User Interface Guide

Version 3

Contents

Introduction

Although mobile transacting technology has existed for several years, transacting at point of sale using only your mobile device is still a relatively new concept for many consumers and cashiers alike. Furthermore, as more applications enter the market, consumers are being asked to transact in many different ways. Similarly, cashiers are being asked to understand many different payment processes, and as well as many different types of transactions (from pay to deposit to loyalty accumulation).

These are some of the reasons why the wiCode platform is so popular amongst merchants, from tier 1 retailers to independent coffee shops. The wiCode platform allows many mobile applications and many merchants across various industries to transact with each other using a mobile transaction standard. Through the platform, it doesn't matter which app is attempting which type of transaction at which retailer: the process is standardised.

The standardisation of mobile transactions has significantly improved customer experience and transaction speed, while essentially removing the need for constant cashier training every time a new payment app launches.

In order to both safeguard this standard, and to ensure that merchants continue to leverage and reinforce consumers' growing awareness of how to transact using a mobile phone, it is important that retail software integrating to the wiCode platform considers these standards.

The purpose of this document is to provide a user interface guide to point of sale providers through detailing typical implementation and the most essential norms with which consumers have become accustomed to. The recommendations contained within this document are made with the full understanding that different point of sale solutions do not look or feel the same and do not always behave the same. This document is not meant to dictate POS design, but rather to compliment and inform the design process.

Please do not hesitate to get in touch with us at support.sa@yoyogroup.com to discuss your point of sale design with one of our experienced integration specialists.

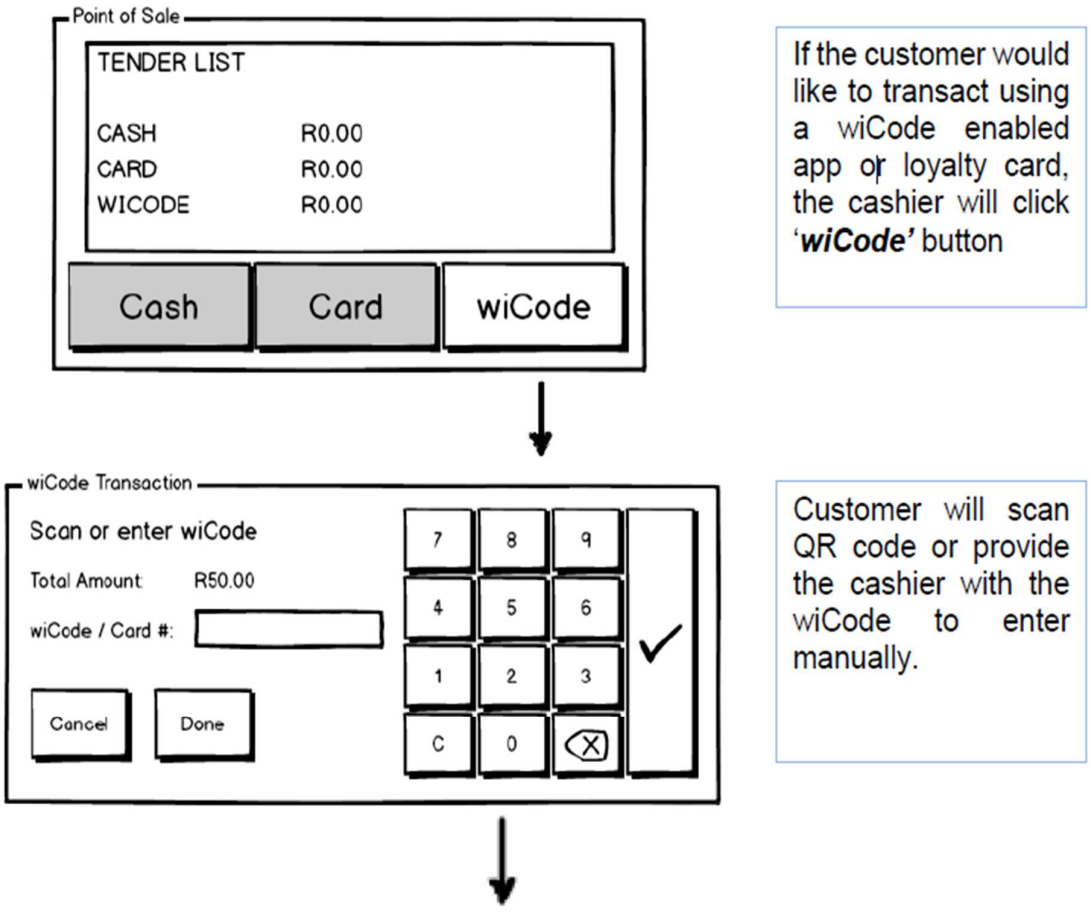
yoyo

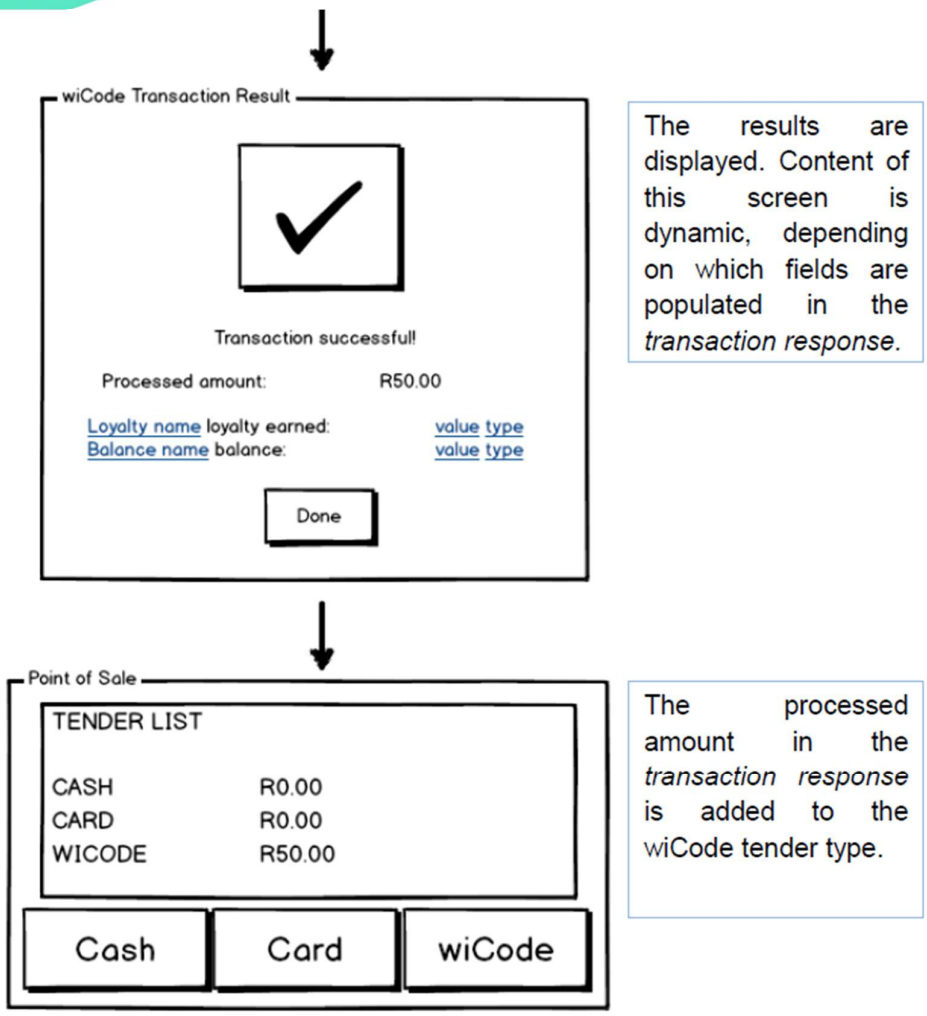
“Over-the-counter” Transactions

Retail and hospitality merchants who accept transactions from the customer at the till point will typically implement an ‘over-the-counter’ method of accepting wiCodes. This method involves scanning a customer’s QR wiCode or manually entering a wiCode provided by the customer. The point of sale then sends the wiCode along with the transaction to the wiCode platform for authorisation.

The wiCode button can be placed in sales mode or in tender mode as another tender option. Adding wiCode to the point of sale as an additional tender type is most correct.

The following are generic mock ups of the wiCode transaction flow for ‘over-the-counter’ transactions.





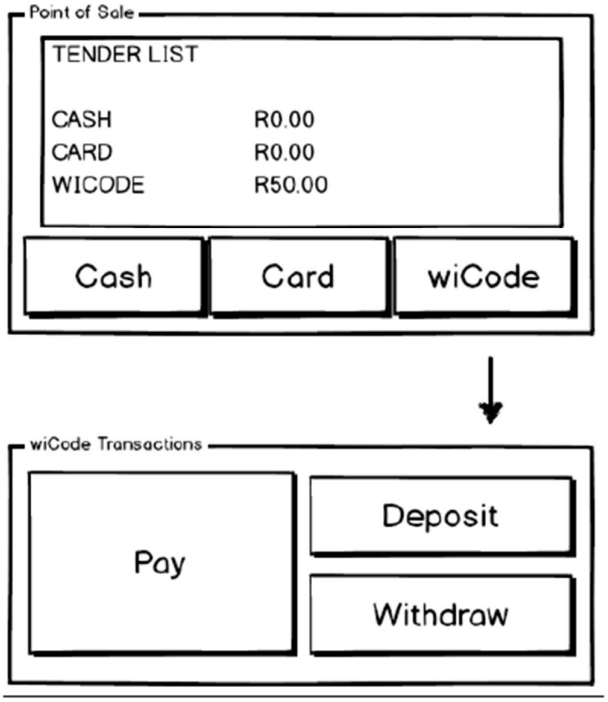
Notes:
- Fields displayed by POS on the result screen should be dynamic, based on the response from the server. A list of loyalty values earned or balances may be returned dependent on what the VSP supports.

- For PAYMENT type transactions, totalAmountProcessed should be subtracted from the outstanding amount and added to the basket as a line item with description “wiCode transaction”.

yoyo

Notes on Deposit and Withdrawal

Should deposit and withdrawal transaction types be included in scope, the recommendation for UI implementation is below:



Once a transaction type is selected, the user flow is identical, however

- The transaction type in the *transaction request* will reflect the user selection
- The transaction response information should be displayed correctly:
  - For type DEPOSIT, totalAmountProcessed should be added to the outstanding amount and included in the basket as a line item with description “wiCode deposit”.
  - For type WITHDRAWAL, totalAmountProcessed should be subtracted from outstanding amount, with description “wiCode withdrawal”.

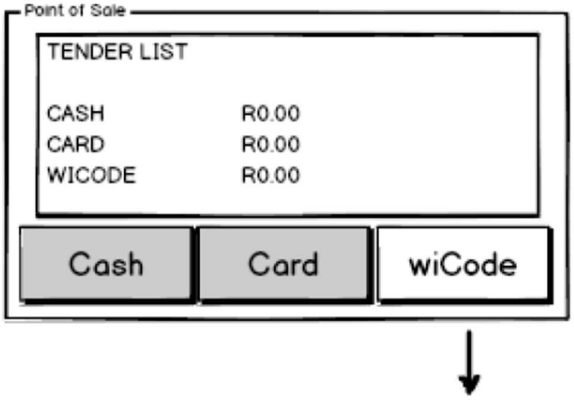
yoyo

“Sit-down” Transactions

Merchants who do not usually accept payments at the till point will typically implement the ‘sit-down’ transaction method, which allows the customer to pay by scanning a merchant QR printed on the bill. The point of sale will then query transactions made against that bill. Note: the over-the-counter method will still be implemented, accessible through an ‘*enter/scan wiCode’* button.

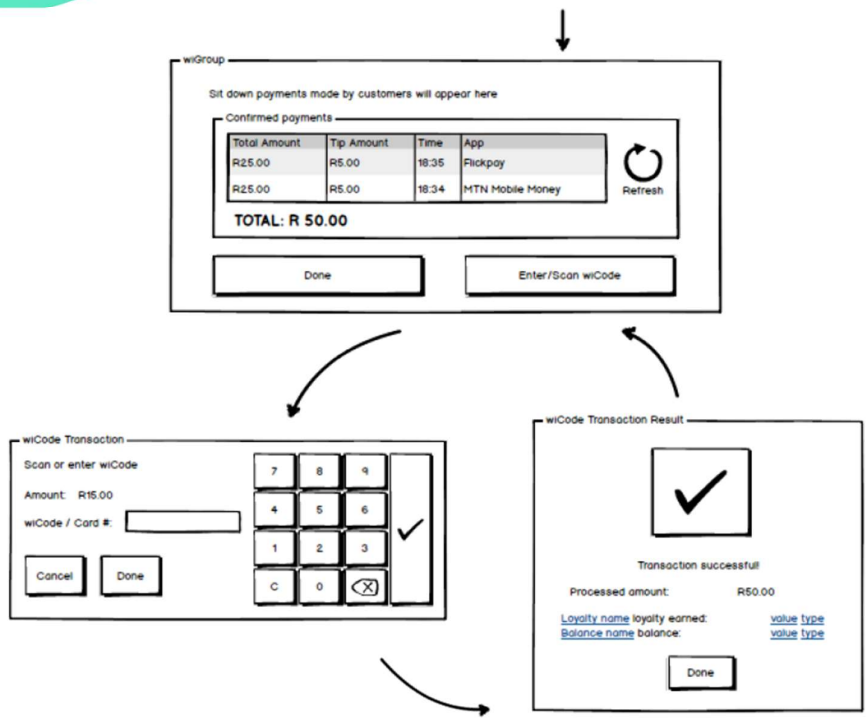
The wiCode button can be placed in sales mode or in tender mode as another tender option. Adding wiCode to the point of sale as an additional tender type is most correct.

The following are generic mock ups of the wiCode transaction flow for ‘sit-down’ transactions.



If the customer would like to transact using a wiCode enabled app or loyalty card, the cashier will click ‘***wiCode’*** button

Display the value, app and time of the transactions returned by the transaction history call. Allow for user to refresh in order to pick up new payments. When selecting ‘done’, return sum of *totalAmountProcessed* of all returned *MainTransaction* objects to tender screen. If Enter/Scan wiCode is selected, navigate to wiCode scan screen.



Follow the UI flows detailed in ‘over-the-counter’ transaction flow. Return the relevant fields to the transaction table.

Notes:
- The total amount of the payments may exceed the total amount due, as a tip may be included in the amount paid by the customer(s).

- There is no limit to the number of payments that can be made against a bill.

- Sitdown payments cannot support product discount coupon redemptions and loyalty if the over-the-counter button is included in the implementation.

Till Slip Enhancements

There are two components which should be considered:

- wiCode QR printed on till slip for ‘sit-down’ transactions
- display of successful wiCode transactions on the invoice

wiCode QR on till slip

The wiCode QR should be generated and printed on the till slip below the list of products which are included on the bill, similar to the below:

The highlighted text above should be configurable, as this is a useful place to educate customers about which apps are accepted by the merchant.

The **wiCode QR** should be a representation of the following information separated by new lines:

Information	Details
QR TYPE	Can be STORE or BILL, usually will be BILL if printed on the bill.
BasketID_MMddyyyHHmmss	BasketID is the unique identifier generated by the point of sale for the bill.
StoreID	Provided by Yoyo, unique per store / merchant.
Bill Value	In cents.

Example for merchant registered on the wiCode platform with storeID of 2000.

BILL INV201_01212015123510 2000 5900	

wiCode transactions on invoice

The invoice should display the following text and fields for each wiCode transaction successfully performed:

wiCode [transactionType]: [processedAmount]

Transaction ID: [*wiTrxId*]

Example:

wiCode Payment: R59.00

Transaction ID: 3948863

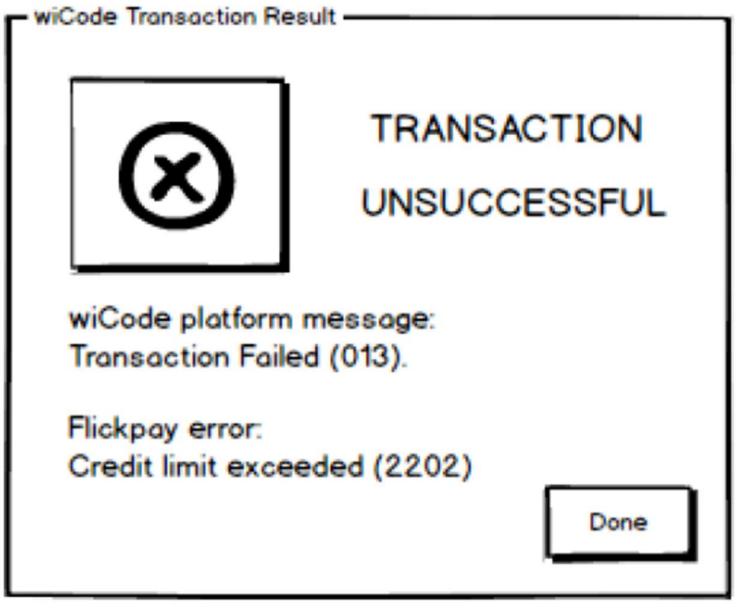
Error Handling

It is important to display the correct information to the POS user when an error occurs, as this will reduce support calls and increase the cashier's ability to provide meaningful feedback to the customer.

When a transaction is failed by the wiCode platform, the *responseCode* and *responseDesc* contains relevant information.

When a transaction is failed by the VSP, the *responseCode* and *responseDesc* contained in the VSP object in the transaction response also contains relevant information.

The recommendation is to thus display the following information in a pop up box:



In the above example, the text above is populated using the following fields:

Text	Field
wiCode platform message:	"wiCode platform message:"
Transaction failed (013)	[responseDesc] ([responseCode])
Flickpay error	[vsp object name] "error"
Credit limit exceeded (2202)	[vsp object responseDesc] (vsp object responseCode])