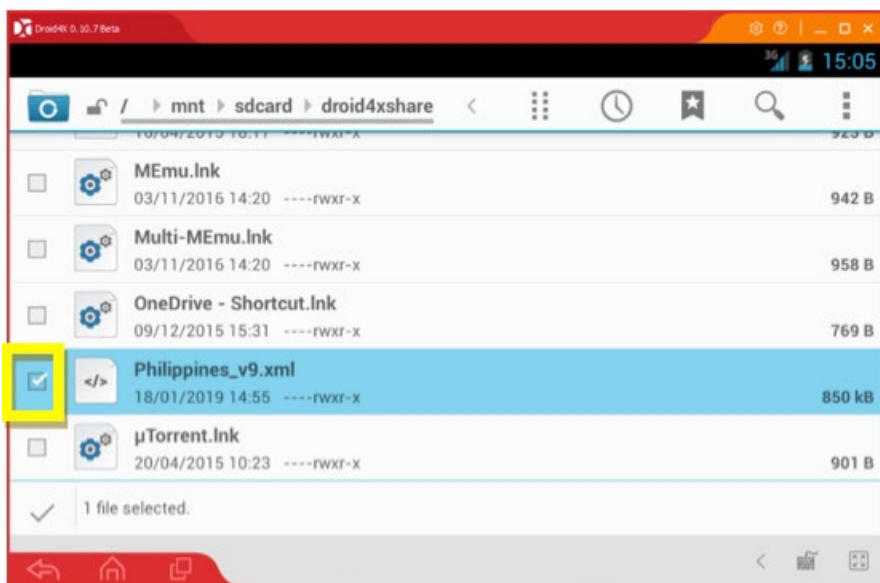With the emulator open, **click** on the **[Settings]** button in the top right.

**Click** on **[Other Setting] > [Share Folder]**.

CRVS technical guide

Select the folder where you have stored your form file (Desktop) and **click** **[OK].**
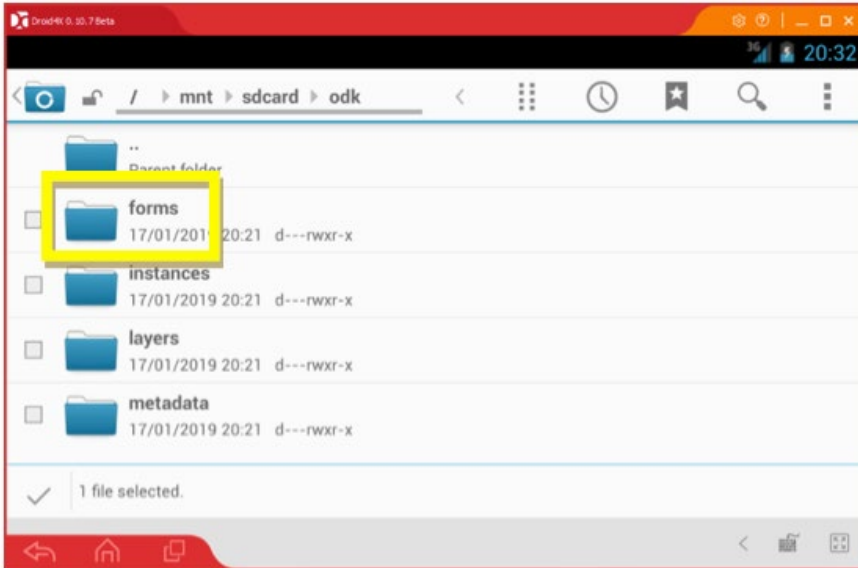
File Manager will open and show you the files on your desktop. Locate the form file (**countryname_v9.xml** in this example) and **select** the **checkbox** next to the file name.
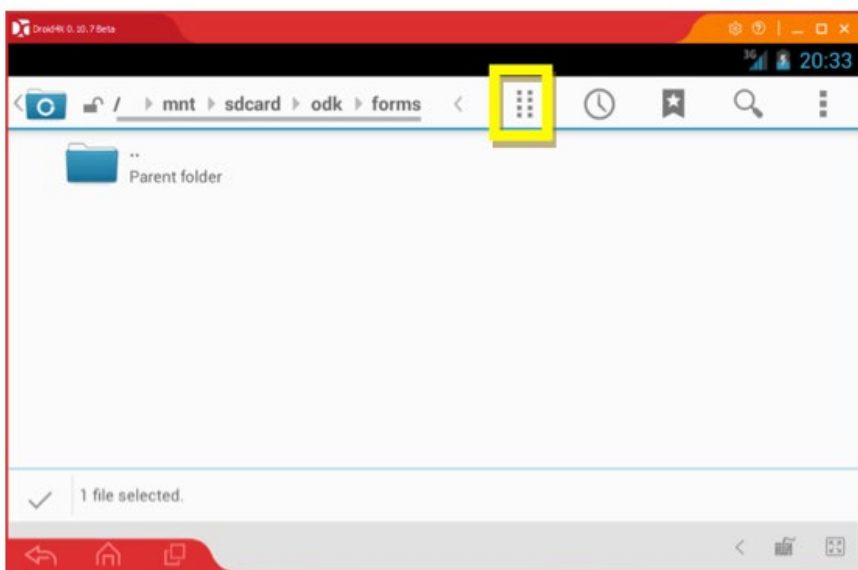

Once selected, **click '[sdcard]'** at the top, scroll down and **click** on the **[odk]** folder.
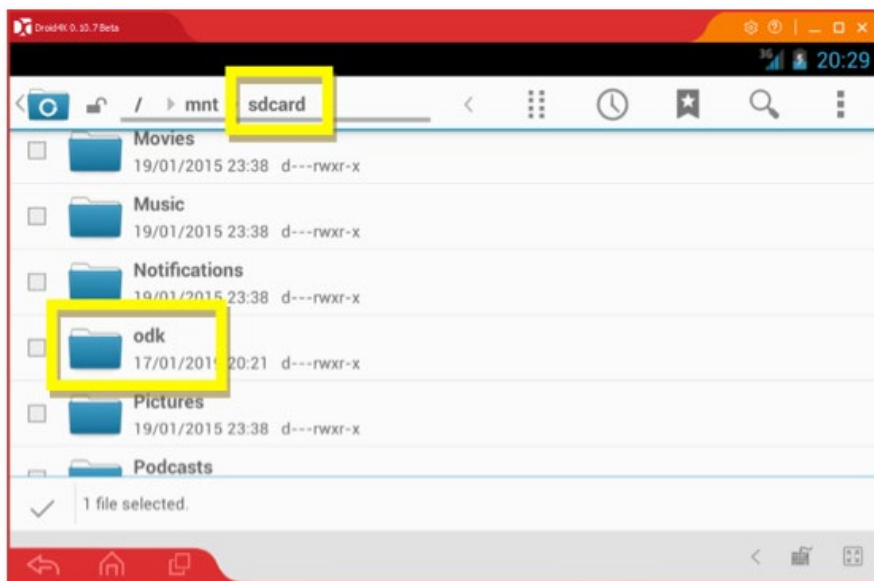
Select the **[forms]** folder.

Click the **eight-dots** button at the top.

<u>**Click**</u> '[**Copy selection here]'**.
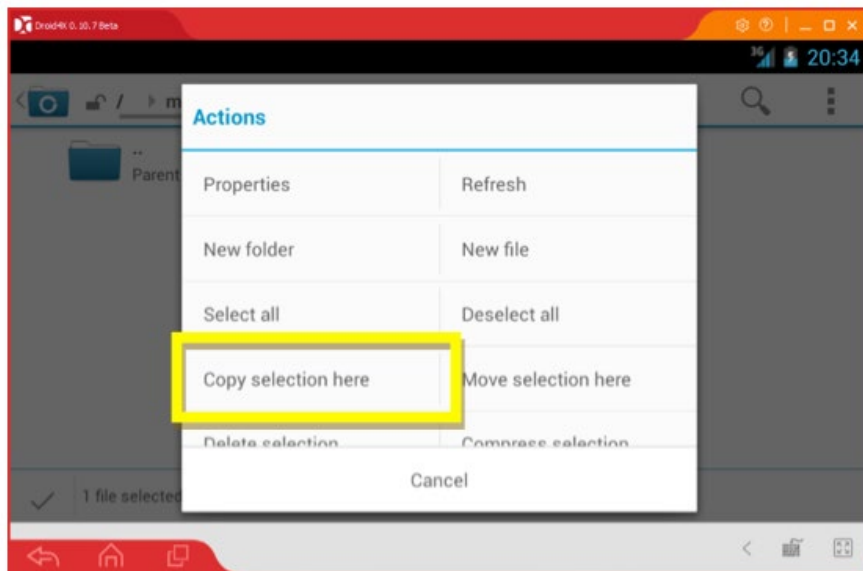
The file should now appear in the folder and you can return to the home screen.
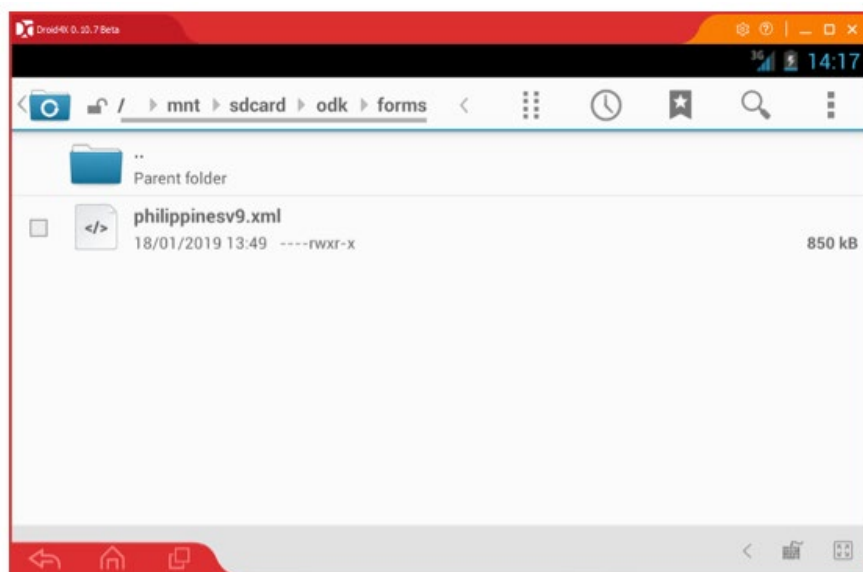
<u>**Launch**</u> ODK Collect and <u>**select**</u> **'[Fill blank form]'**. This will load the new form you have copied.

Note: You may have to go back to the main ODK Collect screen and select 'Fill Blank Form' again for it to appear.

CRVS technical guide

Section 4: Tablet configuration

4.1 Developer mode

All Android devices running version 4.x and above will need to be configured to turn on Developer Mode to allow the program to access the ODK Files on the tablet via USB cable.

Note: Unplug your tablet from the PC *before* you enable these settings and ensure you have restarted your PC.

Each tablet will have slightly different methods to achieve this, but most of the newer devices will have a **Developer Options** area in the **Settings** of the device.

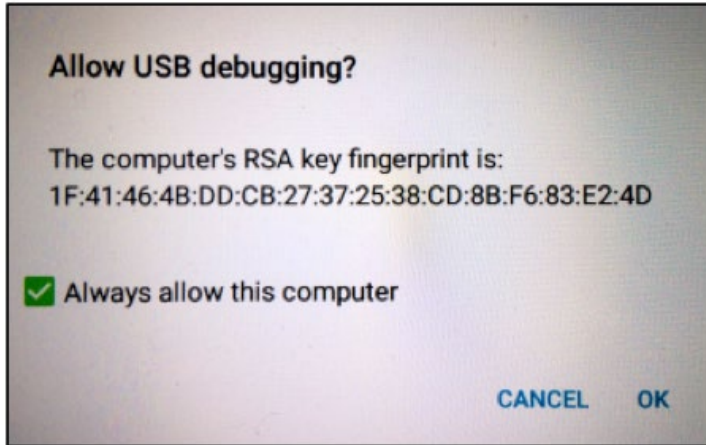Go into **[Settings] > [Developer Options]** and ensure that Developer Mode is turned on, and that USB Debugging is also enabled.

The below is an example from a Samsung tablet.
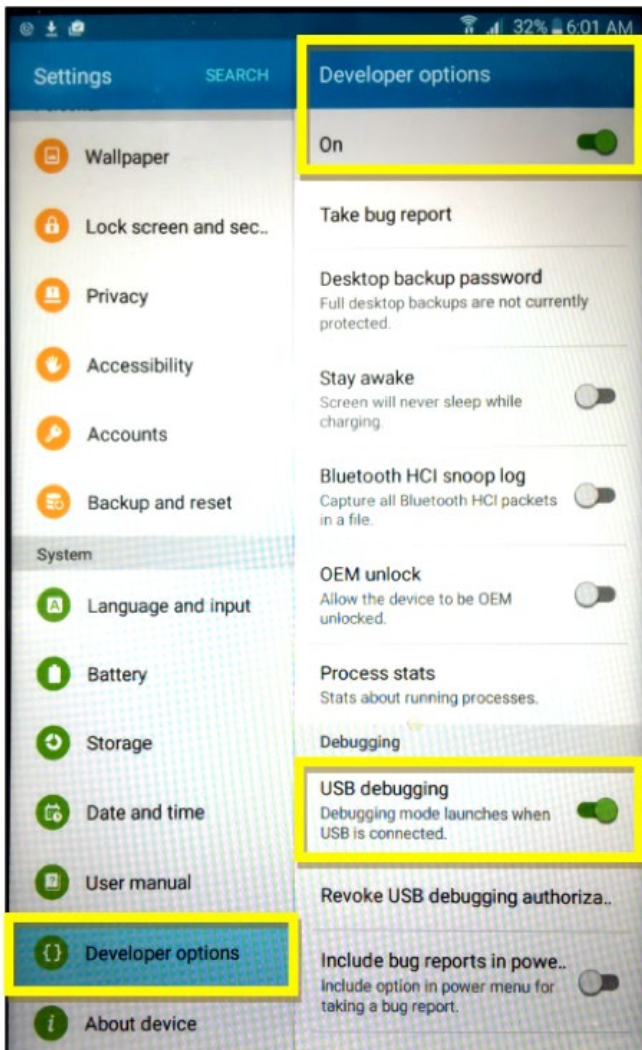
Once you have enabled this, plug the device into your PC/Laptop with the USB Cable. You should be prompted to allow USB debugging from your PC.

<u>Select</u> **'[Always allow this computer]'** and <u>click</u> **[OK].**

Your tablet is now connected and ready to run the program. These steps are only required once and in future you can simply connect the device and run the program as normal.

If you are unable to locate the Developer Options on your device, another common method is the following:

- Open Settings > About > Software Information > More;
- Tap "Build number" seven times to enable Developer options;
- Go back to the Settings menu and now you'll be able to see "Developer options" there;
- Tap it and turn on USB Debugging from the menu on the next screen.

Section 5: Running for the first time – USB connected device

Once you have updated the configuration file to reflect the required settings for your implementation (for example, set the **Device Type: setting to 'd'**, and set the device to Developer Mode), you are ready to run the script for the first time.

With your device connected via USB, start the program from the icon on your desktop.

If presented with the screen below, **press** **'[A]'** and then **'[Enter]'** to continue. This will only be required the first time, in most cases, to set the correct security settings on your computer to allow the program to run.
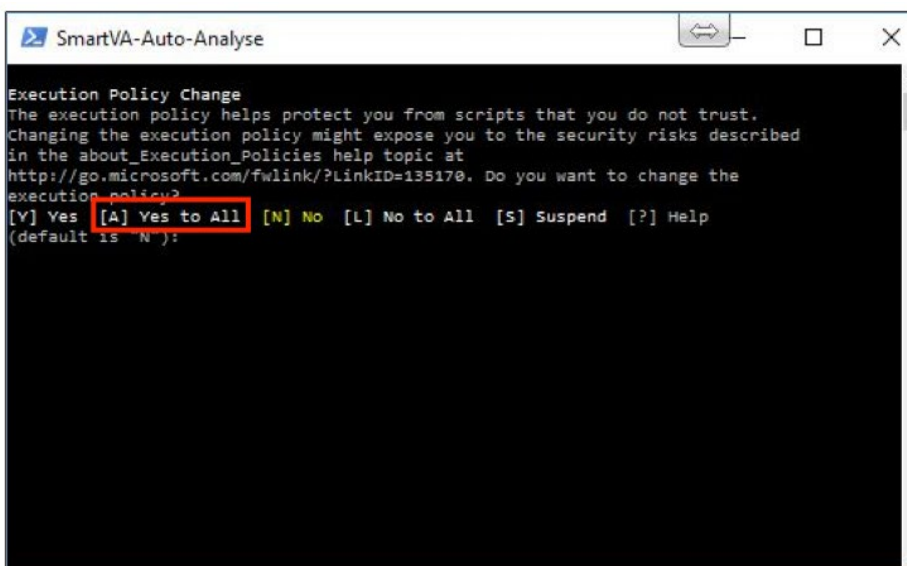
Section 6: Program outputs – what does it do?

6.1 Screen output

Assuming the configuration settings are correctly in place, once the program has begun, a progress bar will be updated to keep you informed of processing progress.
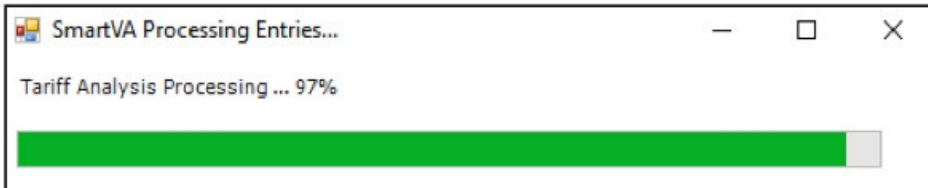

Upon completion, if you have chosen to display results (as set in the **config.txt** file) you will see a window displayed which will contain the first VA entry found on the tablet or Aggregate:
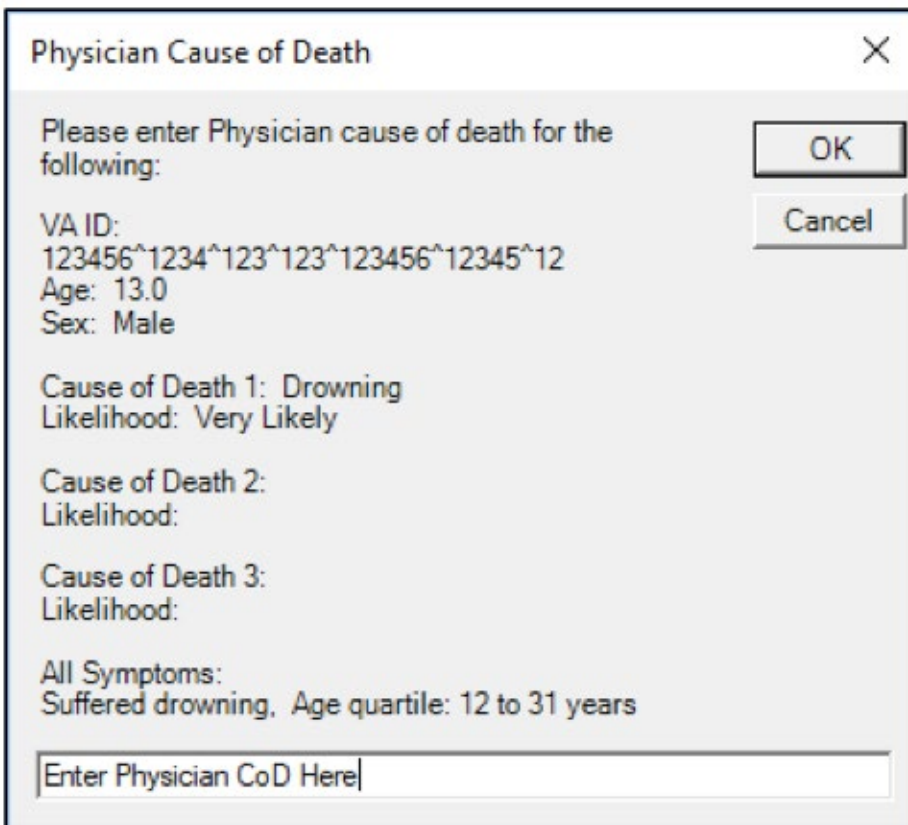
In the text box available at the bottom of the window, __enter__ the physician cause of death and __click__ **[OK]**. This process will be repeated for additional VAs processed from the tablet or aggregate server until there are no more VAs to process.

If you have chosen not to display results, then a confirmation box will show to advise how many records were processed and added to the **results.csv** file in the archive.
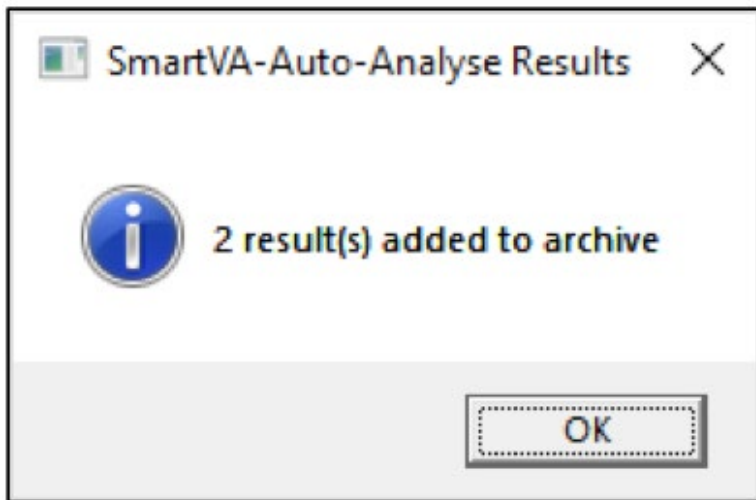
6.2 File outputs

Archive folder

The archive folder is a location that is used to store two key pieces of information:

1. Copy of raw data from the tablet once successfully processed;
2. **results.csv** file.

Note: A copy of the raw data will not be copied if using an aggregate server. It is assumed that the server infrastructure will be backed up in accordance with best practices by the owners/managers of the server.

Once the program has completed successfully, the underlying raw data for each result is copied to the archive, so that it can be used to perform future analysis as required (using the manual analysis method).

The archive folder location can be specified via the **config.txt** file (see section 2.2 [Settings] of this user guide for further information).

It should be noted that only data that has processed successfully will be copied to this archive, and therefore any errors in processing should be resolved to the point of successful program completion before any data is removed from the tablet/device.

results.csv file

In addition to the output to screen, the program also outputs the results to a file called **results.csv.** This file is located in the **Archive** folder, which can be found as a sub-folder of the location set in the **config.txt** file under **ArchiveDir** (see section 2.2 [Settings] of this user guide for more information about locations). For example, the default location for archiving, as configured in the **config.txt** file, is **C:\SmartVA**.

If you navigate to this folder, you will see a sub-folder called **Archive**, and within that folder is the file **results.csv**.

This file is useful if you need to review the results of previous analyses performed, or if you have closed the screen output and wish to review the results again. This will include the results of all VA you have processed through Auto-Analyse on this machine. The screenshots below show the file in the default location, and what its contents look like when opened in Microsoft Excel.
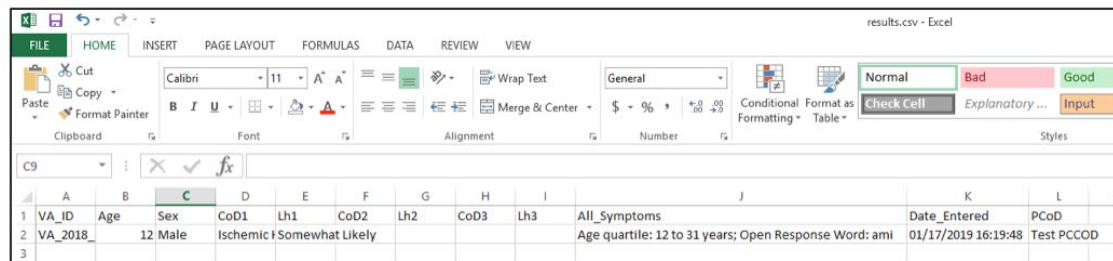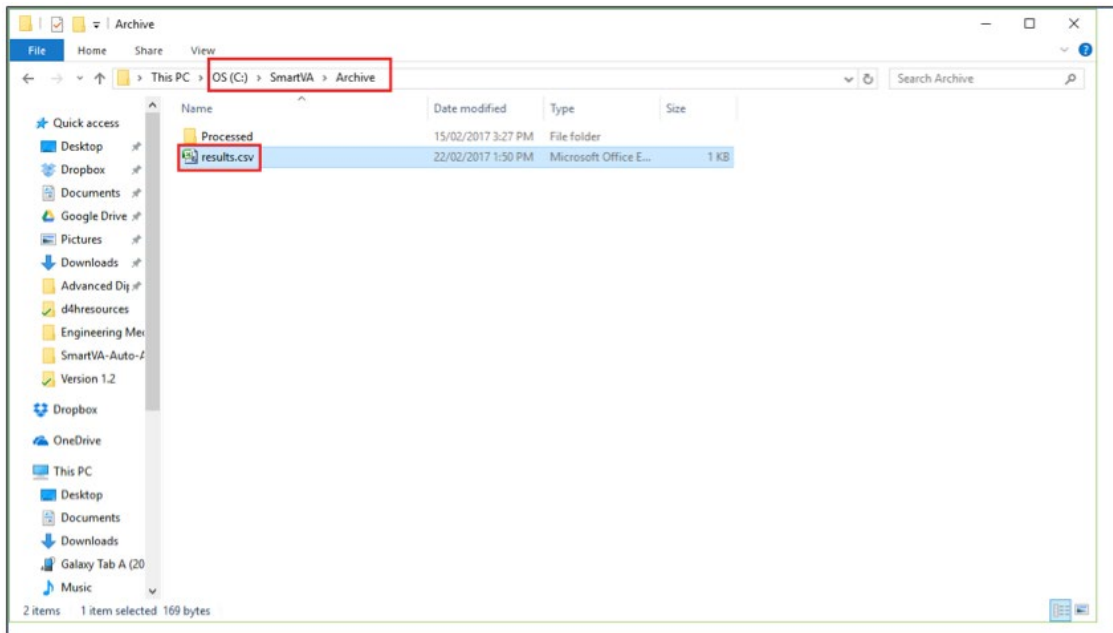
Log files

When the program runs, all component parts (file copy from tablet, ODK Briefcase conversion and Tariff output) generate log files that can be reviewed if issues should arise. The **Logs** folder, contained within the **ProcessDir**, contains the following:

log.txt
This is the program log file and contains entries as each stage of the program runs/completes. Any error messages that are displayed to screen are also written to this file to aid with troubleshooting. This log file should be checked first in case of any issues experienced.

adblog.txt
This log file contains information specific to the file copy from the tablet and can be used to further troubleshoot issues at this stage of the program run.

BriefcaseOutput.txt
This log file contains information specific to the ODK Briefcase conversion process and any issues with this stage of the program run will be logged here.

TariffOutput.txt
This log file contains the specific information relating to the Tariff analysis stage of the program run. Any issues experienced with Tariff analysis will be logged here.

The program partners on this initiative include: The University of Melbourne, Australia; CDC Foundation, USA; Vital Strategies, USA; Johns Hopkins Bloomberg School of Public Health, USA; World Health Organization, Switzerland.

Civil Registration and Vital Statistics partners:

For more information contact:

CRVS-info@unimelb.edu.au
crvsgateway.info

unimelb.edu.au