



Online Services Corp (OSC) – Prepaid Card Solution

2024 - 2025

This presentation is confidential and has been prepared exclusively for internal use. This presentation and its contents are wholly confidential and may not be further distributed or passed on to any other person or published or reproduced, in whole or in part, by any medium or in any form for any purpose.



Product Introduction

Introduction

About

OSC has implemented a prepaid card issuing program

- *Provide a cost effective, easy to implement, white label, prepaid issuing solution for issuers, merchants and processors*
- *Issued under the UnionPay brand, the cards can be used at all UnionPay accepting merchants (depending on card type)*
- *Various card programs and types are supported depending on market and currency*
- *High availability with quality support*
- *Account data accessed via online portal or APIs*

Ideal for merchants who periodically need to pay out; such as Gambling, Gaming and FX/CFD providers; they can pay their clients with a globally accepted card instead of using international transfers



Issuance Model

Form Factor	Plastic Card	Virtual Card
Delivery	Contactless/Chip/Magnetic Stripe	Digital Only
Cardholder	Anonymous	Personalised
Funds	Reloadable	Disposable





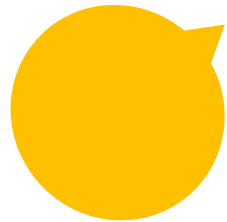
Card and Account Types

Virtual Cards



No physical card

A virtual card is a UnionPay account issued without a physical plastic card



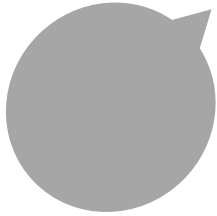
The elements of a virtual card

Card Number, Expiry Date, CVN2

- Virtual cards can be used with mobile applications on NFC enabled mobile devices at UnionPay contactless enabled merchant POS machines
- Virtual cards are most often used online; and they can be used for e-commerce and online transactions wherever UnionPay cards are accepted

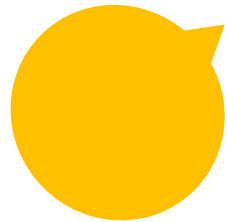


Disposable Cards



Single load cards with variable load amounts

Once the balance of the account is \$0, the card can no longer be used



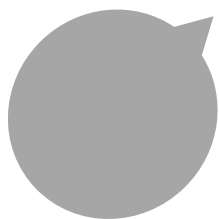
Virtual cards are most often disposable

Some virtual cards can be reloaded; most cannot

- Disposable cards are often issued as non-personalized, anonymous gift cards
- Reloadable cards can be loaded and reloaded multiple times; the card can be used until expiry, similar to a bank account debit card



Anonymous Cards



Cardholder is not linked to the account

No KYC required



Limits on funding parameters

Maximum load values apply

- Anonymous cards are treated like cash - stolen or lost cards can not be refunded; chargebacks are not accepted; refunds need to be raised with the merchant directly
- Anonymous cards have lower maximum load values – our anonymous cards can be loaded once only with a maximum value of AUD1,000
- Cardholders can apply for and hold any number of anonymous cards at the same time





Product Proposition

Product Details

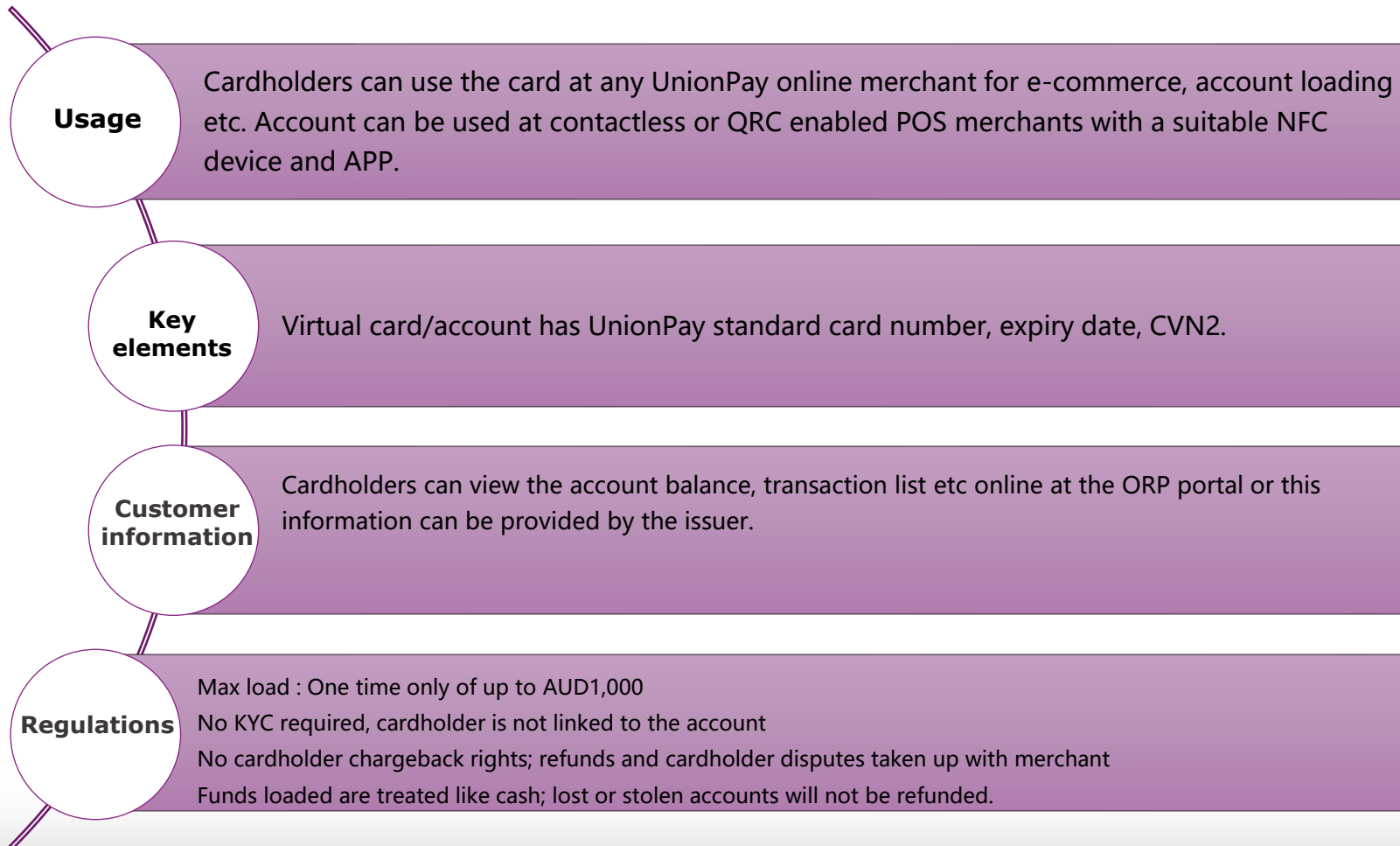
Functions		Virtual Card		Physical Card	
		Reloadable	Disposable	Reloadable	Disposable
ATM	Balance Inquiry	×	×	√	√
	Cash Withdrawal	×	×	√	√
POS	Purchase	O ¹	O ¹	√	√
EC	UnionPay Online Payment (UPOP)	√	√	√	√
Pre-Authorization		√	√	√	√
Mail/Telephone Order (MOTO)		√	√	√	√
Recurring Payments		√	√	√	√
Funding & Transaction Limits		AUD1,00 for anonymous cards	AUD1,000 for anonymous cards	AUD10,000 for personalised cards	AUD10,000 for personalised cards
KYC Required		Not required for anonymous cards	Not required for anonymous cards	Only required for personalised cards	Only required for personalised cards

¹ Virtual cards can be used with mobile applications on NFC enabled mobile devices at UnionPay contactless enabled merchant POS machines



Virtual, Anonymous, Disposable

Our most common and easy to implement product is a virtual, anonymous, non-reloadable UnionPay card account





Interface to the Issuing Platform

Issuer Platform Interface

Components

Account Management to enable cardholders and manage their accounts

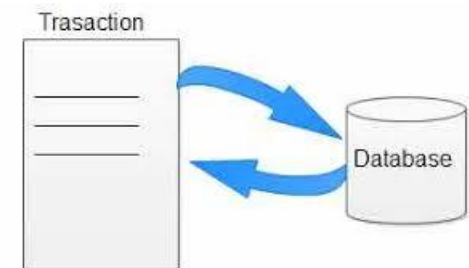
- *API based, server hosted architecture to securely and simply interface your systems*

Online Admin to manage your accounts and cardholders

- *Secure browser access to fully featured customer portal for cardholder and account management and reports access*

System Monitoring to monitor your transactions and manage your systems

- *Graphical user interface to a time series database to control your transactions and systems*



Online Admin

Example Screens

Account Management for Issuers

Card Number ↑	Expiry Date	Currency	Balance
	09/29	AUD	283.39
	09/29	AUD	291.00

Transaction Management

Date & Time ↓	Merchant	Amount	Result	Type
11/09/2024 22:33:09		AUD291.00	Success	Load
20/09/2024 12:49:53		USD5.06	Success	Purchase

Showing 2 transactions (filtered from 2 total transactions)

[Download Transaction List](#)

Manage User Settings

PrepaidPoC	
Login ID	<input type="text" value="PrepaidPoC"/>
User ID	<input type="text" value="PrepaidPoC"/>
Email	<input type="text"/>
Mobile	<input type="text" value="11223344"/>
Issuer Name	<input type="text"/>
Issuer ID	<input type="text" value="RS-20242908"/>





Support & Pricing

Tools and Support

We support implementation

- *Documentation – Product Services and API guides*
- *Sample source code – each API interface call includes sample source code which can be used to expedite implementation*
- *A test system available to issuers and processors to test their implementation*
- *Access to the transaction, settlement and customer information through the Customer Admin portal*
- *Support for implementation from multiple offices based in Asia, Australia and Europe*
- *We provide an end-to-end White Label system, branded by processors, issuers and merchants with their own brands – including documentation, merchant admin, API calls etc*



```
except socket.error as msg:
    print "socket error (%s) for host %s (%s)" % (msg, host, ip)
for h3 in page.findAll("tbody"):
    value = (h3.contents[0])
    if value is "Afdeling":
        print >> txt, value
        import codecs
        f = codecs.open("alle.txt", "w", encoding="utf-8")
        text = f.read()
        f.close()
        # open the file again for writing
        f = codecs.open("alle.txt", "w", encoding="utf-8")
        f.write(value+"\n")
        # write the original contents
        f.write(text)
        f.close()
```



Pricing

Pricing is transparent and based on

- *Per account fees*
- *One time load fees*

We will provide a complete fee structure - there are no hidden or extra charges

