



# DPS IVR Integration Requirements

Version 1.1

## COPYRIGHT

---

© Copyright 2012, Direct Payment Solutions (DPS) Limited  
98 Anzac Avenue  
PO Box 8400  
Auckland, 1150  
New Zealand  
[www.paymentexpress.com](http://www.paymentexpress.com)

All rights are reserved. No part of this work may be reproduced or copied in any form or by any means, electronic or mechanical, including photocopying, without the express written permission of Direct Payment Solutions.

## PROPRIETY NOTICE

---

The information described in this document is proprietary and confidential to Direct Payment Solutions (DPS). Any unauthorised use of this material is expressly prohibited except as authorised by Direct Payment Solutions in writing.

# CONTENTS

---

1	Introduction .....	1
1.1	Background and Overview .....	1
1.2	Assumptions.....	1
2	Call Script Process flow .....	2
2.1	Alterations to call script .....	5
3	Audio File Requirements .....	6
4	Customisation .....	9
4.1	Transaction variables .....	9
5	Transaction Outcomes .....	9
5.1	Payline .....	9
5.2	Auto reports .....	9
5.3	FPRN via SOAP .....	9
5.3.1	WSDL.....	9
6	Limitations.....	11
7	Testing .....	11
8	Appendix.....	11
	General Glossary:.....	11
	DPS ECOM Glossary: .....	11

# 1 INTRODUCTION

---

## 1.1 BACKGROUND AND OVERVIEW

The DPS IVR provides the ability for a DPS customer to take phone based payments inside of a DPS' PCI compliant environment. The call scripts used are configurable and may be varied on a per customer basis. The purpose of this document is to show the "off the shelf" call script and to provide a starting point for merchants to design their own call scripts. Merchants wishing to customise their call script will need to consult their allocated Project Manager at DPS to verify their requirements are able to be met.

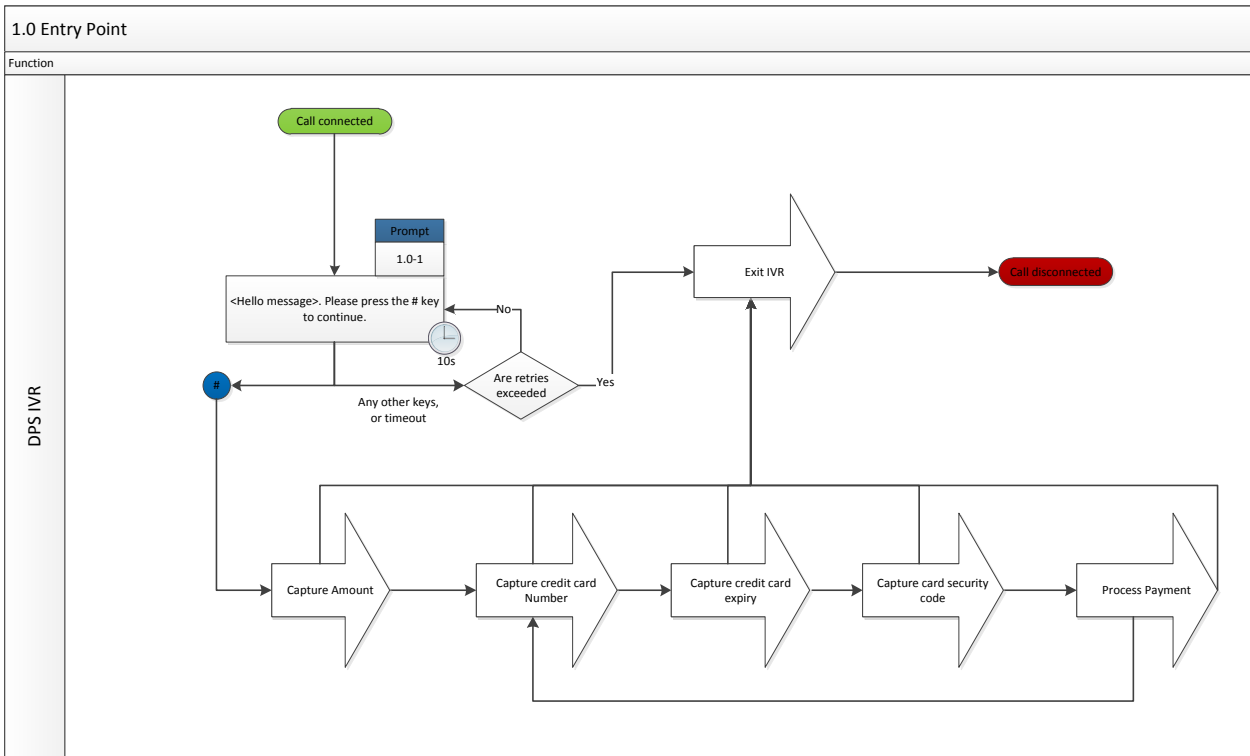
## 1.2 ASSUMPTIONS

Ref	Assumption	Reason
A01	When transferring a call to the DPS IVR the merchants system will fully de-trombone the call.	Failure to do so will mean DTMF tones pass through the merchants system and therefore leave them in scope for PCI requirements
A02	DTMF tones will be received by the DPS system out of band.	

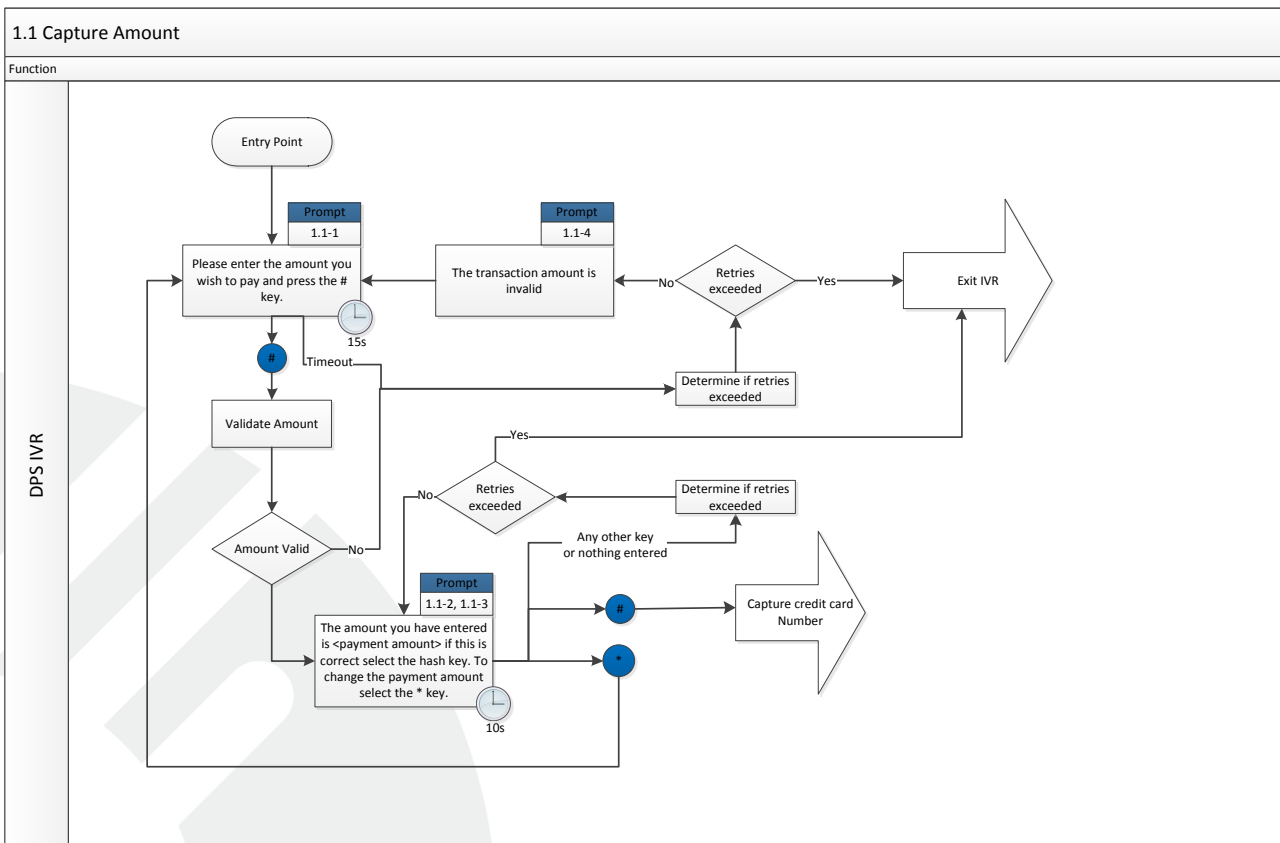
## 2 CALL SCRIPT PROCESS FLOW

This section documents the Call script process flow of the standard (off the shelf DPS IVR)

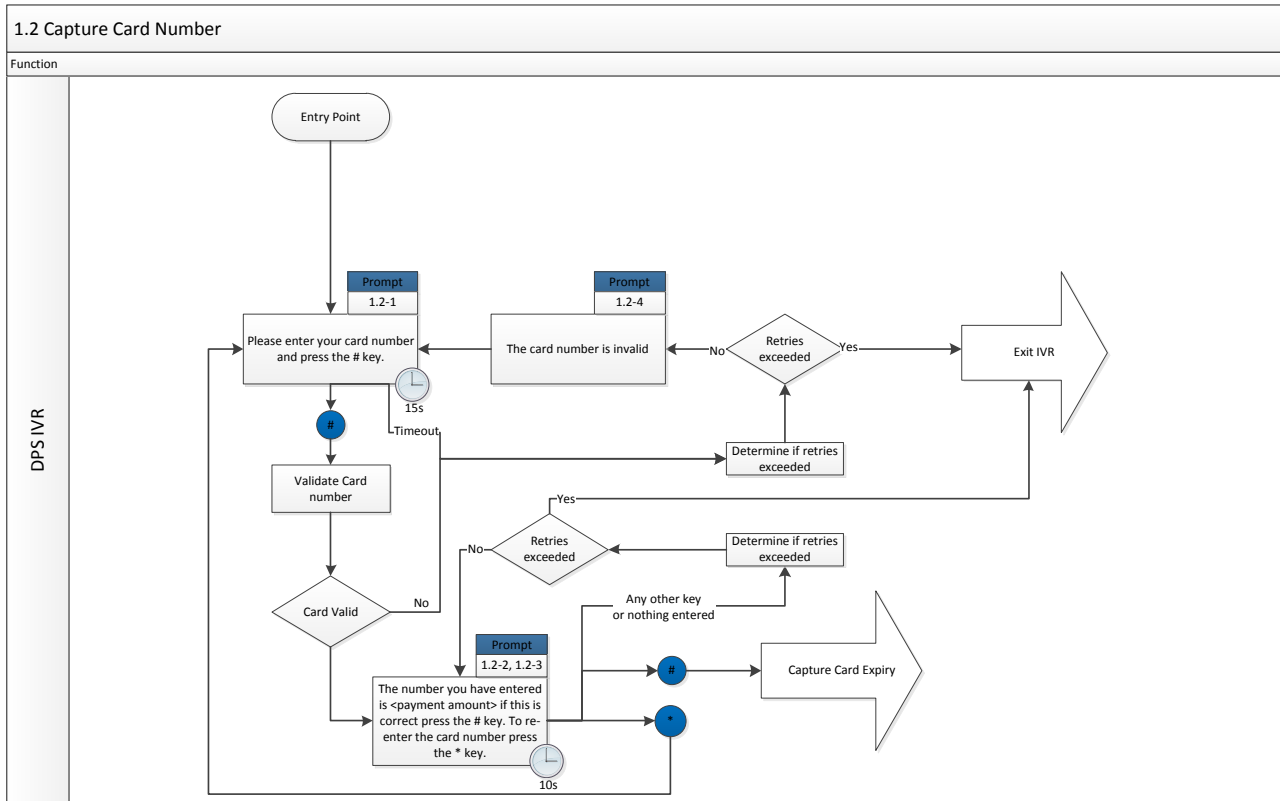
### High level with Entry point



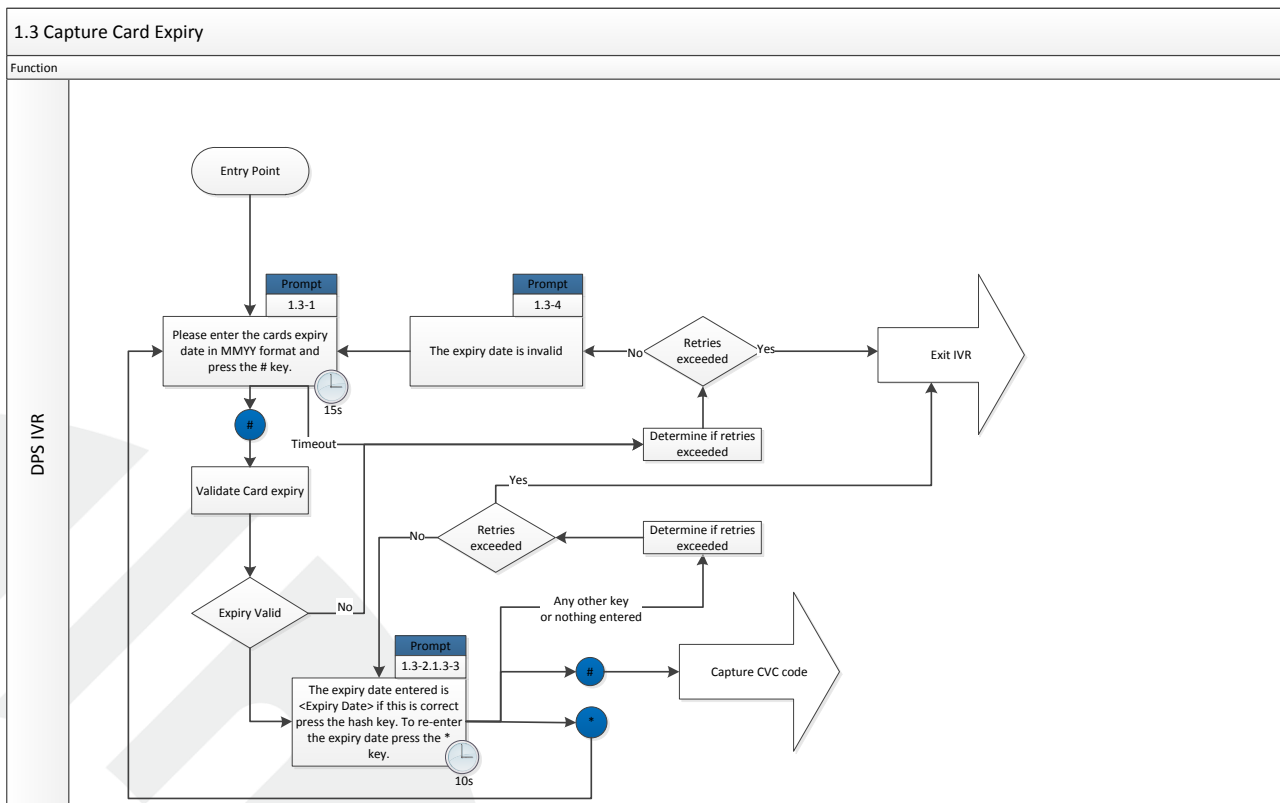
### Capturing the Amount



## Capturing the Credit card number

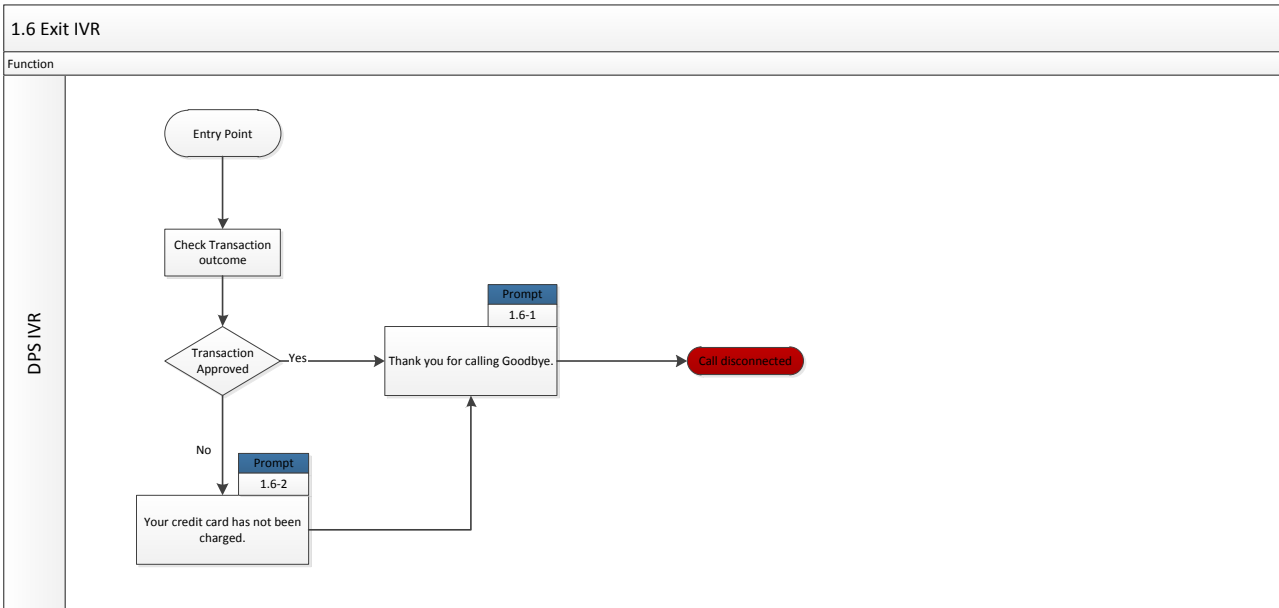


## Capturing the Expiry date



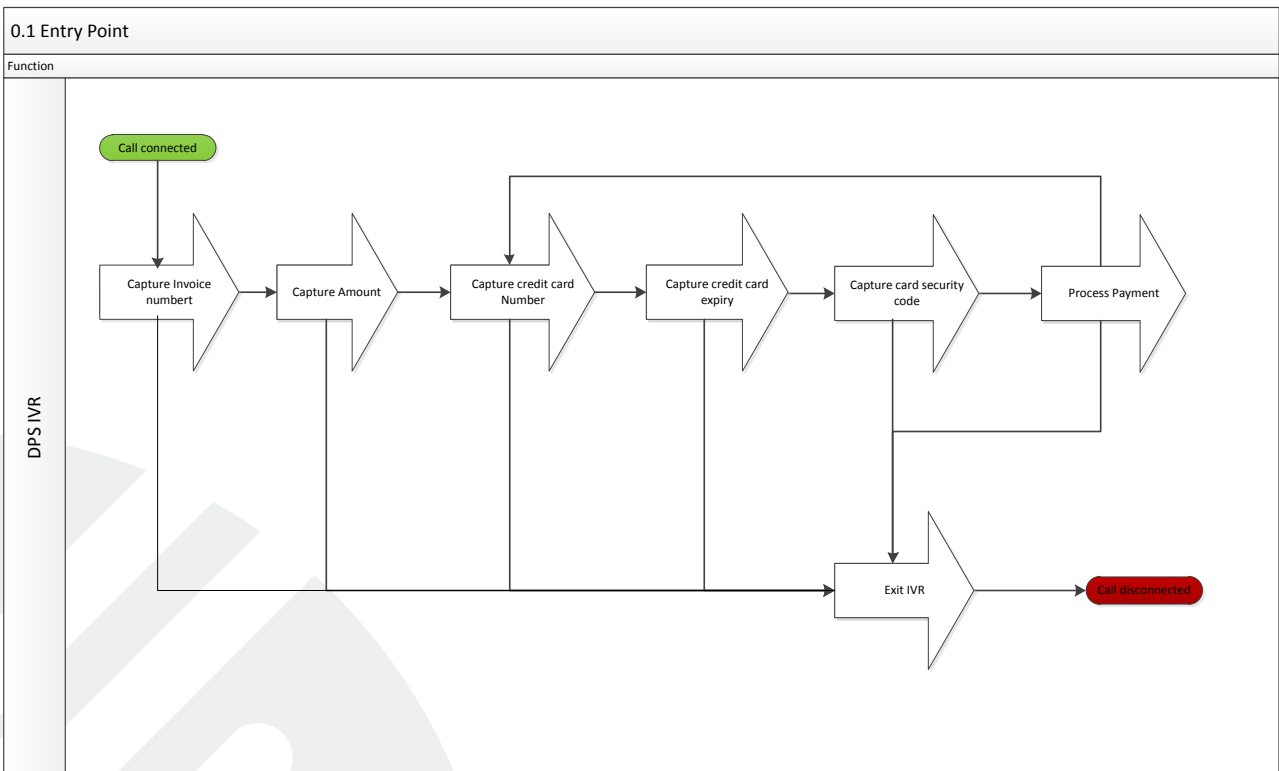


## Exiting the IVR



## 2.1 ALTERATIONS TO CALL SCRIPT

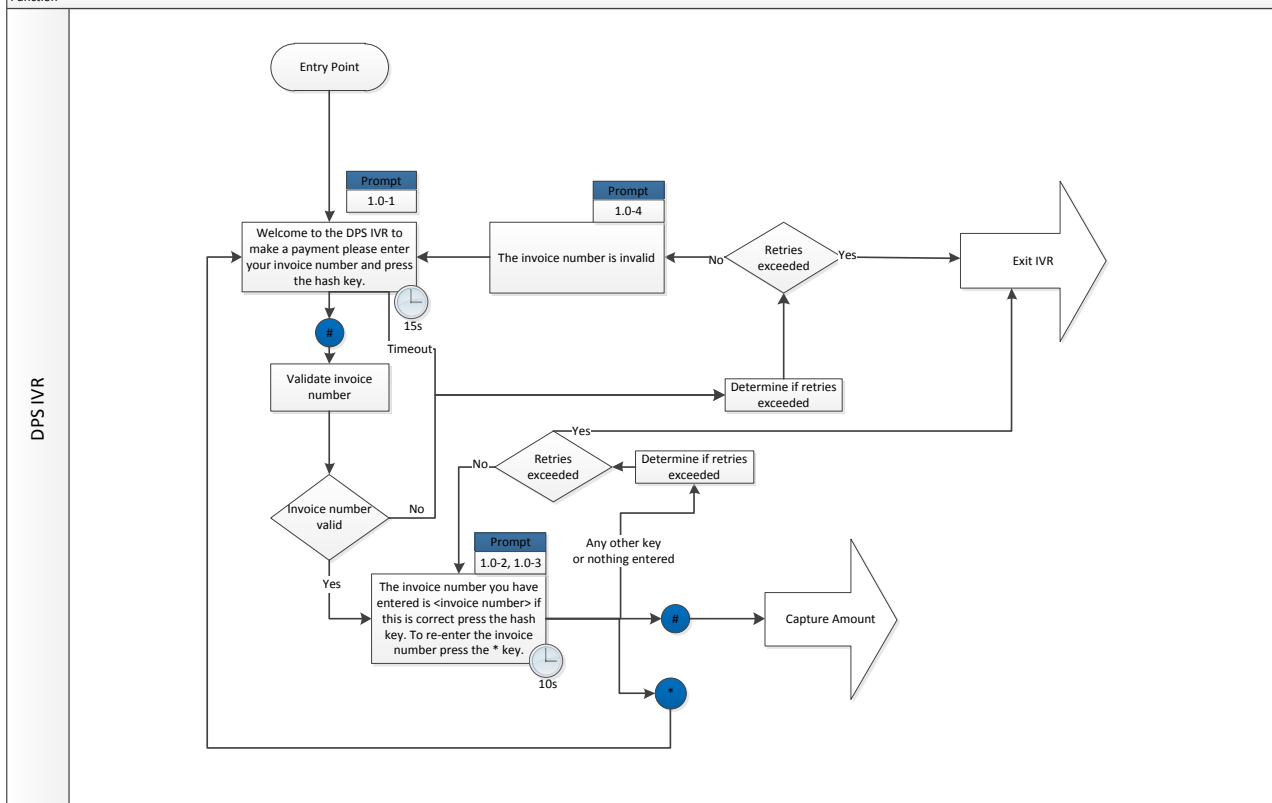
Alterations to the call script may be made as necessary; however alterations will impact on development timeframes. It is expected that in most cases merchants will require a way of matching or importing a transaction into their system. This may be achieved by providing a reference for a transaction that is returned in the transaction result. This could be a customer Id or perhaps an invoice number. In this scenario we recommend altering '1.0 Entry point' to give a general overview of the process flow and insert a new 1.0 similar to '1.1 Capture Amount' an example is below.





## 1.0 Capture Invoice number

Function



## 3 AUDIO FILE REQUIREMENTS

Each of the Prompts in the call script will require 1 or more audio files to be played. It is expected that the DPS customer requesting the IVR will supply these files named accordingly. The Call script field below is a guide and the customer may choose to alter wording for a given file provided this alteration does not influence the expected behaviour of the cardholder at the given prompt. Alternatively a text to speech option is available for customers not wishing to provide their own custom voice files. It is recommended that files have consistent quiet space at beginning and end so that when numbers are being played back to cardholders the speech pattern is consistent.

The audio file format should be 8 bit, 8 kHz, u-low.

Call Script Prompts		
Prompt	File Name	Call Script
1.0-1	1.0-1.wav	Welcome to xxx please press # to enter your payment details.
1.1-1	1.1-1.wav	Please enter the amount you wish to pay and press the # key.
1.1-2	1.1-2.wav	The amount you have entered is
1.1-3	1.1-3.wav	If this is correct press the # key. To change the payment amount press the * key.
1.1-4	1.1-4.wav	The transaction amount is invalid.
1.2-1	1.2-1.wav	Please enter your credit card number and press the # key.
1.2-2	1.2-2.wav	The number you have entered is
1.2-3	1.2-3.wav	If this is correct press the # key. To re-enter the card number press

		the * key.
1.2-4	1.2-4.wav	The card number is invalid.
1.3-1	1.3-1.wav	Please enter the card's expiry date in month month year year format and press the # key.
1.3-2	1.3-2.wav	The expiry date entered is
1.3-3	1.3-3.wav	If this is correct press the # key. To re-enter the expiry date press the * key.
1.3-4	1.3-4.wav	The expiry date is invalid.
1.4-1	1.4-1.wav	Please enter your 3 digit card security code and press the # key. This is usually located on the back of your card
1.4-1A	1.4-1A.wav	Please enter your 4 digit card security code and press the # key.
1.4-2	1.4-2.wav	The card security code entered is
1.4-3	1.4-3.wav	If this is correct press the # key. To re-enter the card security code press the * key.
1.4-4	1.4-4.wav	The card security code is invalid.
1.5-1	1.5-1.wav	Please wait while your transaction is processed.
1.5-2	1.5-2.wav	Thank you your transaction has been approved.
1.5-3	1.5-3.wav	I'm sorry your transaction has been declined.
1.5-4	1.5-4.wav	Press * to try another card, or hang up to exit.
1.6-1	1.6-1.wav	Thank you for calling, goodbye
1.6-2	1.6-2.wav	Your credit card has not been charged.

Playback Prompts		
Prompt	File Name	Call Script
0	Zero.wav	0
1	One.wav	1
2	Two.wav	2
3	Three.wav	3
4	Four.wav	4
5	Five.wav	5
6	Six.wav	6
7	Seven.wav	7
8	Eight.wav	8
9	Nine.wav	9
10	Ten.wav	10
11	Eleven.wav	11
12	Twelve.wav	12
13	Thirteen.wav	13
14	Fourteen.wav	14
15	Fifteen.wav	15
16	Sixteen.wav	16
17	Seventeen.wav	17
18	Eighteen.wav	18
19	Nineteen.wav	19
20	Twenty.wav	20
30	Thirty.wav	30
40	Fourty.wav	40
50	Fifty.wav	50
60	Sixty.wav	60
70	Seventy.wav	70
80	Eighty.wav	80
90	Ninety.wav	90
Hundred	Hundred.wav	Hundred
Thousand	Thousand.wav	Thousand
And	and.wav	And
Cent	cent.wav	Cent

Cents	cents.wav	Cents
Dollar	dollar.wav	Dollar
Dollars	dollars.wav	Dollars
January	January.wav	January
February	February.wav	February
March	March.wav	March
April	April.wav	April
May	May.wav	May
June	June.wav	June
July	July.wav	July
August	August.wav	August
September	September.wav	September
November	November.wav	November
October	October.wav	October
December	December.wav	December
First	First.wav	First
Second	Second.wav	Second
Third	Third.wav	Third
Fourth	Fourth.wav	Fourth
Fifth	Fifth.wav	Fifth
Sixth	Sixth.wav	Sixth
Seventh	Seventh.wav	Seventh
Eighth	Eighth.wav	Eighth
Ninth	Ninth.wav	Ninth
Tenth	Tenth.wav	Tenth
Eleventh	Eleventh.wav	Eleventh
Twelfth	Twelfth.wav	Twelfth
Thirtieth	Thirtieth.wav	Thirtieth
Fourteenth	Fourteenth.wav	Fourteenth
Fifteenth	Fifteenth.wav	Fifteenth
Sixteenth	Sixteenth.wav	Sixteenth
Seventeenth	Seventeenth.wav	Seventeenth
Eighteenth	Eighteenth.wav	Eighteenth
Nineteenth	Nineteenth.wav	Nineteenth
Twentieth	Twentieth.wav	Twentieth
Twenty-first	Twenty-first.wav	Twenty-first
Twenty-second	Twenty-second.wav	Twenty-second
Twenty-third	Twenty-third.wav	Twenty-third
Twenty-fourth	Twenty-fourth.wav	Twenty-fourth
Twenty-fifth	Twenty-fifth.wav	Twenty-fifth
Twenty-sixth	Twenty-sixth.wav	Twenty-sixth
Twenty-seventh	Twenty-seventh.wav	Twenty-seventh
Twenty-eighth	Twenty-eighth.wav	Twenty-eighth
Twenty-ninth	Twenty-ninth.wav	Twenty-ninth
Thirtieth	Thirtieth.wav	Thirtieth
Thirty-first	Thirty-first.wav	Thirty-first

Notes:

\* Should be read as "star"

# should be read as "hash"

## 4 CUSTOMISATION

---

The functionality of the IVR call flow may be customised to suit your needs on a case by case basis. Additional charges may apply, please contact one of our Sales representatives ([sales@paymentexpress.com](mailto:sales@paymentexpress.com)) for more information. Preferably the low level prompts above would not be changed but the need may arise to obtain additional input parameters from the cardholder. It is suggested that any additional prompts appear before the prompts to capture card information. If this is required please document the process flow as above and supply to your DPS Project Manager (once available and project has been initiated) for verification.

One common requirement is to have a merchant provided unique identifier for a transaction. The call script can be structured to have this variable passed in by the merchants system before transferring the call thus making this step invisible to the cardholder. Alternatively a prompt may be added to ask the cardholder to provide the reference (perhaps an invoice Id).

### 4.1 TRANSACTION VARIABLES

The IVR on completion of its call script processes transactions via DPS' PxPost interface. As such it is possible to customise the PxPost input parameters to meet the needs of an individual merchant. Retrieving these variables is covered in section 5.

## 5 TRANSACTION OUTCOMES

---

IVR transactions take place on the DPS' system; therefore the need arises to have mechanisms that will allow the Merchants system to become aware of transactions that have been processed. There are various options available to achieve this.

### 5.1 PAYLINE

A user account to log into the DPS system to view transactions is made available by default for all DPS customers. Reports may also be generated and obtained through this interface.

### 5.2 AUTO REPORTS

A Report that contains transactions processed on a merchant account may be generated and sent on a periodic basis (usually daily). The method of delivery is optional between sFTP and email. Please contact one of our Sales representatives ([sales@paymentexpress.com](mailto:sales@paymentexpress.com)) for more information.

### 5.3 FPRN VIA SOAP

A real-time soap request may be generated with the transaction details and sent to a web server hosted by the merchant. This option allows a merchant to update their system in real time with transaction outcomes. It requires the merchant to develop a SOAP web server that conforms to a specification provided by DPS in the WSDL below.

#### 5.3.1 WSDL

```
<?xml version='1.0' encoding='UTF-8'?>
<definitions xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:soapenv="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tns="https://sec.paymentexpress.com/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns="http://schemas.xmlsoap.org/wsdl/"
targetNamespace="https://sec.paymentexpress.com/" name="FPRN">
<types>
<xs:schema version="1.0" targetNamespace="https://sec.paymentexpress.com/" >
<xs:element name="FailureProofResultNotificationHeader" type="tns:FailureProofResultNotificationHeader" />
<xs:element name="FailureProofResultNotification" type="tns:FailureProofResultNotification" />
<xs:element name="FailureProofResultNotificationResponse" type="tns:FailureProofResultNotificationResponse" />
<xs:complexType name="FailureProofResultNotificationHeader">
<xs:sequence>
<xs:element name="SecurityTokens" type="tns:arrayOfToken" nillable="true" minOccurs="1" maxOccurs="1" />
<xs:element name="ApplicationData" type="tns:arrayOfToken" nillable="true" minOccurs="1" maxOccurs="1" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="FailureProofResultNotification">
<xs:sequence>
<xs:element name="request" type="tns:serviceRequestType" nillable="true" minOccurs="1" maxOccurs="1" />
</xs:sequence>

```

```

</xs:complexType>
<xs:complexType name="serviceRequestType">
  <xs:sequence>
    <xs:element name="TransactionData" type="tns:arrayOfToken" nillable="true" minOccurs="1" maxOccurs="1" />
  </xs:sequence>
</xs:complexType>

<xs:complexType name="arrayOfToken">
  <xs:sequence>
    <xs:element name="Token" type="tns:token" nillable="true" minOccurs="0" maxOccurs="unbounded" />
  </xs:sequence>
</xs:complexType>
<xs:element name="ArrayOfToken" nillable="true" type="tns:arrayOfToken" />
<xs:complexType name="token">
  <xs:attribute name="Name" type="xs:string" />
  <xs:attribute name="Value" type="xs:string" />
</xs:complexType>
<xs:complexType name="FailureProofResultNotificationResponse">
  <xs:sequence>
    <xs:element name="Results">
      <xs:simpleType>
        <xs:restriction base="xs:integer">
          <xs:enumeration value="0" />
          <xs:enumeration value="1" />
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
  </xs:sequence>
</xs:complexType>
</xs:schema>
</types>
<message name="FailureProofResultNotificationHeader">
  <part name="requestHeader" element="tns:FailureProofResultNotificationHeader" />
</message>
<message name="FailureProofResultNotification">
  <part name="parameters" element="tns:FailureProofResultNotification" />
</message>
<message name="FailureProofResultNotificationResponse">
  <part name="parameters" element="tns:FailureProofResultNotificationResponse" />
</message>
<portType name="FPRN">
  <operation name="FailureProofResultNotification">
    <input message="tns:FailureProofResultNotification" />
    <output message="tns:FailureProofResultNotificationResponse" />
  </operation>
</portType>
<binding name="FPRNPortBinding" type="tns:FPRN">
  <soapenv:binding transport="http://schemas.xmlsoap.org/soap/http" style="document" />
  <operation name="FailureProofResultNotification">
    <soapenv:operation soapAction="fprn" />
    <input>
      <soapenv:header message="tns:FailureProofResultNotificationHeader" part="requestHeader" use="literal" />
      <soapenv:body use="literal" />
    </input>
    <output>
      <soapenv:body use="literal" />
    </output>
  </operation>
</binding>
<service name="FPRN">
  <port name="FPRNPort" binding="tns:FPRNPortBinding">
    <soapenv:address location="https://sec.paymentexpress.com/FPRN" />
  </port>
</service>
</definitions>

```

Note:

- Variable length arrays of key value pairs are utilised for both the security tokens and the transaction information to avoid lax versioning: that is, it does not throw any exceptions for new unknown data members in received data.
- Unexpected/unrequired key value pairs must be ignored.

## 6 LIMITATIONS

---

The IVR can accept / allow the cardholder to input buttons over the top of the audio files. However, only the last audio file in a prompt (that contains a sequence of audio files) may be skipped by user input.

## 7 TESTING

---

A test IVR will be made available for the merchant to perform validation that the IVR meets their agreed upon requirements. Once the Merchant has signed off the IVR requirements they may request the code be deployed into the production environment. The environment will remain available for 5 working days following the production deployment. After which it may be taken down and made available to other merchants waiting for development testing.

## 8 APPENDIX

---

### GENERAL GLOSSARY:

Term	Description
Assumption	A condition not certain to happen, outside the control of the project, and necessary for the project or solution to be successful.
Constraint	A condition that limits the project or solution, is outside the control of the project, and needs to be managed around.
Dependency	A relationship between conditions where one cannot start or finish until another starts or finishes.
Recovery Point Objective	This describes the acceptable amount of data loss measured in time.
Recovery Time Objective	This describes the duration of time and a service level within which a business process must be restored after a disaster (or disruption) in order to avoid unacceptable consequences associated with a break in business continuity.
SLC	Solution Life Cycle
API	Application Programming Interface
BAU	"Business As Usual"
EULA	End User License Agreement

### DPS ECOM GLOSSARY:

Term	Description
PX Pay	DPS Hosted Payments Package
PX Post	Merchant Hosted Payments Package
3-D Secure	3 Domains Secure. E-Commerce environment including Acquirers/Merchants, Issuers/Cardholders and Card Schemes. 3-D Secure adds an authentication step for Internet txns where the card holder is redirected to his/her card issuing bank's 3-D Secure page to enter a password or special code. It is currently supported by only some acquirers and only some card issuing banks.
AAV	Accountholder Authentication Value. Unique reference generated by MasterCard and Maestro card issuers to prove authentication took place.
CAVV	Cardholder Authentication Value. Supplied by card issuer as part of a successful 3D

	Secure authentication.
BIN	Bank Identification Number. The first six digits of the number of various financial cards, such as Visa or Mastercard. These digits identify which organization issued it.
CVC/CVV	Stands for "Card Verification Code"/"Card Verification Value", it is the 3-digit number on the back of credit cards used for security purposes.
PCI-DSS	Payment Card Industry - Data Security Standards
ECI	<p>When processing Internet transactions the Electronic Commerce Indicator (ECI) must be included on the payment transaction message format to show that the transaction originated from an Internet source.</p> <p>If an ECI is sent with values of 5, 6, or 7, the transactions are noted as a secure ECI transaction and must be processing card data securely. If transactions are submitted with an ECI value of 8 or 9, the card data is in a non-secure format and the transaction will incur a high MSF (Merchant Service Fee).</p>
MPI	Merchant plug-in. Software that facilitates the 3D Secure process (sending and receiving of PAREq/PARes) through to Visa/MasterCard.
FPRN	Fail-Proof Result Notification