

***Fiscal eposprint for Android Sample Project
Guide***

REVISION HISTORY

A	All	Preliminary.	22 nd July 2013	P. Barnett
---	-----	--------------	----------------------------	------------

Index

- 1 Overview 4
 - 1.1 Description 4
 - 1.2 List of Related Documents 4
 - 1.3 Abbreviations..... 4
 - 1.4 Restrictions of Use 4
- 2 Restaurant Sample Program..... 5
 - 2.1 Restaurant Android Import 5
 - 2.2 Environment Setting Procedure..... 5

1 Overview

1.1 Description

The aim of this manual is to provide information to software designers who wish to make use of the Epson fiscal eposprint for Android sample project in order to connect mobile devices via TCP/IP to Epson intelligent fiscal printers. The sample project aids the developer in a basic understanding of the fundamentals and can be thought of a template starting point for development.

1.2 List of Related Documents

- Fiscal ePOS-Print Solution Development Guide
- Fiscal Printer Intelligent Features Guide
- Epson_FP_Manuale Operatore
- Protocollo di Comunicazione
- Intelligent Printer Printing System ePOS-Print Technical Data

1.3 Abbreviations

- FP: Fiscal Printer

1.4 Restrictions of Use

- The sample program is for evaluation purposes only and is not intended to be used as a finished application in retail outlets.
- No part of this document may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of Seiko Epson Corporation.
- The contents of this document are subject to change without notice. Please contact us for the latest information.
- While every precaution has taken in the preparation of this document, Seiko Epson Corporation assumes no responsibility for errors or omissions.
- Neither is any liability assumed for damages resulting from the use of the information contained herein.
- Neither Seiko Epson Corporation nor its affiliates shall be liable to the purchaser of this product or third parties for damages, losses, costs, or expenses incurred by the purchaser or third parties as a result of: accident, misuse, or abuse of this product or unauthorized modifications, repairs, or alterations to this product, or (excluding the U.S.) failure to strictly comply with Seiko Epson Corporation's operating and maintenance instructions.
- Seiko Epson Corporation shall not be liable against any damages or problems arising from the use of any options or any consumable products other than those designated as Original EPSON Products or EPSON Approved Products by Seiko Epson Corporation.

2 Restaurant Sample Program

The restaurant sample program can be used to print sample fiscal receipts (scontrini) and invoices (fatture). Simply select the items and watch the list grow and when ready choose either to print a fiscal receipt or an invoice. The intelligent fiscal printer responses can be checked under "STATO STAMPANTE:". The list is then cleared ready for the next sample. Invoice numbers are also incremented.

2.1 Restaurant Android Import

Item	Description
Tested environment	Eclipse
Required item	☐ Android SDK 2.3.3 (API 10) ☐ Java SE Development Kit 6 ☐ ADT(Android Development Tools) Plugin for Eclipse
Sample program file name	epsonAndroidFpDemo-1.1.zip
Source editing	Not required

2.2 Environment Setting Procedure

1. Obtain Android SDK. For example:
["http://dl.google.com/android/android-sdk_r22.0.1-linux.tgz"](http://dl.google.com/android/android-sdk_r22.0.1-linux.tgz)
2. Expand the archive to any folder.
3. Download and install the Java SE Development Kit 6.
4. Install ADT (Android Development Tools) plugin in Eclipse:
 - a. Start Eclipse.
 - b. From the menu items, select "Help" → "Install New Software".
 - c. Click the "Add" button.
 - d. Choose a name and enter the following location:
 - <https://dl-ssl.google.com/android/eclipse/>
 - e. Click OK.
 - f. Return to the "Install" screen and click "Developer Tools" → "Next".
 - g. When the installation process finishes, restart Eclipse.
5. Start Android SDK Manager and install the Android 2.3.3 (API 10) package. To save time, deselect all the other API versions.
 - a. Start Eclipse.
 - b. Select "Open Perspective" → "Java".
 - c. Select "Windows" → "Android SDK Manager".

6. Start Android AVD Manager and create an Android Virtual Device to suit your needs:
 - a. Start Eclipse.
 - b. Select "Open Perspective" → "Java".
 - c. Select "Windows" → "Android Virtual Device Manager".
 - d. Select the "New" button.
 - e. On "Create new Android Virtual Device", type the AVD name, choose a device and set the target drop down choice to "Android 2.3.3 – API Level 10".
7. Change the settings of Eclipse.
 - a. Start Eclipse.
 - b. Select "Window" → "Preferences".
 - c. Select "Android".
 - d. Click the "Browse" button.
 - e. Select the Android SDK directory chosen in step 2 above and click the OK button.
Although a warning dialog box appears, telling you that a path to Android SDK has not been set, there is no problem.
8. Configure your computer and TM intelligent printer so that they can connect to each other over the network.
 - a. Uncompress the sample program into any folder.
 - b. Import the sample program using Eclipse:
"File" → "Import" → "Android" → "Existing Android Code Into Workspace" → "Next" and then browse to the project directory.
 - c. Click "Finish".
 - d. Select "Run" → "Run Configurations" → "Android Application".
 - e. Under "New_configuration" choose a name and click browse (the imported project should be visible).
 - f. Close "Save".
 - g. If necessary force the target association of this project to the virtual device previously created.

Once compiled, the sample runs out of the box on emulated or real devices. You can select the printer IP address from a configuration menu ("MENU" button).