

DigitalPersona, Inc.

U.are.U SDK

Version 2

Platform Guide for Android™



digitalPersona.

DigitalPersona, Inc.

© 2014 DigitalPersona, Inc. All Rights Reserved.

All intellectual property rights in the DigitalPersona software, firmware, hardware and documentation included with or described in this guide are owned by DigitalPersona or its suppliers and are protected by United States copyright laws, other applicable copyright laws, and international treaty provisions. DigitalPersona and its suppliers retain all rights not expressly granted.

U.are.U® and DigitalPersona® are trademarks of DigitalPersona, Inc. registered in the United States and other countries. Android is a trademark of Google Inc. Java is a registered trademark of Oracle and/or its affiliates. All other trademarks are the property of their respective owners.

This DigitalPersona U.are.U SDK Platform Guide for Android and the software it describes are furnished under license as set forth in the "License Agreement". Except as permitted by such license, no part of this document may be reproduced, stored, transmitted and translated, in any form and by any means, without the prior written consent of DigitalPersona. The contents of this manual are furnished for informational use only and are subject to change without notice.

Any mention of third-party companies and products is for demonstration purposes only and constitutes neither an endorsement nor a recommendation. DigitalPersona assumes no responsibility with regard to the performance or use of these third-party products.

DigitalPersona makes every effort to ensure the accuracy of its documentation and assumes no responsibility or liability for any errors or inaccuracies that may appear in it.

Technical Support

To access DigitalPersona technical support resources, go to <http://www.digitalpersona.com/support>.

Phone support is available at (877) 378-2740 (US) or +1 650-474-4000 (outside the US).

Feedback

Although the information in this guide has been thoroughly reviewed and tested, we welcome your feedback on any errors, omissions, or suggestions for future improvements. Please contact us at

TechPubs@digitalpersona.com

or

DigitalPersona, Inc.
720 Bay Road, Suite 100
Redwood City, California 94063
USA
(650) 474-4000
(650) 298-8313 Fax

Table of Contents

1	Introduction	4
	Getting Updated Documentation	4
2	Installation	5
	Installing on the Development System	5
3	Developing Applications with C/C++	7
	Pre-Requisites	7
	System Requirements	7
	Development System	7
	Target Hardware (Android-Based Device)	7
	The Sample Applications	7
	Building the Sample Applications	8
	Running the UareUSample Application	8
	Running the UareUCaptureOnly Application	15
4	Developing Applications with Java	20
	Pre-Requisites	20
	System Requirements	20
	Development System	20
	Target Hardware (Android-Based Device)	20
	Installing on the Target Hardware	20
	The Java Sample Application	21
	Selecting a Reader	23
	Enrolling a Finger	25
	Identifying a Fingerprint	26
	Verifying a Fingerprint	27
	Using the Capture and Streaming Feature	28

This manual describes how to use the U.are.U SDK to develop applications that will run on Android™. The U.are.U SDK is available for multiple platforms and this document describes issues specific to developing applications that will run on Android-based platforms:

Chapter 1, *Introduction* (this chapter) describes how to get the latest version of this documentation.

Chapter 2, *Installation* provides instructions for installing on your development system and on the target (Android) reader.

Chapter 3, *Developing Applications with C/C++* lists system requirements for developing and running applications in C/C++ and describes the sample application.

Chapter 4, *Developing Applications with Java* lists system requirements for developing and running applications using Java, provides additional installation instructions and describes the Java sample application.

For a detailed description of the SDK, consult the U.are.U SDK Developer Guide.

Getting Updated Documentation

If you are viewing this guide from the download package for the U.are.U SDK, you may want to check online at our website for an updated version of this document at

<http://www.digitalpersona.com/Support/Reference-Material/DigitalPersona-SDK-Reference-Material/>

Except as noted in the platform/language-specific chapters, the installation process is the same for all Android development.

There are two steps to the installation:

1. Installing on the development system
2. Installing on the device (the target hardware)

These steps are described below.

Installing on the Development System

To install the SDK on an Intel-based Linux development system:

1. Unpack **DP-UareU-2.2.3-*<date,build>*.tar.gz** into a directory.
2. Go to the directory where you unpacked the files.
3. Run `setup`.
4. Follow the prompts as they appear.

The installer copies all necessary files to the selected folder. The files installed on the developer's machine are located in the following folders within the main product folder:

Folder	Contents
Include	Header files for C/C++ API.
Android/docs	End user license agreement (EULA) plus documentation: <ul style="list-style-type: none"> ■ SDK Developer Guide - describes all APIs ■ Platform Guide for Android - Android-specific details ■ C_API - Doxygen for C/C++ API ■ Java_API - Javadoc for Java API
Android/bin/android-14/arch-arm	Drivers for DigitalPersona fingerprint readers Runtime files for: <ul style="list-style-type: none"> ■ arm - C/C++ development for ARM processors ■ java - Java JAR file
Android/Samples	Source files for sample applications: <ul style="list-style-type: none"> ■ UareUSample - C/C++ sample ■ UareUCaptureOnly - C/C++ sample that demonstrates only capture ■ UareUSampleJava - Java sample
Android/Samples/bin	Executables for sample applications: <ul style="list-style-type: none"> ■ UareUSample - C/C++ sample ■ UareUCaptureOnly - C/C++ sample that demonstrates only capture ■ UareUSampleJava - Java sample

Pre-Requisites

This chapter assumes that you have a working knowledge of C/C++ and that you know how to develop for applications that will run on Android platforms and devices.

System Requirements

Development System

The development system can be any Intel-based Linux system. The Android Native Development Kit (NDK) is required to develop native (C/C++) applications and libraries.

Target Hardware (Android-Based Device)

The Android-based device must have Android 4.0 or better installed (API level 14 or better) and the UVC (also known as Video4Linux) kernel driver enabled. You must have root privileges on the Android device in order to develop and run C/C++ applications and samples. A USB port in host mode is required.

The file sizes are (in Kb):

	Android API (Level 14)
Capture runtime (drivers + SDK layer)	580
Fingerprint recognition runtime	460

In addition, the device must have 16 MB free memory.

The SDK works on a variety of hardware and is intended to have a small footprint so that it can run even on minimal hardware. Less capable hardware will work, but response time may not be optimal.

The Sample Applications

U.are.U SDK includes sample applications to demonstrate the features of the SDK. The sample applications are located in subfolders of the `/urusdk-android/Android/Samples` folder:

- The `UareUCaptureOnly` subdirectory contains the sources of the sample application that demonstrates fingerprint capture. This application does not have any dependencies on the FingerJet engine libraries.

- The `UareUSample` subdirectory contains the source for the sample application that demonstrates the full functionality of the SDK: fingerprint capture, feature extraction and verification, identification and enrollment of fingerprints. This application has dependencies on the capture and FingerJet Engine libraries.

Building the Sample Applications

`UareUSample` and `UareUCaptureOnly` are built in the same way. The makefile relies on the following environment variables: `NDK` and `TOOLCHAIN`. The default values are set assuming that the Android NDK is installed in the `$HOME/Android/android-ndk-r8d` directory and the development system is Linux-x86. Modify the makefile or set the environment variables according to the actual locations of the NDK directory and the toolchain directory. Once those are set, you can build the sample application with the `make` command.

Running the UareUSample Application

The `UareUSample` application demonstrates the Capture API and the FingerJet Engine API. It is a console application - it does not require any graphic libraries. However it requires root privileges in order to access `/dev/videoX` devices, and it requires the `adb` shell or an `ssh` connection to run.

In addition, the Gallery app (installed by default on most Android devices) must be disabled to avoid interruption of the sample. To disable the Gallery app, navigate on your device's display to `Settings->Apps->Gallery App` and click the **Disable** button.

To run the `UareUSample` application:

1. Establish the `adb` shell connection with the device:
 - `adb shell`
2. Type `su` to grant permission to the device
3. Create the directory where the DigitalPersona libraries and application executable will be copied:
 - `mkdir /data/dpsample`
4. Exit the root
 - `exit`
5. Exit the shell and copy the libraries and executable to the created directory:
 - `exit`
 - `adb push <UareU SDK folder>/urusdk-android/Android/bin/android-14/arch-arm/libdpfj.so /data/dpsample`
 - `adb push <UareU SDK folder>/urusdk-android/Android/bin/android-14/arch-arm/libdpfpdd5000.so /data/dpsample`
 - `adb push <UareU SDK folder>/urusdk-android/Android/bin/android-14/arch-arm/libdpfpdd.so /data/dpsample`

- `adb push <UareU SDK folder>/urusdk-android/Android/bin/android-14/arch-arm/libdpuvc.so /data/dpsample`
 - `adb push <UareU SDKfolder>/urusdk-android/Android/Samples/bin/UareUSample /data/dpsample`
6. Reestablish `adb` shell connection with the device and change access permissions on the reader device.
 7. Make sure that `LD_LIBRARY_PATH` points to the directory with SDK libraries:
 - `adb shell`
 - `cd /data/dpsample`
 - `LD_LIBRARY_PATH=/data/dpsample ./UareUSample`

When you launch the application you will see the main menu, as shown below:

```
root@android:/data/dpsample # LD_LIBRARY_PATH=/data/dpsample/ ./UareUSample
```

```
UareU SDK 2.x sample application (verification, identification, enrollment)
```

- 1: Select new reader (not selected)
- 2: Run verification
- 3: Run identification
- 4: Run enrollment

```
Enter 1 - 4, or 'E' to exit: █
```

You must begin by selecting a fingerprint reader device to use with the application. After you press **1** and Enter to select the reader, the application will search for any fingerprint readers and display a list of the readers it

finds, as shown below. Choose the desired reader and press Enter to continue.

```
root@android:/data/dpsample # LD_LIBRARY_PATH=/data/dpsample/ ./UareUSample
```

```
UareU SDK 2.x sample application (verification, identification, enrollment)
```

- 1: Select new reader (not selected)
- 2: Run verification
- 3: Run identification
- 4: Run enrollment

```
Enter 1 - 4, or 'E' to exit: 1
```

```
Available readers:
```

```
  /dev/video0
```

```
Reader selection
```

- 1: Select /dev/video0
- 2: Refresh reader list

```
Enter 1, 2, or 'B' to go back:
```

Once a reader is selected, you will see a screen that shows information about the reader and its capabilities, as shown below.

```
Selected reader: /dev/video0
Vendor name:    DigitalPersona, Inc.
Product name:   Civil PIV Sensor
Serial number:  {de2ac125-dec3-449d-9e20-801d074a1bb0}
USB VID:       05ba
USB PID:       000b
USB BCD revision: 0001
HW version:    0.1.0
FW version:    0.1.0

can capture image:    1
can stream image:    1
can extract features: 0
can match:            0
can identify:         0
has fingerprint storage: 0
indicator type:      29
has power management: 0
has calibration:     0
PIV compliant:       1
resolution:          500 dpi

UareU SDK 2.x sample application (verification, identification, enrollment)

1: Select new reader (selected: /dev/video0)
2: Run verification
3: Run identification
4: Run enrollment

Enter 1 - 4, or 'E' to exit:
```

Now let's test verification. To do that, press **2** and press Enter.

Follow the prompts on the screen as shown below and touch your finger on the reader. You will see that the program extracts the fingerprint features. Now touch the reader again and the sample application will extract the features of the second fingerprint and determine whether the fingerprints match or not.

```
UareU SDK 2.x sample application (verification, identification, enrollment)
```

```
1: Select new reader (selected: /dev/video0)
2: Run verification
3: Run identification
4: Run enrollment
```

```
Enter 1 - 4, or 'E' to exit: 2
Verification started
```

```
Put any finger on the reader, or press Ctrl-C to cancel...
fingerprint captured,
features extracted.
```

```
Put the same or any other finger on the reader, or press Ctrl-C to cancel...
fingerprint captured,
features extracted.
```

```
Fingerprints matched.
```

```
dissimilarity score: 0x0.
false match rate: 0.000000e+00.
```

```
Verification started
```

```
Put any finger on the reader, or press Ctrl-C to cancel...
█
```

Try touching your index finger as the first fingerprint and your middle finger as the second fingerprint and note that they do not match.

Press Ctrl-C to return to the main menu.

Now let's test identification. Press **3** and Enter to start.

Remember that identification matches a single fingerprint against a set of fingerprints. You will be prompted to touch each of your fingers on the reader in turn, as shown below. When you have entered all of your fingerprints, you will be prompted to touch any finger on the reader. Try touching one of your fingers to ensure that the fingerprint matches. Try touching the reader with a finger from your other hand and ensure that it does NOT match.

```
Enter 1 - 4, or 'E' to exit: 3
Identification started

Put your thumb on the reader, or press Ctrl-C to cancel...
  fingerprint captured,
  features extracted.

Put your index finger on the reader, or press Ctrl-C to cancel...
  fingerprint captured,
  features extracted.

Put your middle finger on the reader, or press Ctrl-C to cancel...
  fingerprint captured,
  features extracted.

Put your ring finger on the reader, or press Ctrl-C to cancel...
  fingerprint captured,
  features extracted.

Put any finger on the reader, or press Ctrl-C to cancel...
  fingerprint captured,
  features extracted.

Fingerprint identified, your ring finger
dissimilarity score: 0x0.
false match rate: 0.000000e+00.

Identification started

Put your thumb on the reader, or press Ctrl-C to cancel...
```

Press Ctrl-C to return to the main menu.

Let's test enrollment now. Press **4** and Enter to start.

The enrollment test is to enroll a single finger. Follow the prompts on the screen (as shown below) and touch your finger on the reader when prompted. Then when prompted, touch the same finger on the reader again. It typically takes four scans to get a good enrollment. If you touch some other finger on the reader during enrollment, or if a scan is of poor quality, that scan will be dropped and the application will request additional finger scans in order to do a complete enrollment.

```
UareU SDK 2.x sample application (verification, identification, enrollment)
```

```
1: Select new reader (selected: /dev/video0)
2: Run verification
3: Run identification
4: Run enrollment
```

```
Enter 1 - 4, or 'E' to exit: 4
Enrollment started
```

```
1
Put any finger on the reader, or press Ctrl-C to cancel...
  fingerprint captured,
  features extracted.
```

```
2
Put the same finger on the reader, or press Ctrl-C to cancel...
  fingerprint captured,
  features extracted.
```

```
3
Put the same finger on the reader, or press Ctrl-C to cancel...
  fingerprint captured,
  features extracted.
```

```
4
Put the same finger on the reader, or press Ctrl-C to cancel...
  fingerprint captured,
  features extracted.
```

```
Enrollment template created, size: 440
```

```
Enrollment started
```

```
Put any finger on the reader, or press Ctrl-C to cancel...
█
```

Once you have finished testing the enrollment feature of the sample application, press Ctrl-C to return to the main menu.

This completes the demonstration of the `UareUSample` application. Press E to exit from the main menu.

Running the UareUCaptureOnly Application

The `UareUCaptureOnly` application demonstrates the Capture API alone (no dependencies on the FingerJet Engine library). It is a console application - it does not require any graphic libraries. However it requires root privileges in order to access `/dev/videoX` devices, and it requires `adb shell` or an `ssh` connection to run.

To run the `UareUCaptureOnly` application:

1. Establish an `adb shell` connection with the device:
 - `adb shell`
2. Type `su` to grant permission to the device
3. Create the directory where the DigitalPersona libraries and application executable will be copied:
 - `mkdir /data/dpsample`
4. Exit root:
 - `exit`
5. Exit the shell and copy the libraries and executable to the created directory:
 - `exit`
 - `adb push <UareU SDK folder>/urusdk-android/Android/bin/android-14/arch-arm/libdppfpdd5000.so /data/dpsample`
 - `adb push <UareU SDK folder>/urusdk-android/Android/bin/android-14/arch-arm/libdppfpdd.so /data/dpsample`
 - `adb push <UareU SDK folder>/urusdk-android/Android/bin/android-14/arch-arm/libdpuvc.so /data/dpsample`
 - `adb push <UareU SDK folder>/urusdk-android/Android/Samples/bin/UareUCaptureOnly /data/dpsample`
6. Reestablish the `adb shell` connection with the device and change access permissions on the reader device.
7. Make sure that `LD_LIBRARY_PATH` points to the directory with the SDK libraries:
 - `adb shell`
 - `cd /data/dpsample`
 - `LD_LIBRARY_PATH=/data/dpsample ./UareUCaptureOnly`

When you launch the application you will see the main menu, as shown below:

```
root@android:/data/dpsample # LD_LIBRARY_PATH=/data/dpsample/ ./UareUCaptureOnly

UareU SDK 2.x sample application (capture)

  1: Select new reader (not selected)
  2: Capture fingerprints
  3: Stream fingerprints

Enter 1 - 3, or 'E' to exit:
```

You must begin by selecting a fingerprint reader device to use with the application. After you press **1** to select the reader, the application will search for any readers and display a list of any readers it finds, as shown below. Choose the desired reader and press Enter to continue.

```
UareU SDK 2.x sample application (capture)

  1: Select new reader (not selected)
  2: Capture fingerprints
  3: Stream fingerprints

Enter 1 - 3, or 'E' to exit: 1

Available readers:
  /dev/video0

Reader selection

  1: Select /dev/video0
  2: Refresh reader list

Enter 1, 2, or 'B' to go back: █
```

Once a reader is selected, you will see a screen that shows information about the reader and its capabilities, as shown below.

```
Enter 1, 2, or 'B' to go back: 1

Selected reader: /dev/video0
Vendor name:    DigitalPersona, Inc.
Product name:   Civil PIV Sensor
Serial number:  {de2ac125-dec3-449d-9e20-801d074a1bb0}
USB VID:       05ba
USB PID:       000b
USB BCD revision: 0001
HW version:    0.1.0
FW version:    0.1.0

can capture image:    1
can stream image:    1
can extract features: 0
can match:            0
can identify:         0
has fingerprint storage: 0
indicator type:      29
has power management: 0
has calibration:     0
PIV compliant:       1
resolution:          500 dpi

UareU SDK 2.x sample application (capture)

1: Select new reader (selected: /dev/video0)
2: Capture fingerprints
3: Stream fingerprints

Enter 1 - 3, or 'E' to exit:
```

Select **2** and press Enter to test the capture function of the sample application. The capture function simply puts the reader into fingerprint capture mode and waits until a fingerprint is detected, as shown in the picture below.

```
UareU SDK 2.x sample application (capture)
  1: Select new reader (selected: /dev/video0)
  2: Capture fingerprints
  3: Stream fingerprints

Enter 1 - 3, or 'E' to exit: 2
1
Put your finger on the reader, or press Ctrl-C to cancel...
  fingerprint captured
  width: 252, height: 324

2
Put your finger on the reader, or press Ctrl-C to cancel...
  fingerprint captured
  width: 252, height: 324

3
Put your finger on the reader, or press Ctrl-C to cancel...
  fingerprint captured
  width: 252, height: 324

4
Put your finger on the reader, or press Ctrl-C to cancel...
  fingerprint captured
  width: 252, height: 324

5
Put your finger on the reader, or press Ctrl-C to cancel...
█
```

Once you have finished testing the capture feature of the sample application, press Ctrl-C to return to the main menu.

To test the streaming feature (not available on all fingerprint readers), select **3** and press Enter to test the streaming function of the sample application. The streaming function simply puts the reader into fingerprint stream mode and waits until a fingerprint is detected, as shown in the picture below.

```
UareU SDK 2.x sample application (capture)

1: Select new reader (selected: /dev/video0)
2: Capture fingerprints
3: Stream fingerprints

Enter 1 - 3, or 'E' to exit: 3
1 image captured, score: 35
2 image captured, score: 35
3 image captured, score: 34
4 image captured, score: 35
5 image captured, score: 36
6 image captured, score: 109
7 image captured, score: 117
8 image captured, score: 119
9 image captured, score: 116
10 image captured, score: 116
11 image captured, score: 117
12 image captured, score: 117
13 image captured, score: 115
14 image captured, score: 113
15 image captured, score: 115
16 image captured, score: 114
17 image captured, score: 116
18 image captured, score: 114
19 image captured, score: 127
20 image captured, score: 31
21 image captured, score: 31
22 image captured, score: 30
23 image captured, score: 31
24 image captured, score: 29
25 image captured, score: 30
26 image captured, score: 29
27 image captured, score: 31
28 image captured, score: 31
29 image captured, score: 31
30 image captured, score: 31
```

As you press your finger on the reader, images will be captured and their quality score displayed. Press E to exit streaming mode.

This completes the demonstration of the `UareUCaptureOnly` application. Press E to exit from the main menu.

Pre-Requisites

This chapter assumes that you have a working knowledge of Java and that you know how to develop for Android readers.

System Requirements

Development System

The development system may be a Windows or Linux system. The Android SDK is required to develop for Android. We also recommend that you download and use the Eclipse version that is included with the Android SDK.

Target Hardware (Android-Based Device)

The Android-based device must have Android 4.0 or better installed (API level 14 or better) and the UVC (also known as Video4Linux) kernel driver enabled. A USB port in host mode is required. You don't have to have root privileges on the Android device.

The Android-based reader that will run the application must be one of the following hardware platforms:

- Nexus 7 with Android 4.2.1 (Jellybean)

The file sizes are (in Kb):

	Java SDK Layer
Capture runtime (drivers + SDK layer) with fingerprint recognition	47

The SDK works on a variety of hardware and is intended to have a small footprint so that it can run even on minimal hardware. Less capable hardware will work, but response time may not be optimal.

Installing on the Target Hardware

To install the run-time environment on the target hardware platform:

1. Enable USB debugging mode by navigating on your device's display to Settings->DeveloperOptions->Enable USB debugging.
2. Connect the target hardware platform to the development system.
3. Establish an adb connection with the device.

```
4. adb install <UareU SDK folder>/urusdk-android/Android/Samples/bin/  
   UareUSampleJava.apk
```

Installing the .apk file will install the following into its own application zone:

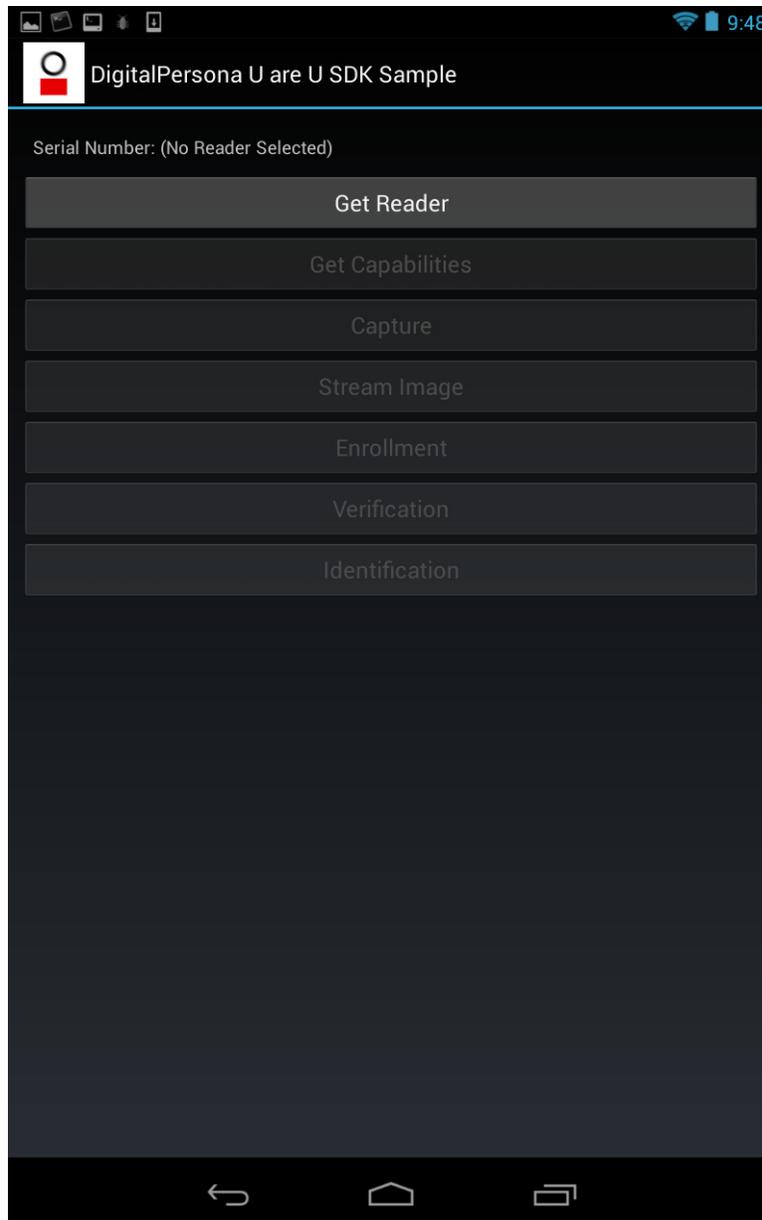
- Driver and engine libraries
- Java SDK wrapper
- Java Sample Application

If you install any of those components elsewhere on the target, you will have duplicate files installed. The intended architecture of Android is for all application to have all dependencies secured within their own application zones. All dependencies, UIs and services are installed and started automatically.

The Java Sample Application

U.are.U SDK includes a sample application to demonstrate the features of the SDK when using the Java API. The sample application is located in the `Samples` folder.

The application demonstrates the features of the SDK. When you launch the application, you see the main screen as shown below.



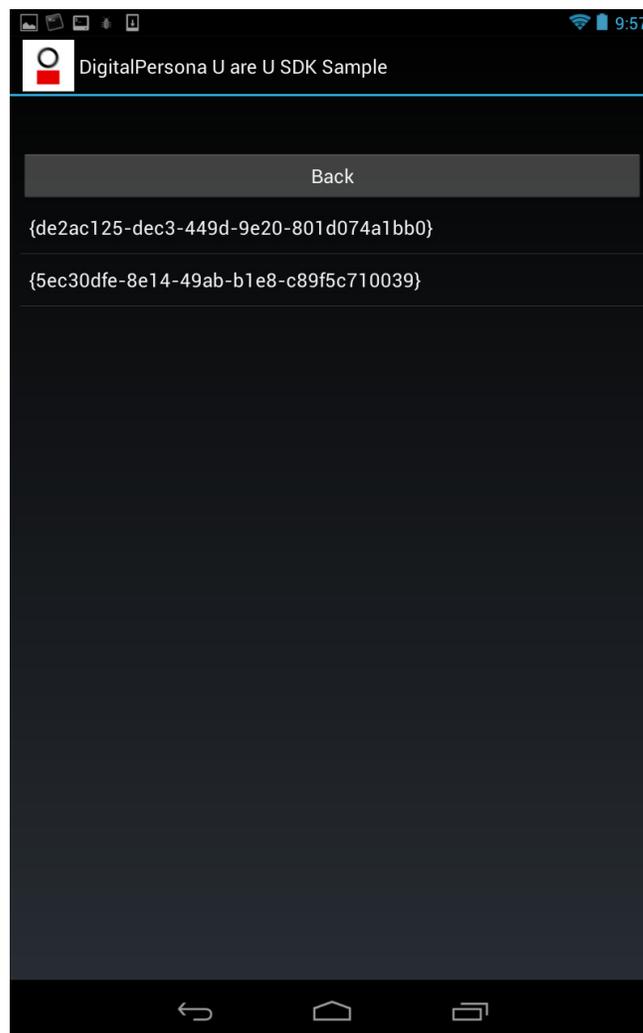
The sample program demonstrates:

- How to enroll a subject finger
- How to identify a fingerprint
- How to verify a fingerprint

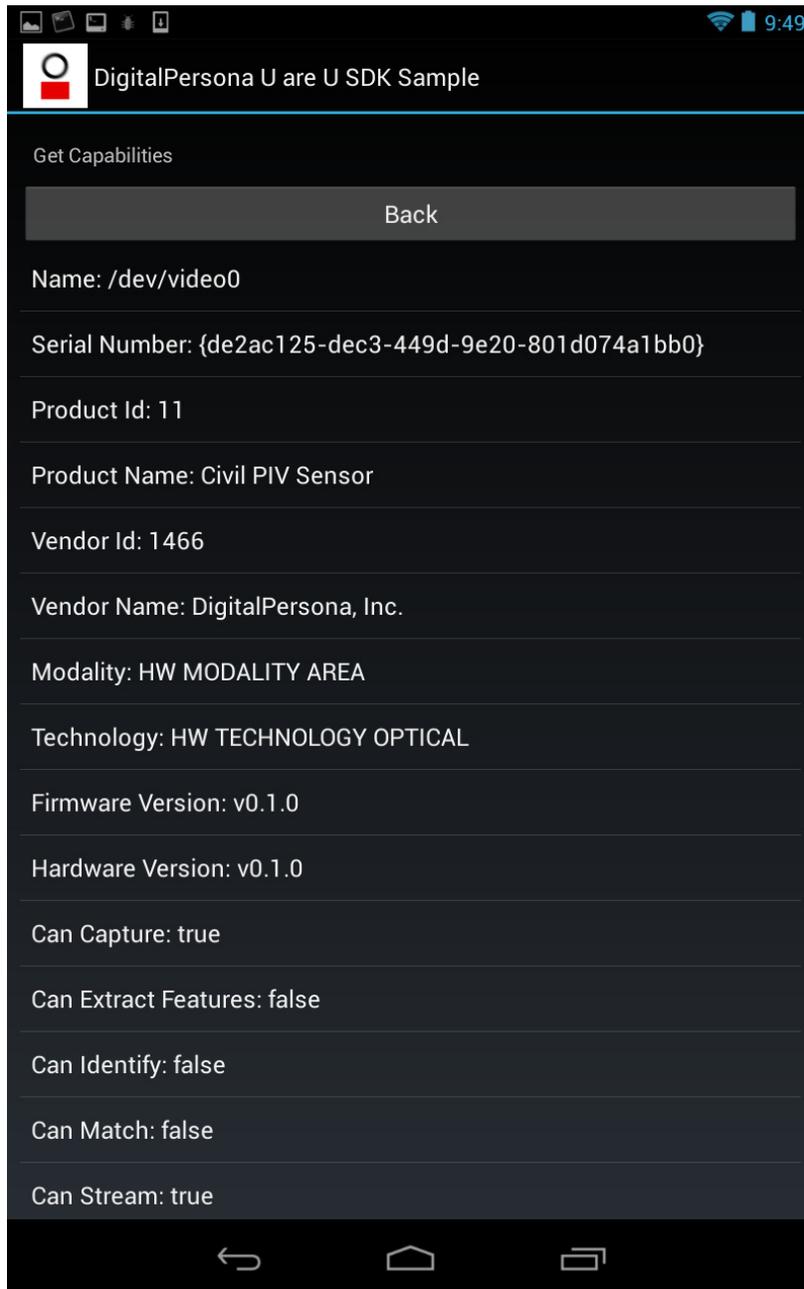
- The built-in control for enrollment
- The built-in control for identification
- How to use the streaming feature to display live fingerprint data on the screen

Selecting a Reader

To choose the reader, click on the **Get Reader** button. If there is only one reader attached, that reader will be selected automatically. If there are multiple readers, you will see a list of available readers and you can choose the desired device, as shown below. Click on the desired device and click on the **Back** button.



To see the reader capabilities, click on the **Get reader capabilities** button. The capabilities will be displayed, as shown in the image below.

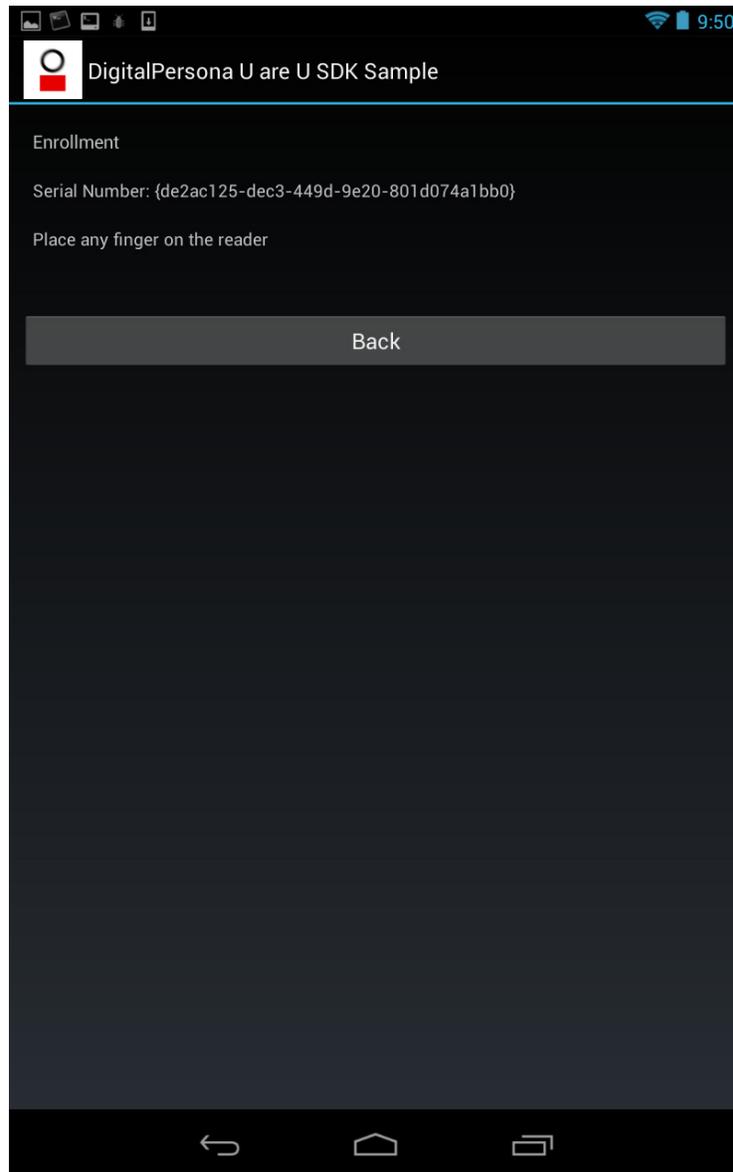


Click on the **Back** button to return to the main screen.

Enrolling a Finger

Click on **Enrollment** to begin enrolling a test subject.

You will see a series of prompts to scan fingers for enrollment, as shown below.



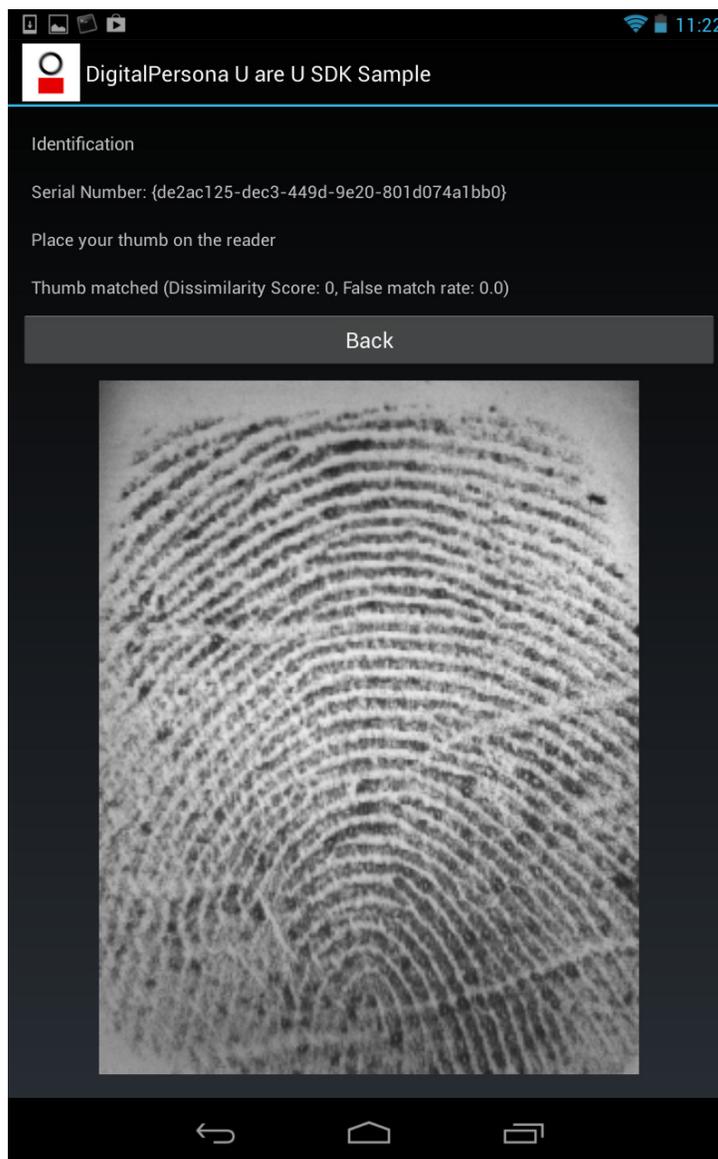
After the first finger is successfully scanned, you will be prompted to scan additional fingers until a sufficient number of high quality scans are complete. The number of fingers requested will vary depending on the image scans - the enrollment functions will continue to request scans until an acceptable enrollment record has been created.

When enrollment is complete, click **Back** to return to the main screen.

Identifying a Fingerprint

To test the identification feature, click on the **Identification** button. Recall that identification is a 1-to-many comparison where the application searches through all of the enrolled fingers to find a match. For this example, we do not have a stored database, so the sample application first prompts you to put fingers on the reader so that it can create a FMD. After a FMD is created, you will be prompted to put any finger on the reader to identify against the FMD that was just created.

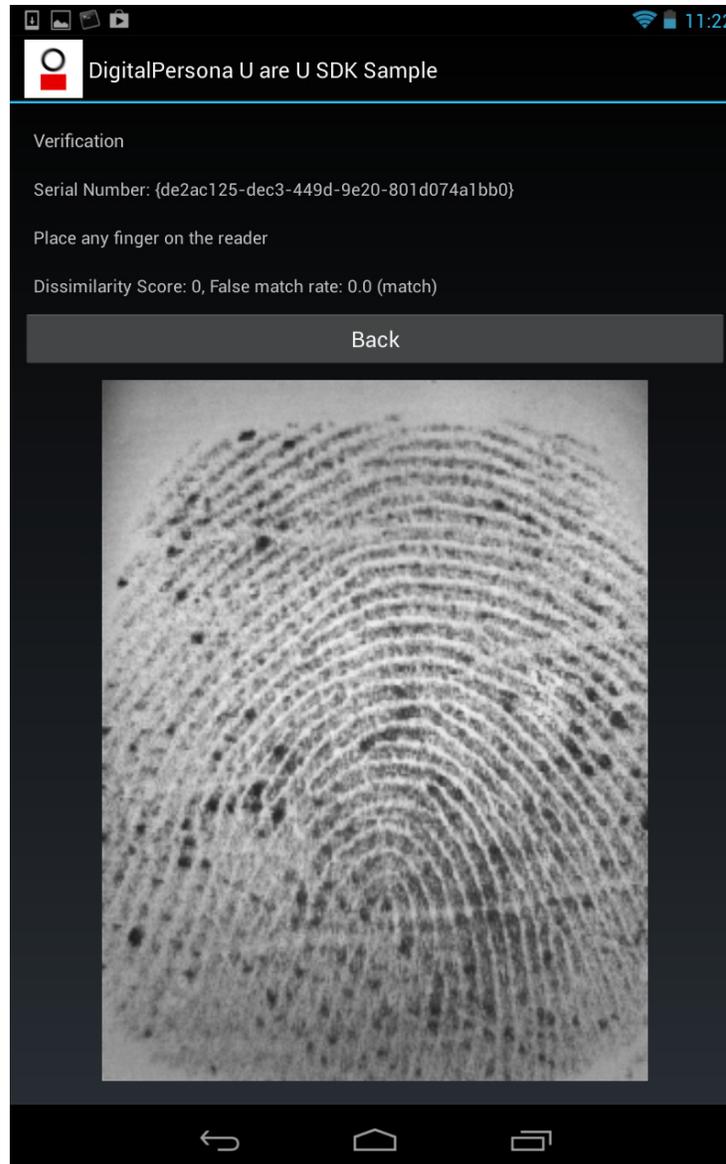
When you click the **Identification** button, you will be prompted to place your fingers on the reader. If you press an enrolled finger on the reader, you will see that a match was found. In the screen image below, we successfully identified a user.



To exit identification mode, click on the **Back** button.

Verifying a Fingerprint

To test the verification feature, click on the **Verification** button. Recall that verification is a 1-to-1 comparison where the application matches against a specified fingerprint. When you click the **Verification** button, you will be prompted to place your finger on the reader. As with the identification example above, in the screen below, we have successfully verified a user.

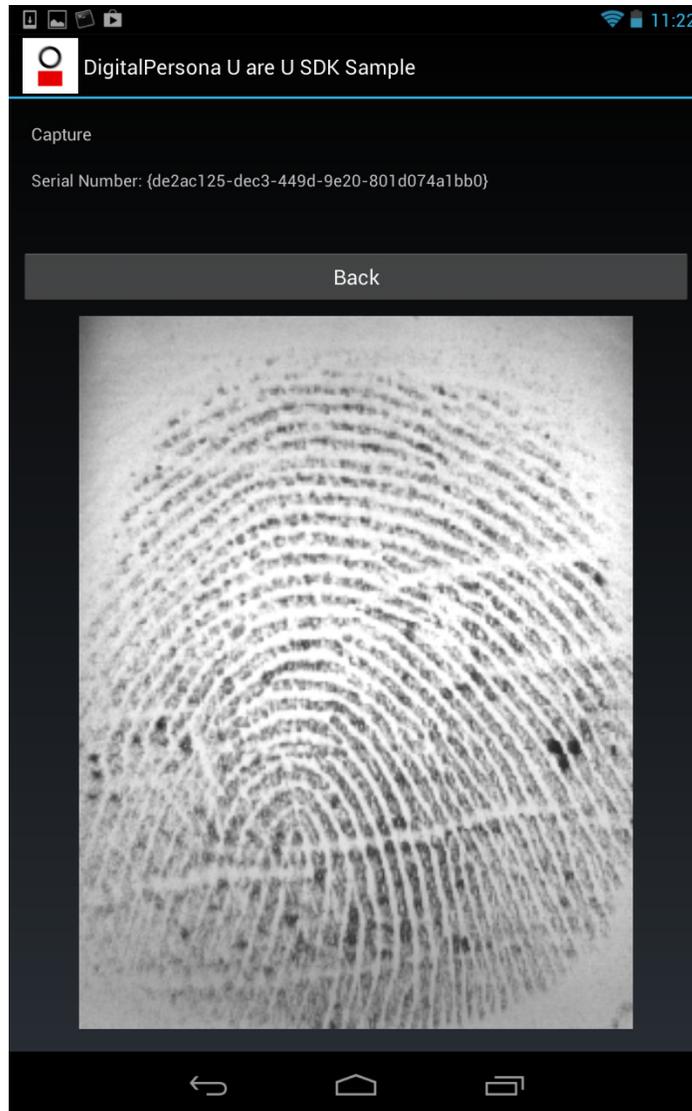


To exit verification mode, click on the **Back** button.

Using the Capture and Streaming Feature

The sample application also demonstrates the streaming feature (on fingerprint readers that support that feature). To test capturing or streaming, from the main window, click on the **Capture** or **Stream Image** button.

This puts the reader into capture/streaming mode and immediately the results are displayed in the window. For streaming mode, the window then becomes like a live window on the reader as it streams results. Placing a finger on the reader displays the streamed fingerprint, as shown below.



For streaming, removing the finger shows a blank stream.

To exit capture / streaming mode, click on **Back**.