

# Interoperabilidad

## TPV



## Índice de Cambios

<u>Versión</u>	<u>Fecha</u>	<u>Cambio(s) efectuado(s)</u>	<u>Pgs. Afectadas</u>
1.0	04/04/2017	Inicio	Todas
2.1	11/05/2017		Todas
2.2	09/06/2017	General Use Case	Todas
2.4	29/08/2017	Mas detalle y Domains	Todas
2.5	31/08/2017	Eliminación Transaption	Todas
2.6	05/09/2017	Add response field in Validate WS	Pag 12 y13

## INDEX

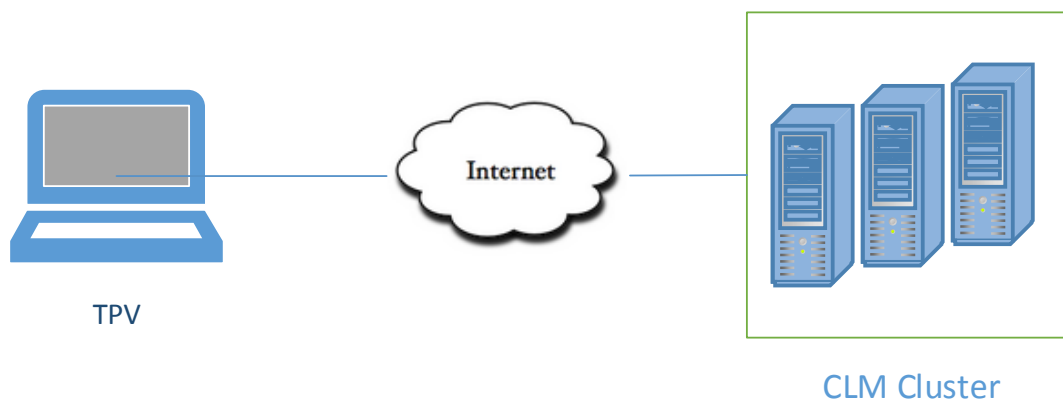
<b>INDEX</b> .....	<b>3</b>
<b>1. DESCRIPCIÓN DEL SERVICIO</b> .....	<b>4</b>
<b>2. ARQUITECTURA TÉCNICA</b> .....	<b>5</b>
2.1. VISIÓN GENERAL DE ARQUITECTURA. DIAGRAMA .....	5
2.2. ELEMENTOS DE SEGURIDAD .....	6
<b>3. CLM. RESTFUL API</b> .....	<b>6</b>
WS TPV/LOYALTY (POST) .....	7
WS TPV/RESUME (GET) .....	10
WS TPV/VALIDATE (GET) .....	12
GENERAL USE CASE .....	14

# 1. DESCRIPCIÓN DEL SERVICIO

---

CLM es una plataforma software que permite fidelizar a clientes ocasionales en clientes regulares a través de los hábitos consumos y la explotación de las diferentes variables que intervienen en la venta.

En este punto se hace necesario la integración del terminal de venta con la plataforma CLM de manera que permita al sistema recolectar los datos necesarios para su posterior análisis. La explotación de estos datos a través de la plataforma permite realizar campañas de marketing y establecer acciones de “**business intelligence**” directamente a los clientes.



El proceso de venta se define de la siguiente forma:

## **Operación Con descuento/ cupón oferta:**

1. Un cliente realiza un pedido
2. El cliente muestre su código de cliente o código de cupón
3. El empleado escanea el código integrado en la caja.
4. El TPV envía la información sobre el código leído a la plataforma CLM en donde se validará los datos. CLM envía la respuesta validando la operación e informando al TPV sobre la validez de la oferta/cupón o descuento.
5. La caja actualiza el importe total añadiendo la promoción o el descuento
6. El empleado termina la operación cobrando al cliente el importe final.

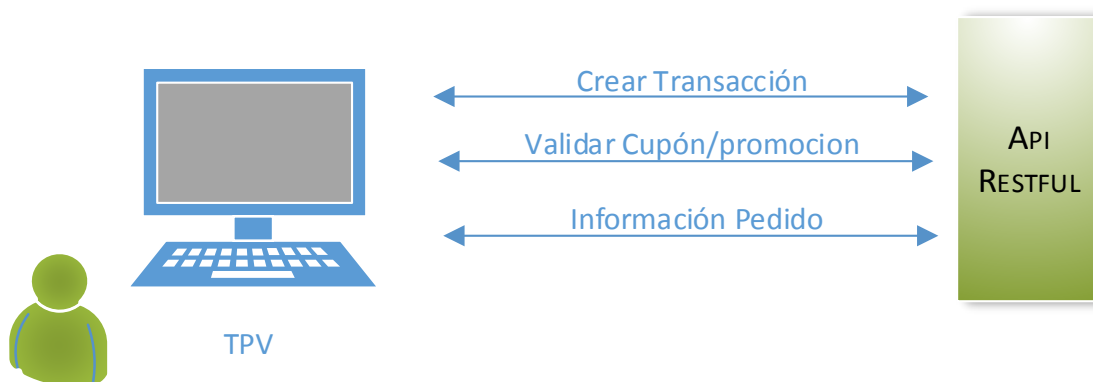
### **Operación Sin descuento/ cupón oferta:**

1. Un cliente realiza un pedido
2. El cliente no tiene la aplicación. El empleado le informa sobre las ventajas de la aplicación.
3. El empleado teclea el importe de la operación y el TPV envía la información a la plataforma CLM en donde se registrará los datos.
4. El empleado termina la operación cobrando al cliente el importe final.

## 2. ARQUITECTURA TÉCNICA

### 2.1. Visión General de Arquitectura. Diagrama

La realización del proceso se debe de establecer los mecanismos de integración en ambos sistemas.



El sistema contará con tres partes fundamentales:

- **Sistema TPV.** Dispositivo que permite centralizar y gestionar la venta comercial al público.
- **CLM Webservices Restful.** CLM dispondrá de una serie de Webservices en tecnología Restful con funcionalidades concretas necesarias para recolectar y aplicar descuentos, promociones y cupones.

## 2.2. Elementos de Seguridad

---

Para evitar el uso malintencionado de la aplicación por personas ajenas a Dunkin o al sistema se han dispuesto las siguientes medidas de seguridad:

- **Cifrado de las comunicaciones**

Los intercambios de información entre el TPV y CLM RestFul Api se establecerán mediante protocolo https. El sistema HTTPS utiliza un cifrado basado en **SSL/TLS** para crear un canal cifrado cuyo nivel de cifrado depende del servidor remoto. De este modo se consigue que la información sensible no pueda ser usada por un atacante que haya conseguido interceptar la transferencia de datos de la conexión, ya que lo único que obtendrá será un flujo de datos cifrados imposible de descifrar.

## 3. CLM. RESTFUL API

---

<i>Name</i>	<i>Tipo</i>	<i>Versión</i>	<i>Descripción</i>
Loyalty	POST	1.0	Este método permite a un usuario acumular puntos de fidelización.
Trasaction	POST	1.0	Este método permitirá enviar la transacción de cliente.
Resume	GET	1.0	Este método permite verificar cupones y promociones de un cliente a través del código de barras.
Validate	GET	1.0	Permite Canjear un cupón/promoción a través del código de barras.

## WS TPV/LOYALTY (POST)

### Description

Este método permite a un usuario acumular puntos de fidelización.

input		
Type	Field	Description
<a href="#">TpvTransVo</a>	<a href="#">*tpvTransVo</a>	

(\*) Mandatory fields


output		
Type	Field	Description
		<a href="#">http Code 200 - Success</a>
		<a href="#">Error -Code</a>

(\*) Mandatory fields

URL path: <https://srv.dunkincoffee.es/dunkin/api/v1.0/tpv/loyalty>

Error Code	Type	Description
<a href="#">500</a>	<a href="#">INTERNAL_SERVER_ERROR</a>	<a href="#">GENERAL</a>
<a href="#">400</a>	<a href="#">BAD_REQUEST</a>	<a href="#">IllegalArgumentException</a>
<a href="#">404</a>	<a href="#">NOT_FOUND</a>	<a href="#">Client Not Found Exception</a>
<a href="#">406</a>	<a href="#">NOT_ACCEPTABLE</a>	<a href="#">PVP less than Avg Check</a>

TpvTransVo		
Type	Field	Description
<a href="#">String</a>	<a href="#">code</a>	<a href="#">Client's Code</a>
<a href="#">String</a>	<a href="#">ticket</a>	<a href="#">Ticket number</a>
<a href="#">String</a>	<a href="#">tpv</a>	<a href="#">TPV identification</a>
<a href="#">String</a>	<a href="#">shop</a>	<a href="#">Shop Identification</a>
<a href="#">Double</a>	<a href="#">pvp</a>	<a href="#">Total Amount</a>
<a href="#">Double</a>	<a href="#">pvpPromo</a>	
<a href="#">Long</a>	<a href="#">fxOper</a>	<a href="#">current time in milliseconds</a>

Request Example	
POST	<code>http://80.32.19.228:28080/dunkin/api/v1.0/tpv/loyalty</code>
<pre>{   "ticket": "12564895154",   "code": "xssxzcsdfs",   "tpv": "4",   "shop": "4",   "pvp": "15.20",   "fxOper": "1494512639570" }</pre>	
	
Response	
200	

USES CASE
<ol style="list-style-type: none"> <li>1. El usuario realiza un pedido</li> <li>2. El empleado prepara el pedido e incluye en el ticket todos los elementos del pedido.</li> <li>3. El usuario utiliza sus promociones y cupones. <b>Ver WS TPV/VALIDATE.</b></li> </ol>




- Una vez aplicada las promociones y cupones que usuario disponga en la app el usuario puede mostrar su código de cliente, para obtener puntos de fidelización.



En este caso se activará el lector de códigos y el cliente procederá a escanear este código. Una vez escaneado el código el TPV enviará la información de la transacción invocando a este servicio (WS TPV/LOYALTY), en donde se enviará la información de la transacción como en el ejemplo anterior.

Nota: El campo "**pvpPromo**" es el precio que pagará el cliente teniendo en cuenta todas las promociones y descuentos aplicados. Si el cliente no tiene descuento aplicado se enviara por defecto 0 o no se enviará este campo. El campo "**pvp**" es el precio que pagaría el cliente sin aplicar ningún descuento.

- El cliente realizará el pago finalizando la transacción.

## WS TPV/RESUME (GET)

### Description

Este método permite verificar cupones y promociones de un cliente a través del código de barras y saber en caja si la promoción o el cupón es válido o no.

Este servicio no realiza ningún acción sobre el cupón o la promoción y solo permite saber si es válido o no.

Este servicio existe en la actualidad ya que el empleado verifica la validez de la promoción o cupón para después realizar la acción de canje.



input		
Type	Field	Description
String	barcode	PathParam. Promo code
String	tpv	QueryParam. TPV Identification
String	shop	QueryParam. SHOP Identification

(\*) Mandatory fields

output		
Type	Field	Description
String	status	
String	estado	

(\*) Mandatory fields

### URL path:

<https://srv.dunkincoffee.es/dunkin/api/v1.0/tpv/{barcode}/resume?tpv=1&shop=44>

Error Code	Type	Description
500	INTERNAL_SERVER_ERROR	GENERAL
400	BAD_REQUEST	IllegalArgumentException
401	UNAUTHORIZED	uuidDevice not found
404	NOT_FOUND	Coupon not found

Request	
GET	<a href="http://80.32.19.228:28080/dunkin/api/v1.0/tpv/XJ23ETS1/resume?tpv=1&amp;shop=44">http://80.32.19.228:28080/dunkin/api/v1.0/tpv/XJ23ETS1/resume?tpv=1&amp;shop=44</a>

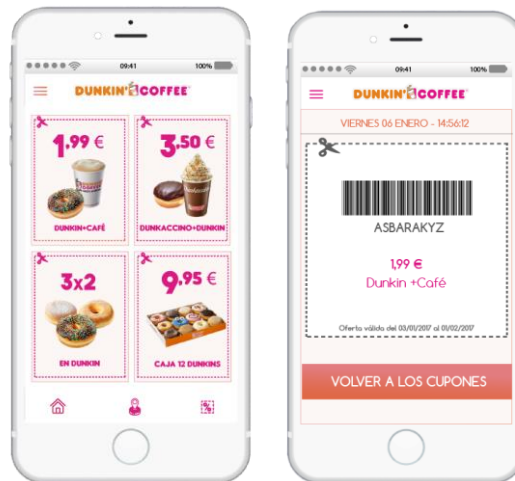
Response
{"status":"Este codigo de barras no ha sido utilizado","estado":1}"

```
"{"status":"Este codigo de barras ya ha sido utilizado","estado":0}"
```

```
"{"status":"No existe codigo de barra con numero +barCode+", "estado":1}"
```

## USES CASE

1. El usuario realiza un pedido
2. El empleado prepara el pedido e incluye en el ticket todos los elementos del pedido.
3. El usuario utiliza sus promociones y cupones. El empleado activa el escáner:



4. El cliente escanea el código de la promoción, enviando el código a CLM a través de este servicio (**Ws TPV/Resume**)
5. CLM responderá indicando la validez de la promoción.

## WS TPV/VALIDATE (GET)

### Description

Permite Canjear un cupón/promoción a través del código de barras.



Input Type	Field	Description
String	barcode	PathParam
String	tpv	QueryParam. TPV Identification
String	shop	QueryParam. SHOP Identification

(\*) Mandatory fields

output Type	Field	Description
String	status	Message
String	estado	0 Error; 1 OK
String	code	PLU

(\*) Mandatory fields

### URL path:

<https://srv.dunkincoffee.es/dunkin/api/v1.0/tpv/{barcode}/validate?tpv=1&shop=44>

Error Code	Type	Description
500	INTERNAL_SERVER_ERROR	GENERAL
400	BAD_REQUEST	IllegalArgumentException
401	UNAUTHORIZED	uuidDevice not found
404	NOT_FOUND	Client or Inbox not found

Request	
GET	<code>http://80.32.19.228:28080/dunkin/api/v1.0/tpv/XJ23ETS1/validate?tpv=1&amp;shop=44</code>

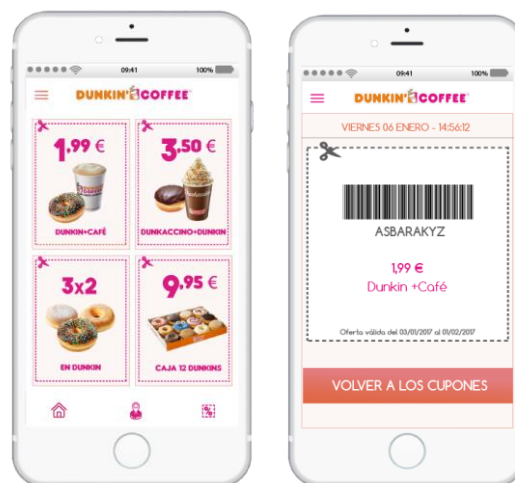
Response
<pre> {"status":"Codigo de barras validado exitosamente","estado":1,"code":"210014"} {"status":"No existe codigo de barra con numero "+barCode+"","estado":1} {"status":"Cliente No Encontrado","estado":1} {"status":"Parametros incorrectos. Vuelva a escanear de nuevo el codigo","estado":1} {"status":"Se ha producido un error. Consulte con el administrador","estado":1} {"status":"Promocion Desactivada Temporalmente","estado":1} {"status":"Promocion Expirada.","estado":1} {"status":"Este codigo de barras ya ha sido utilizado","estado":0} {"status":"Copuon No es Valido por la fecha ","estado":0} </pre>

USES CASE
<ol style="list-style-type: none"> <li>1. El usuario realiza un pedido</li> <li>2. El empleado prepara el pedido e incluye en el ticket todos los elementos del pedido.</li> <li>3. El usuario utiliza sus promociones y cupones. El empleado activa el escáner: <div data-bbox="577 1352 1094 1832" data-label="Image">  </div> </li> <li>4. El cliente escanea el código de la promoción, enviando el código a CLM a través de este servicio (<b>Ws TPV/VALIDATE</b>)</li> <li>5. CLM responderá indicando la validez de la promoción.</li> </ol>

6. En caso afirmativo el TPV aplicará la promoción. En caso de error el TPV no aplicará la promoción en el ticket final.
7. EL cliente realizará el pago finalizando la transacción.

## 4. GENERAL USE CASE

1. El usuario realiza un pedido.
2. El empleado prepara el pedido e incluye en el ticket todos los elementos del pedido.
3. (**OPCIONAL**) El usuario quiere utilizar las promociones existentes en al app. El empleado activa el escáner desde el TPV:



- a. El cliente escanea el código de la promoción, enviando el código de la promoción a CLM a través del servicio (**Ws TPV/VALIDATE**)
  - b. CLM responderá indicando la validez de la promoción.
  - c. En caso afirmativo el TPV aplicará la promoción. En caso de error el TPV no aplicará la promoción en el ticket final.
4. (**OPCIONAL**) Una vez aplicada las promociones y finalizado el pedido, el cliente puede mostrar su código de cliente, para obtener puntos de fidelización.



En este caso se activará el lector de códigos y el cliente procederá a escanear este código. Una vez escaneado el código, el TPV enviará la información de la transacción invocando a este servicio (**WS TPV/LOYALTY**). El envío de la información debe de ser enviada cuando se vaya a finalizar la transacción.

5. El sistema envía la información de la transacción a CLM a través de este servicio (TPV/TRASANCTION).

## 5. ENVIOREMENTS

Env	Domains
PROD	<a href="https://srv.dunkincoffee.es">https://srv.dunkincoffee.es</a>
PREP	<a href="http://194.140.146.67:28080/">http://194.140.146.67:28080/</a>