

# Certification Criteria - Pay@table

Tyro's Integrated Pay@Table is an integration mode that allows Tyro's mobile terminals to communicate with your integration product via the implementation of a Pay@table server, and allows you to take card payments directly at the table. The Tyro EFTPOS terminal is brought directly to the diners so they can make payments, split-bills, and tips at their table.

The main benefits of the Pay@table integration mode are:

- The point-of-sale is not locked down while bills are being processed, this is due to the '**Pull**' nature of the integration, as opposed to the POS system pushing the sales onto the terminals it is the Tyro terminal pulling the sales off of the Pay@table server and the Point-of-sale till can remain open as a result of this.
- In-built capability to split bills which is perfectly tailored to the hospitality industry, customers can split bills if required and pay at their table which is an all-round more convenient experience for the customers.
- A more discrete tipping workflow that is a better and more comfortable user-experience for the user as compared to the tipping workflow for standalone and standard integration.
- Pay@Table allows for more of a personal and hands-on experience for the customer/s and employees.
- Pay@Table also follows a very straightforward and uncomplicated workflow which is easy-to-understand.

Given below is a summary of the feature-set and the feature-options along with the requirement level associated with them:

Features and Feature-options	Requirement level (Mandatory/Optional)
Passphrase Configuration	<b>Mandatory</b>
Integrated Purchases	<b>Mandatory</b>
• Feature option: Pay@table Integration Modes	<b>Optional</b>
• Feature option: In-POS Split Payments	<b>Optional</b>
Integrated Surcharging	<b>Mandatory</b>
Integrated Tipping	<b>Mandatory</b>
Table locking	<b>Optional</b>
HMAC verification	<b>Mandatory</b>
HTTPs basic authentication (Cloud-based Pay@table integrations only)	<b>Mandatory</b>

## Feature Criteria Descriptions and Deliverables

Each of the developed Tyro Pay@table integration features from the feature-set must meet a defined criteria in terms of functionality, workflow, and the outputs/end result delivered to stand eligible for certification. This section defines each feature and any related feature-options in terms of their functionality, and gives specific examples of the deliverables that the POS must provide, as well as the ideal end-result that is to be achieved.

### Passphrase Configuration

Pay@table integrations, both cloud-based and LAN-based, rely upon a terminal passphrase to authenticate the communication between the terminal and the POS server, the passphrase is generated by the POS and is set in both the Tyro merchant configuration and the POS configuration.

The criteria for Passphrase generation and configuration is as follows:

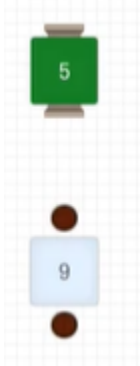
1. The POS must allow generation and configuration of the passphrase through suitable constructs within the POS user interface, only administrator-level users must have the ability to generate passphrases from within the POS.
2. The POS must not set a hard-coded passphrase for all merchants.
3. Passphrases must be globally unique i.e. unique for each location and a passphrase generation strategy that allows for this must be chosen.
4. The passphrase must also be a minimum of 6 characters in length, and contain a suitable mix of alpha-numeric characters.

### Integrated Purchase

Integrated Purchases for the Pay@table integration mode are a mandatory feature that each integration product must provide to the merchants. The integrated purchase feature entails the ability to be able to create and store open sales on the Pay@table server through the POS, pull the created open sales through the Tyro terminal, and process the payment fully, or partially whilst handling the response from the terminal and subsequently the open sale in the required manner so that the outstanding amount is updated on the Pay@table server and reflected correctly on the POS interface as well in real-time.

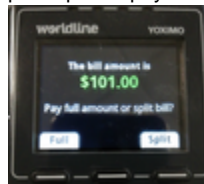
In terms of workflow and deliverables, the Integrated Purchase feature criteria is:

1. The user must have the ability to create open sales on the table map through the POS interface, this involves the user adding the required sale items to the table and the POS storing all the relevant open sale details on its database correctly. These details include:
  - a. Table number (can be omitted to accommodate the [Pay@table Restaurant and Bar Mode](#))
  - b. Operator ID
  - c. Sale Amount
  - d. POS reference
  - e. Display text pertaining to the open sale
2. Please note that the POS must not allow a null amount or 0.0 \$ to be stored on an open sale.
3. Once the open sale is created, the table must be marked as unavailable on the POS interface as is shown below, with the table number clearly displayed as well, an example is shown below:



Here, the green table signifies it being unavailable as there is an open sale on it at present, the blue icon is used to indicate tables that are available for open sales to be created on. It must be made clear as to how each table status is being represented i.e. via an on-screen legend.

4. Once the open sale has been created on the table, the table must remain securely stored in the POS database and continue to remain marked in the appropriate status on the interface (as per point 2) this means that closing down or logging off and back on on the POS must not result in the open sales being removed or erased.
5. The user must be able to pull off an existing open sale on the Tyro terminal by entering the Operator ID and/or Table number, once this is done the terminal will send the GET open sales request to the POS server, the POS server and the POS must return the response to the terminal. The response **must** have the content-type set to text/plain (all other content-types are rejected) and a character encoding of ISO-8859-1 (LATIN1).
6. The user can make the full payment or choose to make a split or partial payment using the options presented on the terminal:



If the 'Split' option is selected the user will then have the option to pay off a part of the sale amount and close the sale on the terminal, the P@T server and the POS must ensure that the correct outstanding unpaid amount must be reflected on the POS interface at all times without delay e.g. a 100.0 \$ table is pulled on the terminal and split is selected, then a 50.0 \$ payment is made, the terminal returns the approved transaction response JSON block to the POS server as a POST request

7. The POS must use this to:
  - a. Update the outstanding amount on the open sale.
  - b. Create the relevant transaction invoice/s within the POS.
8. The POS must not return unique POS references in the `pos-reference` field in the response for payments split on the terminal.
9. The user can press the '**STOP**' button on the Tyro terminal to return the control back to the POS, in these cases the open sale must remain open with the correct outstanding amount on it, as per point 2 until the payment has been made in full.
10. Upon the payment being completed, the sales/transactions must be stored correctly on the POS, the table must be marked as closed, and made available for an open sale to be stored on it.
11. The Tyro terminal is allowed 10 seconds as a time-out period to send the approved transaction response back to the POS server, the implementation must take this into account, so as to keep the latency between the sale being completed on the terminal and it being received by the POS server to a minimum.

### Pay@table Integration Modes

Pay@table integration entails three modes of operation: the Restaurant mode, the Restaurant and Bar mode, the Retail mode. These modes are designed with specific workflows tailored to suit the needs of the particular business type as is explained below:

## Pay@table Restaurant Mode

Pay@table Restaurant mode entails the terminal workflow asking the staff member to enter in their operator ID and table number. The terminal passes these values to the POS system and the POS system returns a list of (up to 100) open sales. Typically the POS server will return a single open sale however by allowing the POS system to return a list of open sales, the POS system can then optionally return a list of open sales by seat or a list of sub tabs. If more than one open sale is returned, the terminal will present the open sales in a list allowing the staff member to select the correct one. In restaurant mode, the terminal will offer the choice of paying the full amount or splitting the bill. If enabled the terminal will also offer tipping.

## Pay@table Restaurant and Bar Mode

In restaurant and bar mode, the Tyro terminal offers a combination of Restaurant and Retail modes. On the '**Enter table number**' screen, the staff member can press the '**BAR**' button to skip the table entry section and request the POS system to return the open sales based only on their operator ID. This mode is useful in the situation where Pay@Table is used in both Pay@table or the more conventional pay-at-the-counter scenarios. Please note that if '**BAR**' is pressed and the selected open sale contains a table parameter, then the terminal workflow will revert back to the restaurant workflow so that split-billing can be offered.

## Pay@table Retail Mode

In retail mode, the terminal workflow asks the staff member to enter in their operator ID. The terminal passes this value to the POS system and the POS system returns a list of (up to 100) open sales. If more than one open sale is returned, the terminal will present the open sales in a list allowing the staff member to select the correct one. If enabled the terminal will also offer tipping.

The criteria that the POS server is required to comply to be compatible with Pay@table Restaurant mode, Restaurant and bar mode, and Retail mode is:

1. The POS server and the POS program must be able to handle both cases i.e. where the open sales GET request contains:
  - a. The operator ID and table number both are specified.
  - b. Only the operator ID is specified.
2. The POS server must not have any kind of limitation on the number of open sales that can be sent back at any given moment, and must be able to support the 100 open sales limit that the Pay@table protocol can accommodate.

## In-POS Split-payments

The POS may additionally also implement the option allowing the POS user to split a single table sale into multiple open sales associated to the same table number, these open sales can be individually paid off and would allow the user the ability to split items between diners and pay them individually.

For example, Table 1 has two items on it i.e. Item A costing \$ 20.0 and Item B costing \$ 30.0 with a total of \$ 50.0.

The diners indicate that they want the sale split by item, the Table 1 sale can be split into two with Item A on one and Item B on the other split sale.

These sales can then be pulled off for table 1 and will be seen on the terminal as something like this:



The criteria for this feature-option is as follows:

1. The POS must provide the user an interface through which the user can specify the items and create the split sale.
2. The POS must not create an open sale with a different table number, but rather create another open sale with the same table number as the original sale.
3. The POS must return a suitable description, this description must be returned in the `display-text` field (maximum 6 characters)
4. The POS should return distinct POS references for the split payments in the `pos-reference` field.

## Integrated Surcharging

Integrated Surcharging is a compulsory feature for the retail industry, which allows Tyro's dynamic surcharging to be applied to any given purchase transaction and reflected on the POS User Interface and the POS sales receipt. A prerequisite is that the surcharge rates specified on the Online **Tyro Merchant Portal** and surcharging be enabled on the terminal.

The workflow should ideally be as follows:

1. An open sale being held on a table is retrieved on a terminal, the cardholder taps the card and the Tyro terminal dynamically applies surcharge to the transaction (Only cards of scheme EFTPOS are non-surchageable).
2. The transaction is processed and approved on the Tyro terminal.
3. The Tyro terminal sends the approved transaction response back to the POS server on the transaction result URL as a JSON object, and the surcharge amount is returned as a dollar amount in the JSON object in the `surcharge-amount` field.
4. The POS must use the surcharge amount from the `surcharge-amount` field, this surcharge amount must be included on the POS sales receipt as well, as can be seen below the surcharge amount is clearly marked and included on the POS receipt.

```
TAX INVOICE
Merchant 165650
Address line 1
Address line 2
PERFECT POS
Items          $
Coffee         $4.00
Croissant      $6.00
Sub-total     $10.00
Surcharge      $1.00
Tip            $2.00
Total         $13.00
GST inc       $1.00
POS ID: 2
Ref: 12345abc
04 Jan 2019 at 02:12 PM
-----
CUSTOMER COPY
Tyro HealthPoint
125 York Street
Sydney NSW 2000
Tyro Payments EFTPOS
Card: xxxxxxxxxxxx0010(c)
VISA CREDIT
AID: A0000000033008
Purchase      AUD $10.00
Surcharge     AUD $1.00
Tip           AUD $2.00
Total         AUD $13.00
Terminal ID: 4
Transaction Ref: 047206
Authorisation No: 014458
08 Apr 2020 at 09:55 AM
```

5. This surcharge amount must be stored in the POS in the sales journal/sales invoice for the respective approved transaction.
6. For transactions accruing surcharge that are being refunded, the POS must recall the surcharge amount from the stored sales invoice and includes it in the full amount to be refunded automatically without any manual interference (if the refund is being made through standard EFTPOS integration using Tyro).
7. Additionally, the surcharge amounts per transaction and total amount per day must be reflected accurately on any POS reconciliation reports that have been developed.

## Integrated Tipping

Integrated tipping is a mandatory feature for the hospitality industry, which allows tips applied to any given purchase transaction to be registered and reflected end-to-end from the Tyro reporting to the POS User Interface, and the POS cashdrawer.

The certification criteria for this feature is as follows:

1. An open sale is retrieved on the Tyro terminal, the cardholder enters the tip amount on the terminal and presents their card on the terminal, the transaction is processed and approved on the Tyro terminal.
2. The Tyro terminal sends the approved transaction response back to the POS server on the transaction result URL as a JSON object. The tip amount is returned as a dollar amount in the JSON object in the `tip-amount` field.
3. The POS must use the tip amount from the `tip-amount` field and include it on the POS sales invoice as well, as can be seen below the tip amount is clearly marked and included on the POS receipt.

TAX INVOICE

Merchant 1655650  
Address line 1  
Address line 2

PERFECT POS

Items	\$
Coffee	\$4.00
Croissant	\$6.00
Sub-total	\$10.00
Surcharge	\$1.00
Tip	\$2.00
Total	\$13.00
GST inc	\$1.00

POS ID: 2  
Ref: 12345abc  
04 Jan 2019 at 02:12 PM

-----  
CUSTOMER COPY

Tyro HealthPoint  
125 York Street  
Sydney NSW 2000

Tyro Payments EFTPOS

Card: xxxxxxxxxxxx0010(c)  
VISA CREDIT  
AID: A0000000031010

Purchase	AUD	\$10.00
Surcharge	AUD	\$1.00
Tip	AUD	\$2.00
Total	AUD	\$13.00

Terminal ID: 4  
Transaction Ref: 547286  
Authorisation No: 014458  
08 Apr 2020 at 09:55 AM

4. This tip amount must be stored in the POS in the sales journal/sales invoice for the respective approved transaction.
5. For refunds of transactions where a tip was applied, the POS must recall the tip amount from the stored sales invoice and includes it in the full amount to be refunded automatically without any manual interference.
6. Additionally, the tip amounts per transaction and total tip amount per day must be reflected accurately on any POS tipping and reconciliation reports that have been developed.

### Table Locking

Table locking is a feature of Pay@table integration that allows:

- **Terminal-side locking:** A table that has been successfully retrieved from the POS and is currently open on the Tyro terminal to become locked or unavailable on the POS.
- **POS-side locking:** A table that is currently open on the POS to become locked or unavailable for the Tyro terminal.

The criteria for terminal-side locking is as follows:

1. An open sale is created on the POS.
2. The respective table number and/or operator ID is used to pull the open sale on the Tyro terminal.
3. The open sales GET request response sent back to the POS server must contain the `always-return-outcome` set to 'true' so that the terminal can participate in the table locking workflow.
4. The terminal displays the sale amount and prompts the cardholder to complete the sale.

5. The POS must mark the table that has been opened on the Tyro terminal as locked.
6. Attempts to access this table on the POS must be blocked and the POS must display a user-friendly message advising that the table is currently open on the Tyro terminal.
7. Once the open sale is released by the Tyro terminal it must be available to be accessed on the POS and the relevant status indicators for the open sale reflecting the same, once the sale has been released by the Tyro terminal the terminal will return the cancelled transaction response (as can be seen below) confirming that the table has been released by the POS, and can hence be made available on the POS.

```

{
  "mid": "850",
  "operator-id": "10",
  "result": "CANCELLED",
  "table": "0",
  "tid": "437"
}

```

The criteria for POS-side locking is as follows:

1. An open sale is created and saved successfully on the POS.
2. The open sale is re-opened on the POS, and the POS marks this table as locked.
3. The respective table number and/or operator ID is used to attempt to pull the open sale on the Tyro terminal.
4. The POS server checks if this terminal is locked and upon confirmation returns the HTTP code 412 and a suitable and user-friendly message that is included in body of the response e.g. ***"This table is currently open on the Point-of-Sale program. Please close the table and try again"***
5. Upon the sale being released on the POS the open sale is made available to be pulled off by the POS.

#### HMACH verification

HMACH verification is a mandatory feature of the Tyro Pay@table integration feature-set that is implemented to secure the POST requests that the terminal sends to the POS server.

The criteria for certification of HMACH verification for Pay@table integrations is as follows:

1. The Tyro terminal calculates the HMACH using the HMACH-sha1 algorithm and includes it in the HTTP request header `x-tyro-mach` for the POST requests that are sent through by the terminal.
2. The POS server then computes the HMACH using the message received and the secret i.e. the Pay@table passphrase.
3. The POS must use the message JSON block as is received and not parse, edit, modify, or omit any characters or sections from the message block since this will result in the HMACH verification failing.
4. If verification fails the POS server must return a HTTP status code of 403.
5. If verification passes the POS server returns a 200 HTTP status code.

#### HTTPs Basic authentication (Cloud-based Pay@table integrations only)

HTTPs basic authentication is a mandatory feature for Cloud-based Pay@table integrations. The cloud-based pay@table integrations entail HTTP requests being sent between the Tyro Cloud Pay@table server and the POS cloud server. The support for Cloud based POS integrations extends the support for LAN-based POS integrations by supporting HTTPs for the three URLs as well as including HTTP Basic Authentication. As an additional security measure, all communication between the terminal and the data centre hosting the POS is proxied via Tyro's data centres. This allows the POS to only accept Pay@Table requests from Tyro's IP ranges.

The criteria for this is given below:

1. Each of the three URLs (open sales, transaction result, and diagnostics) must start with the https prefix.
2. Only a single pair of username and password used in the HTTP Basic Authentication can be configured per POS partner i.e. individual credentials per merchant must not be used or specified.
3. The password used for HTTP Basic Authentication must be different to the one used for the message MACs which is the terminal secret /passphrase.
4. Any customer identifiers that you require to be included in the URLs must be in the host name or the resource section of the URL and not in the query string. i.e. customer identifiers must be to the left of the '?' in the http request.  
e.g. **`https://merchant-name.mypos.com/{company id}/{site ID}/open-sales?operatorId=2175&mid=1234&tid=5[&table=37]`**
5. A credential management strategy for HTTP Basic Authentication credentials must be agreed with Tyro during the certification process, this involves providing us details such as **HTTPs credentials change frequency** i.e. how frequently the credentials will be changed e.g. annually, biannually, or quarterly, **HTTPs credentials change communication method** i.e. how these credentials will be provided to us whenever they are updated, and any additional details that are relevant to the process.