

# **Android Development Guide**

Aptean Ltd  
Copyright © 2011-2026.

# Contents

<b>1 Android Development Guide</b> .....	<b>1</b>
1.1 Android Development.....	1
1.2 Testing.....	2
1.3 Releasing.....	2

# 1 Android Development Guide

## 1.1 Android Development

Android development is supported by the Barcoders ABS Wavelink replacement.

This consists of 2 classes to embed in the application

- ABSConstants.bas
- ABSStudio.cls

The class ABSStudio.cls creates every object and method that would have been created by WavelinkOLE.tlb.

Therefore, whenever these objects are instantiated, the code must now instantiate as ABSStudio rather than the separate Wavelink classes, for example:

Original code:

```
Private termIO As New RFTerminal
Private rfMainMenu As New RFMenu
Private rfConsole As New RFIO
```

Becomes:

```
Private termIO As New ABSStudio
Private rfMainMenu As New ABSStudio
Private rfConsole As New ABSStudio
```

The RDTMenu1 Project has been modified to include a number of new compile switches, which are accessible through VersionMaker:

- ANDROID - part of the Simul group. This controls swapping Wavelink, Debug and Android includes when building.
- ABS\_LOGGING - additional logging for every call to an ABSSyudio method, including the call parameters.
- TALKIE - controls whether the app makes use of ABSSpeak to speak text labels. This is a tri-state flag:
  - ◆ 0 - disabled
  - ◆ 1 - Enabled for RFEError and DialogBox text
  - ◆ 2 - Enabled for text labels when entering an InputField as part of an InputScreen

All changes to the code in order to support ABSStudio are wrapped in the ANDROID compile switch. Search for ANDROID to find the changes.

Most changes are in the RDT\_IO module (RDTInput.bas).

ABSStudio does not support line-drawing characters, so ensure that the system is set to Block characters to ensure that this works as expected.

ABSStudio does not return errors like Wavelink, and so this code has been removed for Android (RDT\_IO.ErrorAction)

ABSStudio does not support hiding the cursor "off-screen" and therefore this code is removed for Android (RDT\_IO.HideCursor)

ABSStudio does not support or need Flush Output and so this code is removed for Android (RDT\_IO.FlushOutput)

ABSStudio supports only 1 menu object, which is automatically reused for every menu displayed. As Wavelink supports multiple named menus, the code has been modified fairly extensively to control rebuilding of the main menu if any sub-menu has been displayed (RDT\_IO.ResetMainMenu and MainProcess.Main and ResetMainMenu)

ABSStudio does not support menu coordinates and so this code is removed for Android (RDT\_IO.MenuSetCoordinates)

ABSStudio does not support screen timeouts and so this code is removed for Android (RDT\_IO.Setup)

ABSStudio returns semi-colon instead of CR or NL on get event and so this code is changed for Android (RDT\_IO.GetEvent)



## 1.2 Testing

The changes for Android are at the same level as Wavelink and Wavelinkesque and can be tested as such - new modular functionality should not require specific Android device testing after the product is bedded in.

## 1.3 Releasing

Releasing should be through the same VersionMaker build and release process that is use today - build the Android version fo the app to "RDTMenu1\_Android.Exe". Modify your VesionMaker configuration file for this, as per the example below):

```
[Id]
Id{T}Site
[Major Vars]
Simul{O}RDT(WaveLinkOLE.tlb[HKEY_LOCAL_MACHINE\SOFTWARE\Classes\TypeLib\{08105893-8012-11D0-82BE-444553540000}
ESQUE(WLesque.dll[HKEY_LOCAL_MACHINE\SOFTWARE\Classes\TypeLib\{3692ACB7-79BB-4112-8FE3-AEE2EB8A2F2D}\3.0\0\0\0]
ANDROID(=RDTMenu1_Android.exe)
[Minor Vars]
Logging{C}LOGGING,LOGGING_WINDOW,LATE_BACKOUT
Android-Specific{C}ABS_LOGGING,TALKIE
```

Note that the release script has been modified to support the application being build in this way with this naming convention. This script will also require updating to any new installations of WCS, and to any existing installations that will start to use ABSStudio.

 **Note:** Upgrades to our software requiring upgrade to Barcodes ABS Server is an exceptional process.

The install to both the Android ABS Studio Server and Android client must be rolled in or back manually.

Never upgrade the ABS Server or Android app unless required by an upgrade.

