阿

一、需求

1.显示的每一行的内容不固定。

2.可设置字体的颜色、大小、左右上下间距、布局位置gravity等属性。

二、设计

1.打印的格式json对应的实体对象的模型建立。

路径：C:\Code\old\MobilePOS\app\src\main\java\com\witpos\mobile\entity\print\format

①文字的打印格式(TextPrintFormat)

②分割线格式（DividerPrintFormat）

③打印区域格式（AreaPrintFormat）

④整体的打印格式（PrintFormat）

路径：C:\Code\old\MobilePOS\app\src\main\java\com\witpos\mobile\config\print

①打印区域类型的代号（PrintAreaCode）

②打印文本类型的代号（PrintTextCode）

③打印的分割线类型的代号（PrintDividerCode）

④打印排列方式（PrintGravity）

2.老后台表格设计与评审（打印小票的格式表）

WITPOSB\_Print\_Ticket\_Format

3.终端打印的数据实体对象模型的建立

路径：C:\Code\old\MobilePOS\app\src\main\java\com\witpos\mobile\entity\print\info

①打印的商品数据的实体对象(ProductPrintEntity)

②打印的支付方式名称和金额的实体类（PayInfoPrintEntity）

③打印数据的实体对象（PrintInfoEntity）

4.打印机指令集的封装和测试。

路径：C:\Code\old\MobilePOS\app\src\main\java\com\witpos\mobile\util\print

热敏打印机通用打印指令集(PrintCommand)

5.建立 打印格式+打印指令+打印数据 的打印主体字符串的生成工具类,建立模拟测试数据，测试和完善。

路径：C:\Code\old\MobilePOS\app\src\main\java\com\witpos\mobile\util\print

生成打印主题的工具类（GeneratePrintBodyHelper）

1. /\*\*
2. \* 生成打印的主体
3. \* @param paperWidth 打印纸张的宽度（目前为58mm和80mm两种)
4. \* @param printFormat 打印的格式
5. \* @param printInfoEntity 打印的数据实体
6. \* @return 打印的主体字符串
7. \*/
8. public String generatePrintBody(int paperWidth, PrintFormat printFormat, PrintInfoEntity printInfoEntity)

6.建立连接调用不同类型打印机的工具类

路径：C:\Code\old\MobilePOS\app\src\main\java\com\witpos\mobile\util\print

调用不同类型打印机打印的通用工具类（PrinterHelper）

7.建立生成格式json小程序，根据需求图和打印格式模型对象，填入数据，获取到具体的格式json。

8.根据品牌代号、打印的纸张类型、打印的业务类型，往 打印小票的格式表WITPOSB\_Print\_Ticket\_Format 插入数据。

9.写获取打印格式json的接口和存储过程。

接口：GetPrintPatternJson

存储过程：get\_print\_pattern\_json

10.终端调用后台GetPrintPatternJson接口，下载后台配置的打印格式数据到终端表WPM\_Print\_Ticket\_Format

根据配置项：PARAMETER\_PRINT\_TICKET\_TYPE 若配置为2（使用后台设置模板的方式打印小票）则在开机时判断表的版本号下载后台表数据到终端；并且在下载数据中根据配置项来确定是否显示下载该表数据。

11.

在终端打印业务触发处，把打印的数据填入打印数据的实体对象（PrintInfoEntity）;

传入纸张宽度paperWidth、打印格式PrintFormat、数据实体对象PrintInfoEntity，调用生成打印主题的工具类（GeneratePrintBodyHelper）生成打印主体;

并调用打印机打印的通用工具类（PrinterHelper）的连接和打印方法，打印小票！