

A digital wallet to receive and spend
Singapore Government payouts in a
secure and convenient way



Overview

1 Introduction & walkthrough of GovWallet

2 Technical deep dives

3 Question & Answer

Introduction & walkthrough of GovWallet



Immanuella Lim
Software Engineer
Government Digital Services, GovTech

GovWallet Trivia

Oct 2020

Nov 2021

Apr 2022

Trial

**CPFIB GovCash
Launch**

**Payment
via PayNow**

**Payment
via NETS**

**Payment
via NETS**

**SingPass
Mobile Info
Card Launch**

Started as a trial
within GovTech

Allow for cash
withdrawal from
OCBC ATMs

Allow PayNow
payment via the
LifeSG app

Allow NETS
payment via the
H365 app

Allow NETS
payment via the
LifeSG app

View GovWallet
information via
the SingPass
mobile app

Agencies we've worked with thus far



Central
Provident
Fund Board



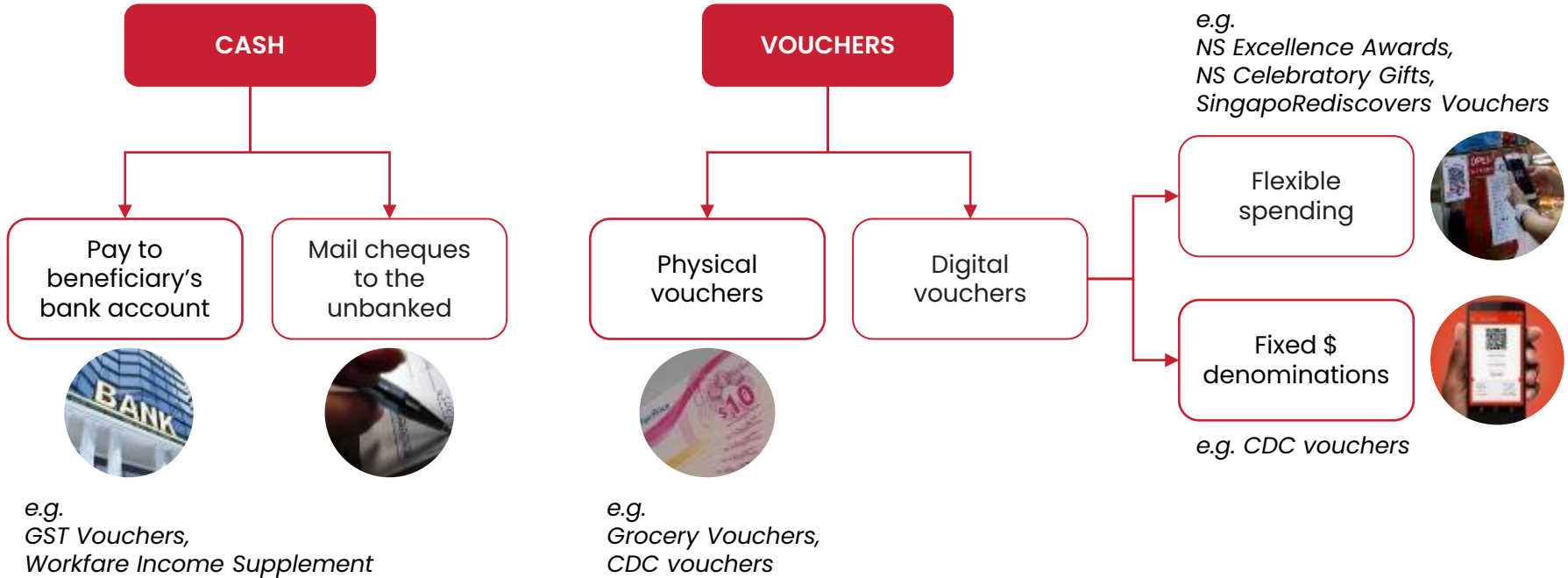
MINDEF
SINGAPORE



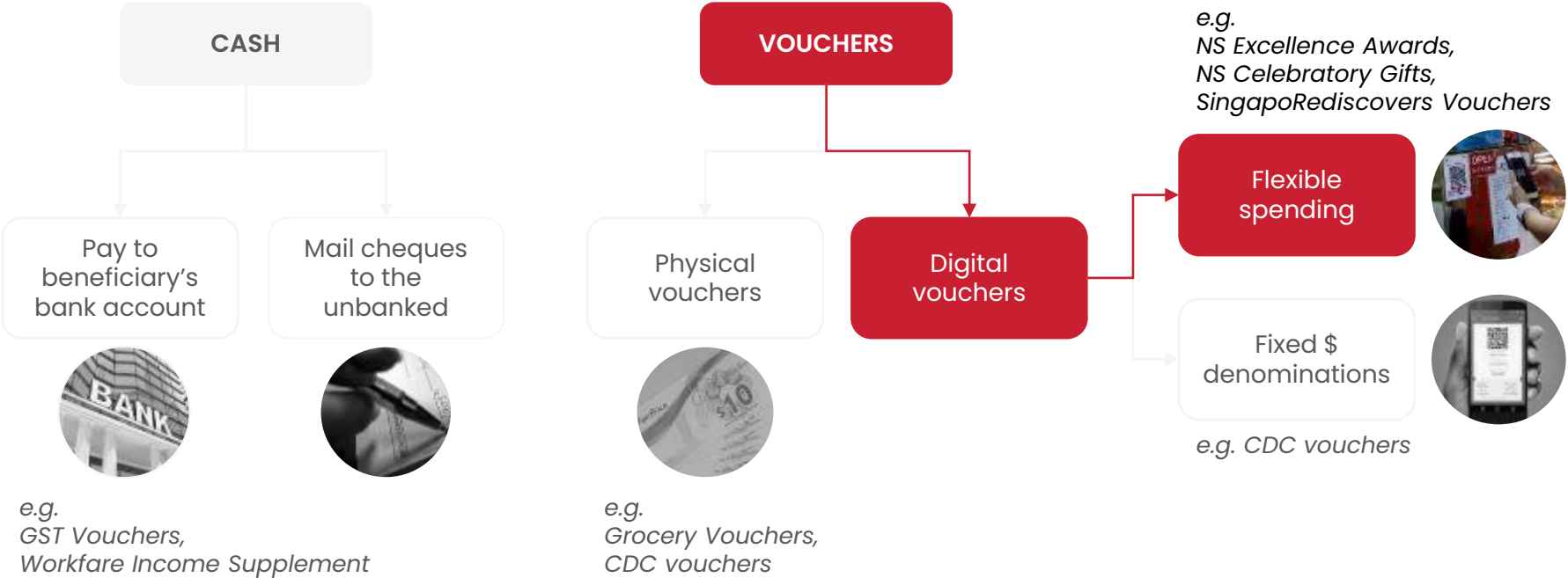
Health
Promotion
Board



Various modes of government payouts



How to make credits spending flexible



**What are some challenges faced
with using voucher payouts?**



Self-employed home-baker & mother

Sue is a self-employed home-baker and a mother. She lives a busy life, striving to balance motherhood with her home-baking business.

She is educated and tech savvy and she relies on her digital devices to remind her on the appointments she needs to keep up.

Last year, Sue was given some government vouchers to defray her household expenses. In her busyness, she often forgot to bring them out or use them before the vouchers expire.

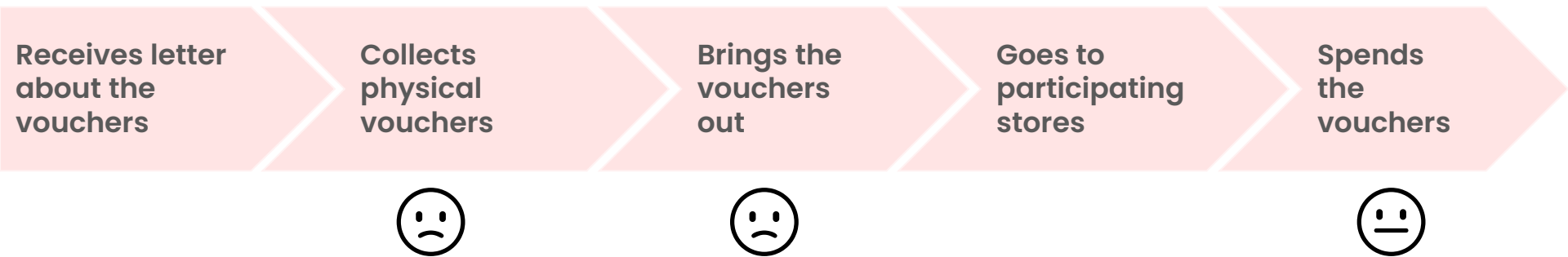
“I just wish I remembered to bring out these vouchers and used them before they expired!”



Photo by kevin liang on Unsplash



Sue's As-Is Journey





Government officer

Gabriel is a government officer in-charge of the citizens voucher disbursement program. He has been with the government agency for more than 10 years and each year, when it comes to the voucher giving period,

Gabriel would be busy coordinating work with different departments such as procurement team to procure vouchers, logistics partner to distribute the vouchers, engagement team to onboard the merchants and finance team for merchants reimbursement.

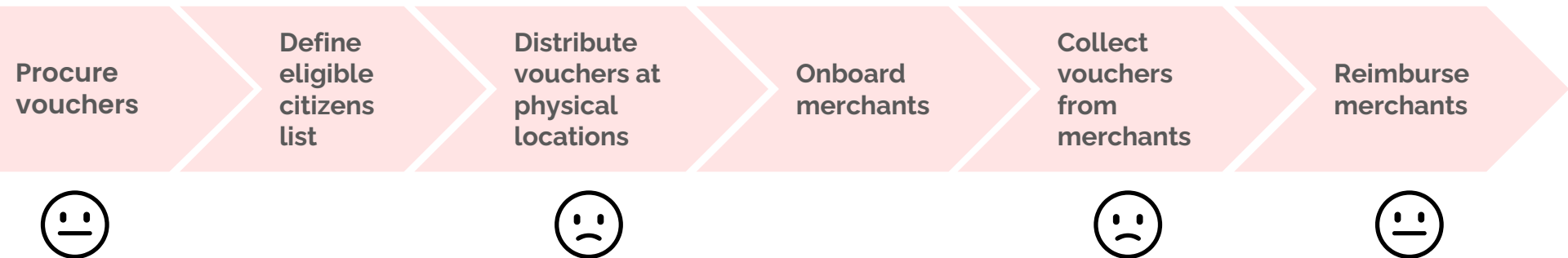
“I am always so busy during this period. Is there a better way to disburse the vouchers and reimburse the merchants?”



Photo by Wes Hicks on Unsplash



Gabriel's As-Is Journey





Auntie Annie

— — —

Friendly auntie that operates a drink store

Auntie Annie operates a drink store at the market.

In recent years, there's so many different methods of paying for your drink and it can get a bit confusing for Auntie.

She also has to take time to exchange vouchers for cash or learn a new tool to accept voucher payments.

“Aiyo, so many ways to pay these days. How to remember and learn everything?”



Photo by John T on Unsplash



Auntie Annie's As-Is Journey

Learn about new vouchers



Learn operational process of accepting vouchers



Count vouchers (if physical)



Reimburse vouchers (if physical)



Wait for reimbursement

Summary of challenges faced

 **Sue**

Must remember to collect vouchers

Must remember to use up vouchers before it expires

 **Gabriel**

Need to procure vouchers

Need to onboard merchants

Need to process physical vouchers and pass merchants cash

 **Auntie Annie**

Need to learn about new voucher scheme & operational steps to accept the vouchers

Need to wait for cash proceeds after submitting physical vouchers

**Why leverage on current
payment ecosystems?**

164,000

This is the number of merchants that have registered for SGQR PayNow as at September 2021.

Examples: Shopee, Lazada, Giving.sg

Source: The Association of Banks in Singapore (ABS)

120,000



This is the number of acceptance points that NETS Group supports.

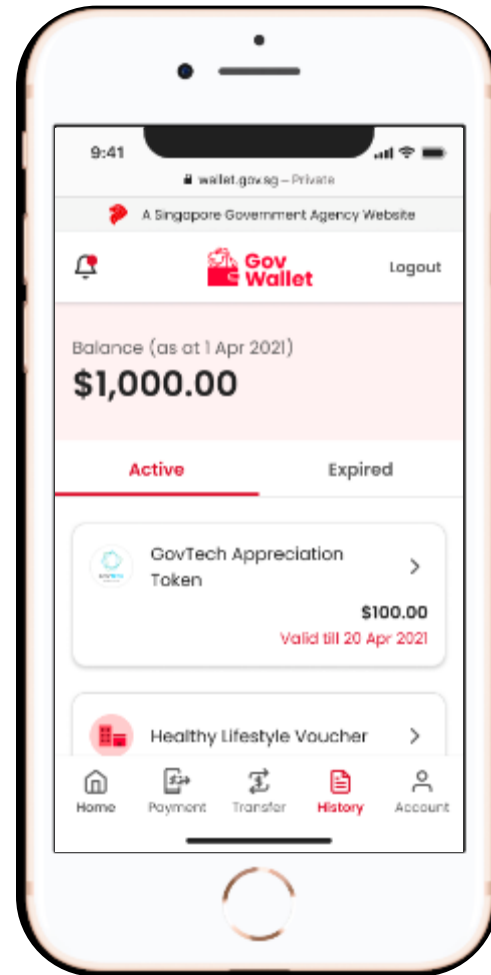
How can we improve this?

- Direct disbursement of credits to beneficiaries
- Giving reminders on unspent credits
- Showing past spending transactions
- Leverage on current payment ecosystems to create little to zero merchant onboarding
- Faster reimbursement to merchants
- Allow selection of specific merchants to spend at

How would the **improvements** look like for Sue, Gabriel & Auntie Annie?

GovCredits in GovWallet

Easily spend your payouts like
any other e-wallet mobile apps



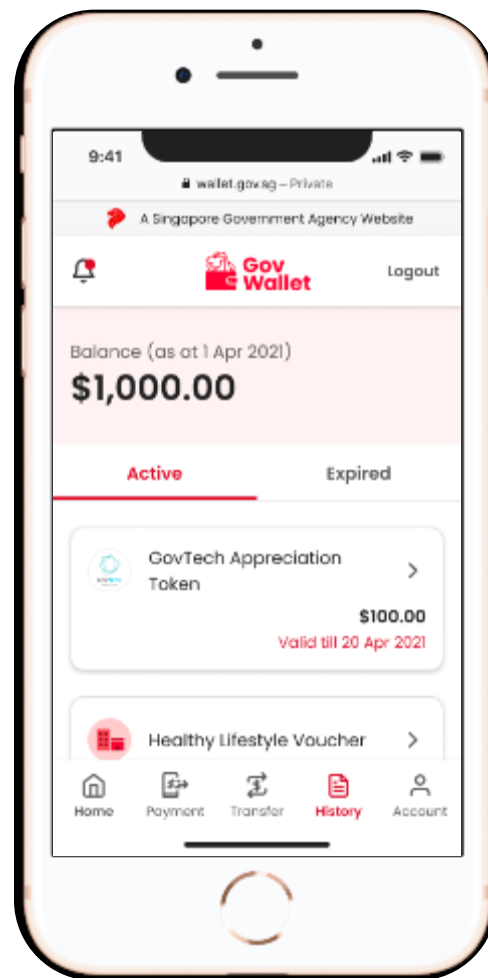
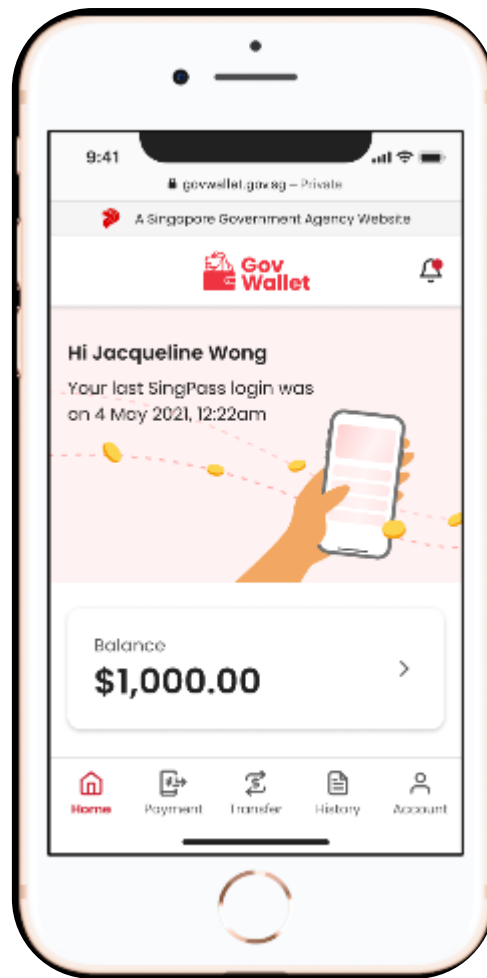
*Images are for illustration purposes only

Direct disbursement into Sue's GovWallet

 **Gabriel**

Gabriel can now instruct GovWallet to disburse payouts to the list of eligible citizens that his agency have identified.

There's no need to procure vouchers and coordinates of the location for redemption.



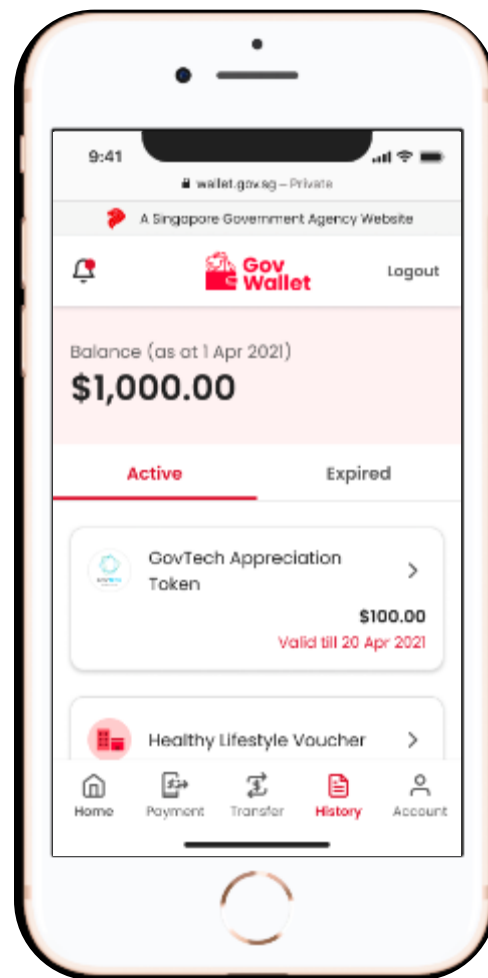
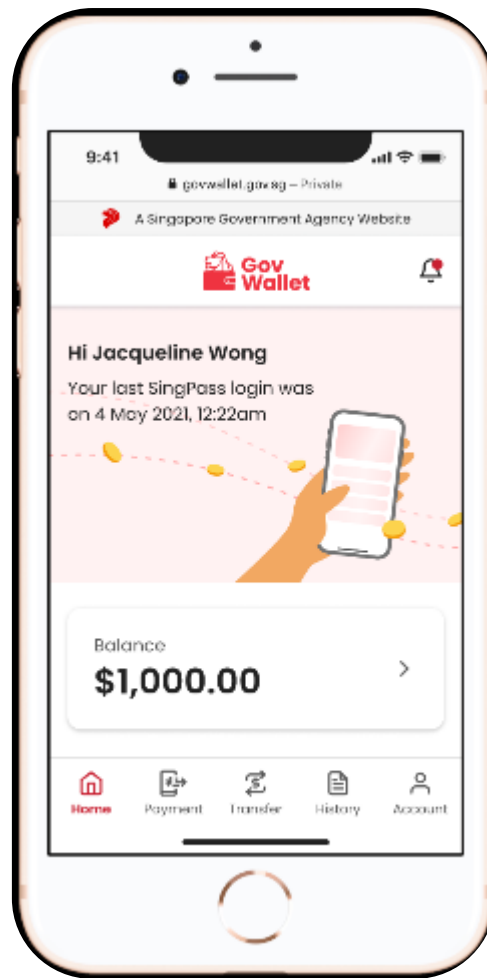
*Images are for illustration purposes only

Direct disbursement into Sue's GovWallet



Sue can now see all her government payouts that she is entitled to in a single place.

There's no need to set aside a date and time to collect the vouchers at a physical location.

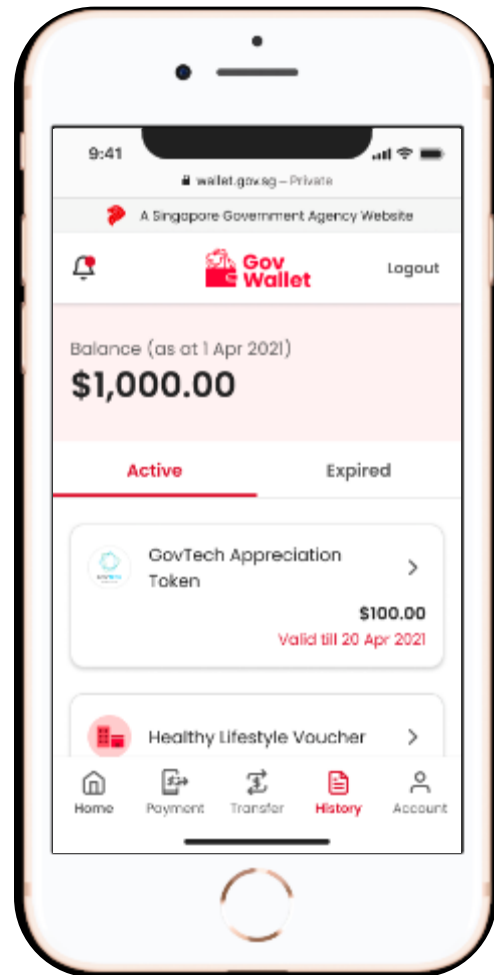
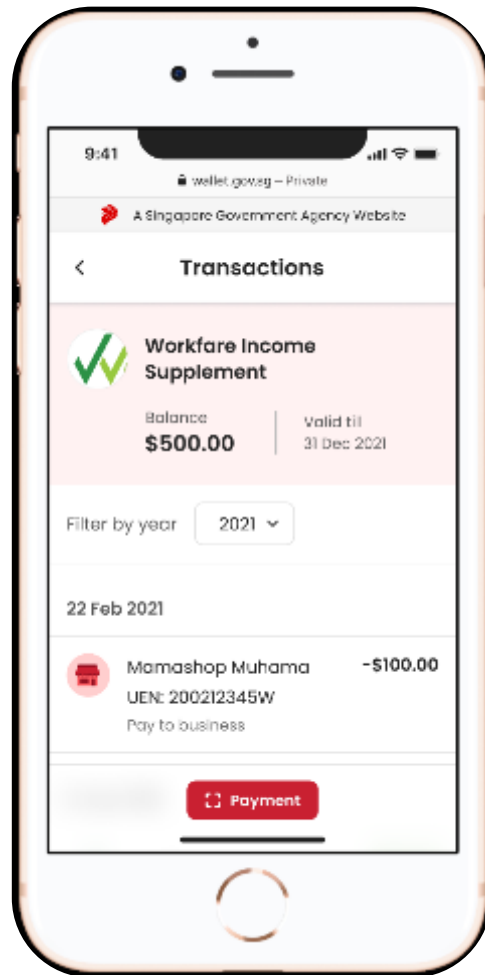


*Images are for illustration purposes only

Giving reminders on unspent credits

 Gabriel

Gabriel can now remind the citizens about unspent credits before they expire.



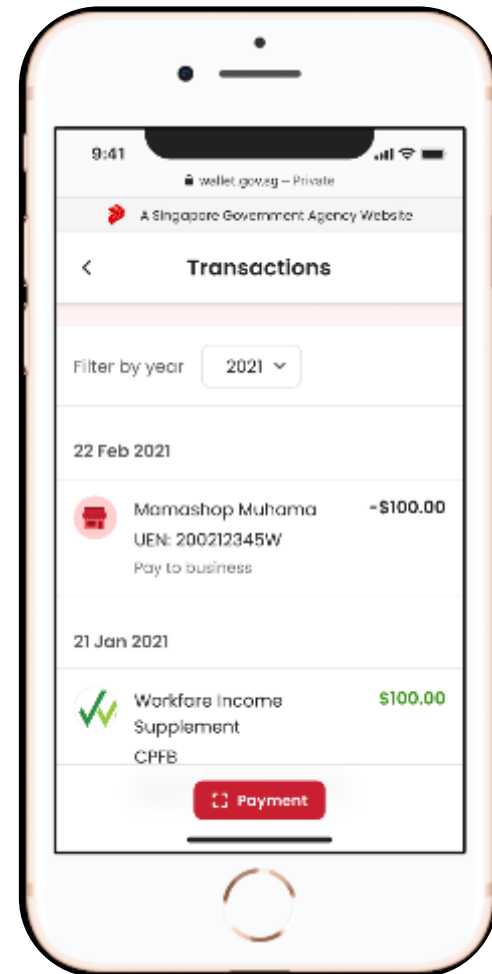
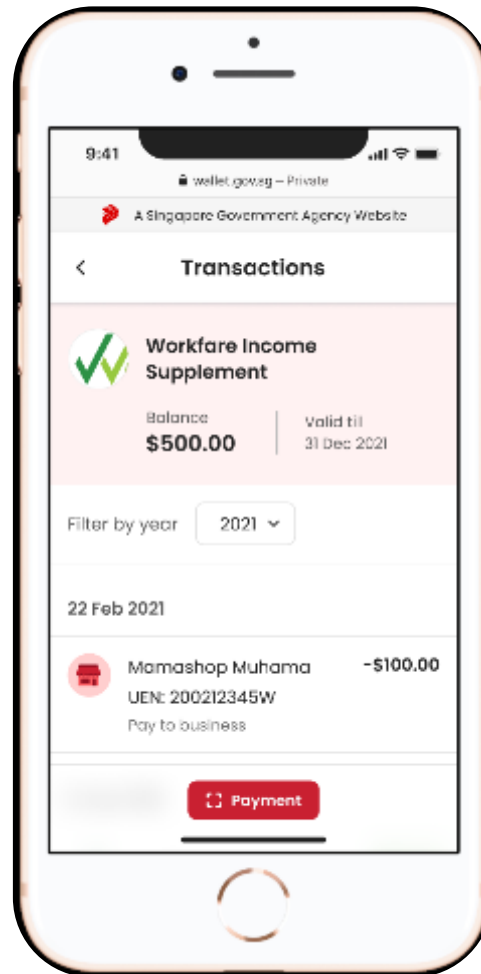
*Images are for illustration purposes only

Getting reminders & viewing spending history



Sue gets a reminder if she have any unspent credits yet.

She can also view her past spending transactions to account for her usage of the credits thus far.



*Images are for illustration purposes only

Simple merchant onboarding

Gabriel

As most shops will accept either PayNow or NETS, Gabriel can spend much lesser time to convince merchants to onboard the programme.



Simple merchant onboarding



Sue no longer needs to bring out paper vouchers.

To spend her credits, Sue just needs to look for places that can accept either PayNow or NETS payment methods, just like how she would use her other e-wallets, e.g. GrabPay.

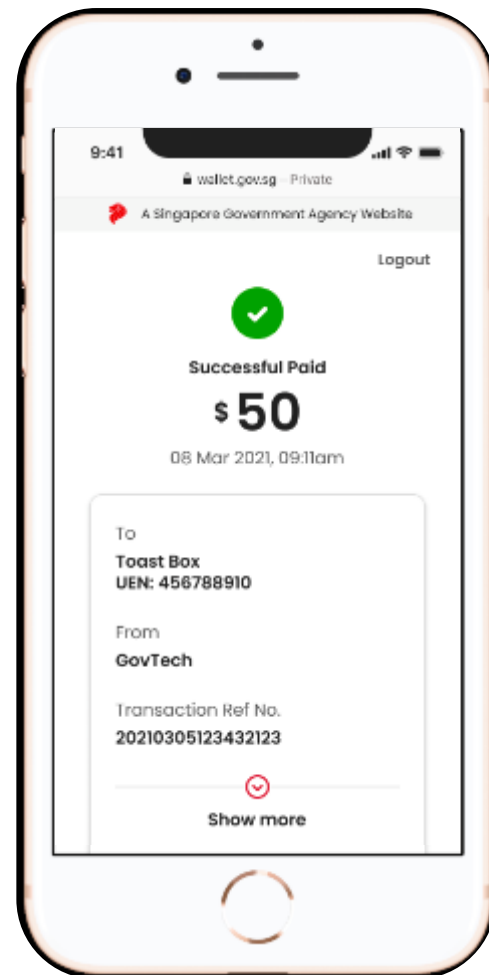


Transfer of cash at real-time

 **Gabriel**

Gabriel no longer needs to arrange for collection of vouchers from merchants for reimbursement.

With GovWallet, his agency's bank funds gets transferred to the merchant with every successful scan-and-pay.



*Images are for illustration purposes only

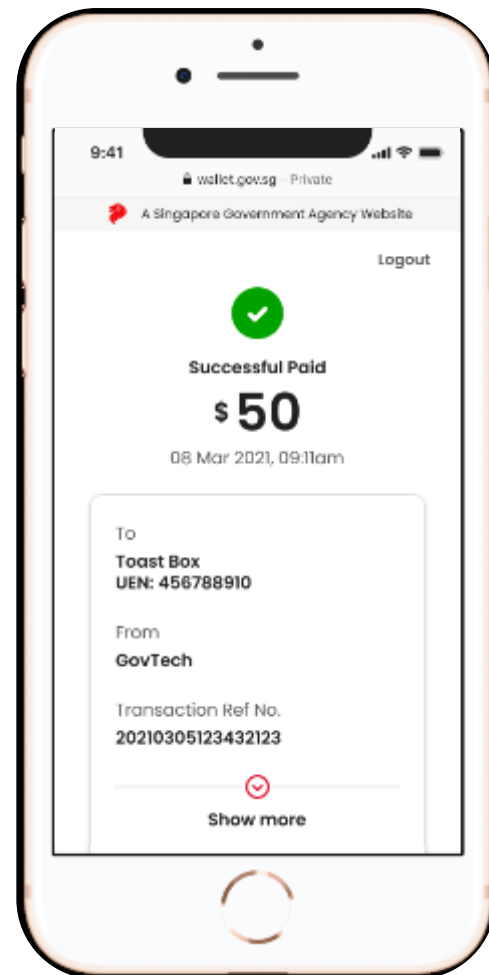
Transfer of cash at real-time

Auntie Annie

There's no need for her to wait for cash reimbursement.

Auntie Annie will now get cash for every successful scan-and-pay.

Depending on the payment method, she will either receive cash in real-time, or T+1 day.



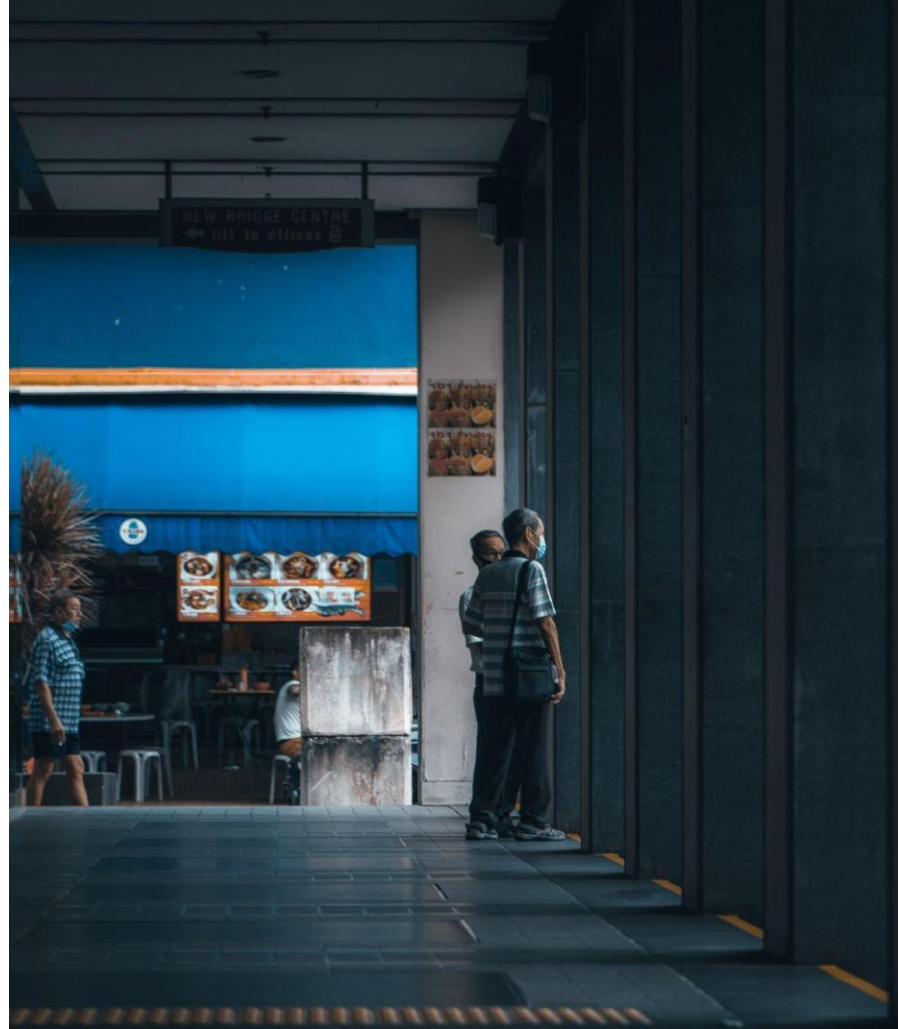
*Images are for illustration purposes only

Selection of specific merchants to spend at

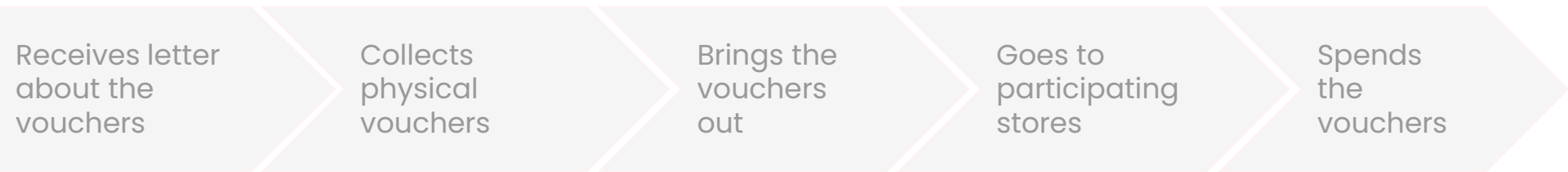
Gabriel

Gabriel's agency would like to decide the sector of merchants the payouts would ultimately help.

His agency can tap on GovWallet's ringfencing feature to specifically allow a narrowed list of merchants that would benefit from this scheme.



Sue's New Journey



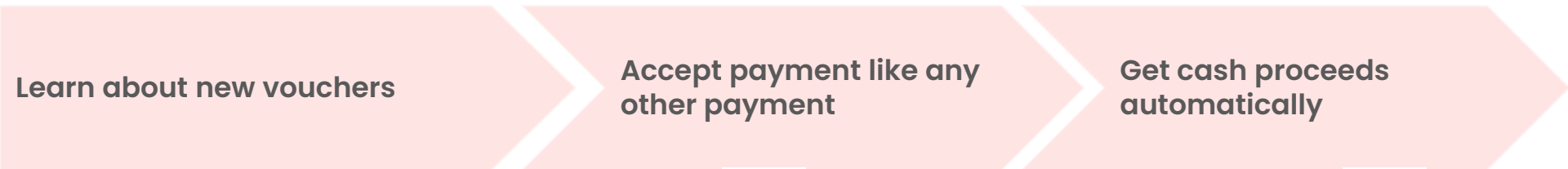
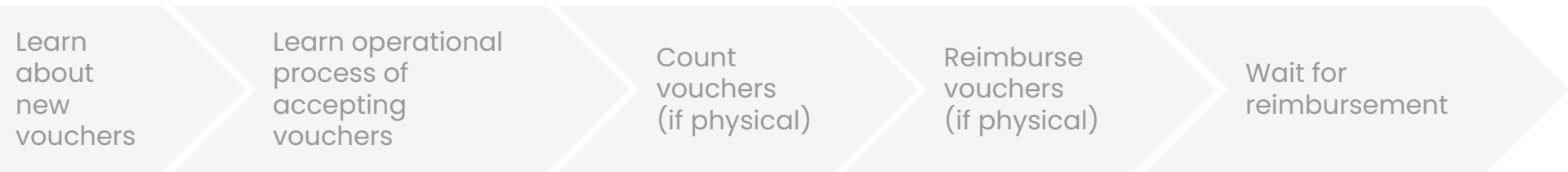


Gabriel's New Journey





Auntie Annie's New Journey



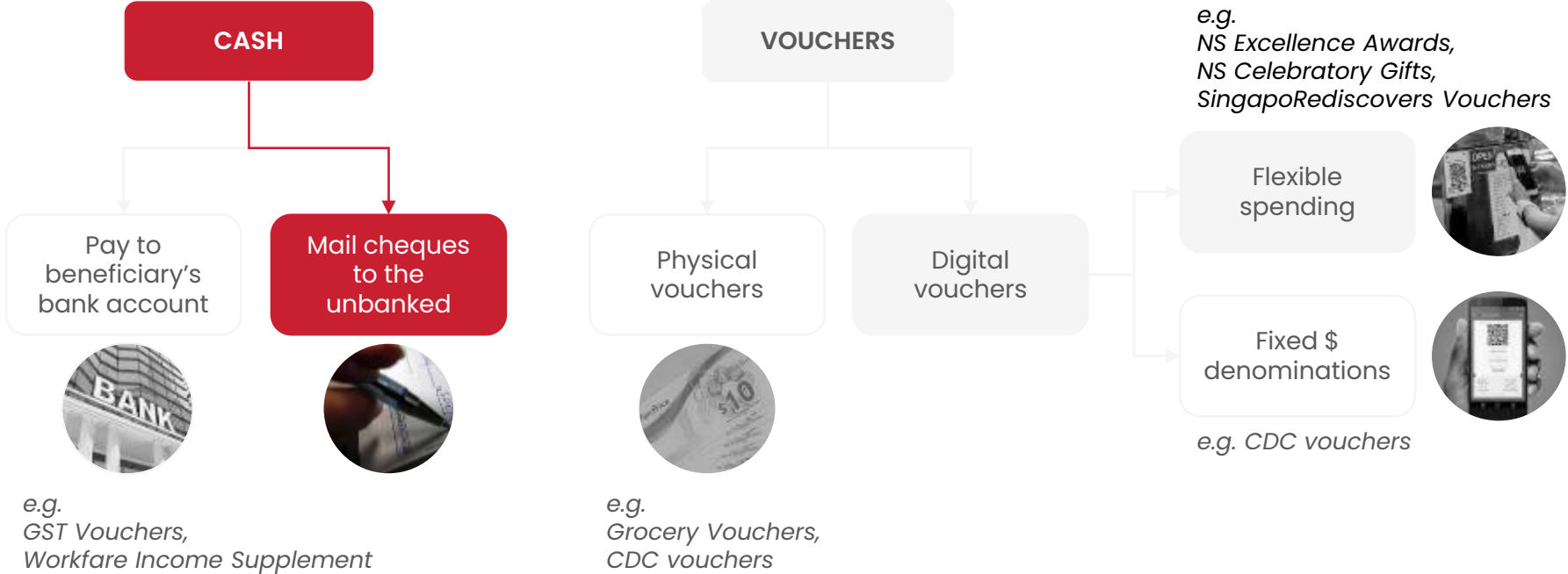
Video demo of scan-and-pay



Photo by Koh Mui Fong/TODAY



Another method besides mailing cheques?



**What are some challenges faced
with cheque payouts?**

Costly high-touch services

Challenge #1

There are millions of outbound cheques from the Government yearly.

The process of issuing cheques requires high-touch services that are costly & resource intensive, making this practice unsustainable.



Barriers in opening a bank account

Challenge #2

Citizens may face difficulty in the creation of a bank account due to the various reasons, such as

- Not comfortable with sharing bank account details with the government
- Unable to maintain the minimum average monthly value of \$500/month
- Classified as high-risk individuals by the banks



Photo by Debby Hudson on Unsplash

The need for a different approach

— — —

As the banks move away from cheque issuance and cashing-out services, there's a need to find an alternative payment method for the unbanked.

Singapore to cut use of cash, aiming to become cheque-free by 2025: Ong Ye Kung



The use of cash and cheques has been coming down steadily in recent years, said Minister for Education and Monetary Authority of Singapore board member Ong Ye Kung. PHOTO: ST FILE

Siow Li Sen

PUBLISHED 20 JUN 2018, 8:45 PM SGT

**How might we help agencies to
continue disbursing payouts
to the unbanked?**

GovCash in GovWallet

Allow users to
withdraw cash from ATMs
without an ATM card

A collaboration between
CPF, OCBC & GovTech



Photo by TODAY Online



Alice

Savvy auntie

Alice is a savvy auntie that likes to use Whatsapp to catch up with her family and friends.

Due to reasons, she does not have a bank account. Whenever she receives a cheque payout from the government, she will bring it to the bank to cash it out.

For certain technical issues, she would seek help from the youngsters.



Photo by Chua Zi Hui on Unsplash



Caregiver to his mum

Being the only child, Bob is the main caregiver for his mum who currently stays in a nursing home due to a need for 24/7 caregiving assistance.

He will take the cheque and cash it out based on the authorized 3rd party method.



Photo by Debby Hudson on Unsplash

3 ways to withdraw payouts from ATMs

— — —

1. Users who can self-help at the ATM
2. Users who may require assistance
3. Users who require a 3rd party assistance

Note: OCBC have 2 types of ATMs, the normal ATMs and the smart ATMs. Only smart ATMs have the ability to ask for assistance.



Alice's Journey in self-service withdrawal

Visits an OCBC ATM

Authenticate
using 3 steps

Select scheme to
withdraw from

Gets cash from
the ATM

1. NRIC
2. SingPass Face Verification
3. 8-digit value as 2FA



Alice's Journey with assistance

Visits an
OCBC
smart ATM

Calls for
OCBC Digital
Ambassadors

DA guides user
through 3 step
authentication

Select
scheme to
withdraw
from

Gets cash
from the
ATM

1. NRIC
2. SingPass Face Verification
3. 8-digit value as 2FA

Bob's Journey for 3rd party assistance



1. NRIC
2. 8-digit value as 2FA

Video demo of ATM flow



Technical deep dives

Overview of technical deep dives



GovWallet Tech Stack

GovWallet Tech Stack

Backend (a subset)



Configs & Infra



We're practicing



API Gateway



Amazon Lambda



DynamoDB

Languages: TypeScript, Python

Technical deep dives

Disbursements of payouts



Dalson Tan

Associate Software Engineer
Government Digital Services, GovTech

Agencies we're working with



Central
Provident
Fund Board



MINDEF
SINGAPORE



Health
Promotion
Board



Types of disbursement methods

Immediate

Given immediately, this is not a common method chosen, as most schemes are enabled at 12 midnight.

There's additional complexity that have to be handled to process a large amount of payouts in a just-in-time manner.

Scheduled

Common method when we integrate with various government agencies.

Allows both teams to properly process and schedule a large batch of payouts ahead of time, and rely on the system to activate it when the payouts are usable.

**How does
scheduled disbursements
work on GovWallet?**


Let's issue payouts of \$100 as an example!

 Tom



 Mary



 **Tom**
Balance: \$100

 **Mary**
Balance: \$150

4th Feb

Disbursement of credits

Tom and Mary are eligible for \$100 each, but only in effect after 4th June (4 months later)

Feb to June

Credits not in effect yet, still on hold

Tom and Mary's credits are not in effect yet. Balance remains the same

4th June


Credits now in effect, balance incremented

Credits are now valid. We start to process all scheduled disbursements, & balance will now be incremented

June onwards

Credits can be used!

With balance incremented, both Tom and Mary can now use their credits!

 **Tom**
Balance: \$100

 **Mary**
Balance: \$150

4th Feb

Disbursement of credits

Tom and Mary are eligible for \$100 each, but only in effect after 4th June (4 months later)

Feb to June

Credits not in effect yet, still on hold

Tom and Mary's credits are not in effect yet. Balance remains the same

4th June

Credits now in effect, balance incremented

Credits are now valid. We start to process all scheduled disbursements, & balance will now be incremented

June onwards

Credits can be used!

With balance incremented, both Tom and Mary can now use their credits!

 Tom
Balance: \$100

 Tom
Balance: **\$200**

 Mary
Balance: \$150

 Mary
Balance: **\$250**

4th Feb

Disbursement of credits

Tom and Mary are eligible for \$100 each, but only in effect after 4th June (4 months later)

Feb to June

Credits not in effect yet, still on hold

Tom and Mary's credits are not in effect yet. Balance remains the same

4th June


Credits now in effect, balance incremented

Credits are now valid. We start to process all scheduled disbursements, & balance will now be incremented

June onwards

Credits can be used!

With balance incremented, both Tom and Mary can now use their credits!

 **Tom**
Balance: \$200

 **Mary**
Balance: \$250

4th Feb

Disbursement of credits

Tom and Mary are eligible for \$100 each, but only in effect after 4th June (4 months later)

Feb to June

Credits not in effect yet, still on hold

Tom and Mary's credits are not in effect yet. Balance remains the same

4th June

Credits now in effect, balance incremented

Credits are now valid. We start to process all scheduled disbursements, & balance will now be incremented

June onwards

Credits can be used!

With balance incremented, both Tom and Mary can now use their credits!

**What are some challenges
working with scheduled
disbursements?**

Pain points of scheduled disbursements

1

Future Date

2

Large Volume

3

Instantaneous Effect



Pain points of scheduled disbursements

1

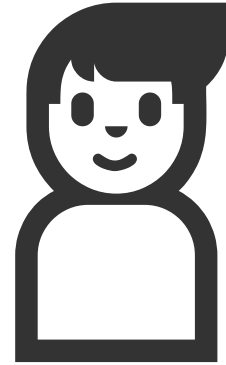
Future Date

2

Large Volume

3

Instantaneous Effect



Pain points of scheduled disbursements

1

Future Date

2

Large Volume

3

Instantaneous Effect



Lessons learnt from
processing & scaling
scheduled disbursements

Efficiency in calculating balance

Ensuring that the balance of credits is reflected correctly is a tricky part, especially when disbursements are valid from a particular point in time.

What's the best way to update the balance in the fastest way possible?

Efficiency in calculating balance

What's the best way to update the balance in the fastest way possible?

Option 1: In-depth live calculations

1. Get all valid disbursements
2. Get all past transactions
3. Sum them up to get the accurate balance

Option 2: Adhoc updates

1. User balance is computed & stored
2. Upon a new active disbursement, update user balance

Example of in-depth live calculation

Computation on 4th June

Date	Transaction / Disbursement
Valid from 1st Jan	\$100 disbursed
Valid from 1st Feb	\$100 disbursed
Transaction on 2nd Feb	\$150 used
Valid from 1st Aug	\$100 disbursed
Total	

Example of in-depth live calculation

Computation on 4th June

Date	Transaction / Disbursement
Valid from 1st Jan	\$100 disbursed ✓
Valid from 1st Feb	\$100 disbursed ✓
Transaction on 2nd Feb	\$150 used ✓
Valid from 1st Aug	\$100 disbursed ✗
Total	\$50

Efficiency in calculating balance

Option 1: In-depth live calculations

1. Get all valid disbursements
2. Get all past transactions
3. Sum them up to get the accurate balance

Works very well if the record size is small.

May still need to handle concurrency edge cases

Option 2: Adhoc updates

1. User balance is computed & stored
2. Upon a new active disbursement, update user balance

Scales better for large amount of transactions & disbursements

Need to handle concurrency edge cases

Scaling for large amounts of disbursements

We illustrated how scheduled disbursements works for a scenario where only two citizens, Tom & Mary, receive payouts.

Let's now scale that up by 50,000 times, such that we now cater for ~100,000 citizens to have their payouts valid at midnight.

Scaling for large amounts of disbursements

Goal: ~100,000 citizens' payout to be processed at midnight

What's the best way to scale this?

Split everything into buckets

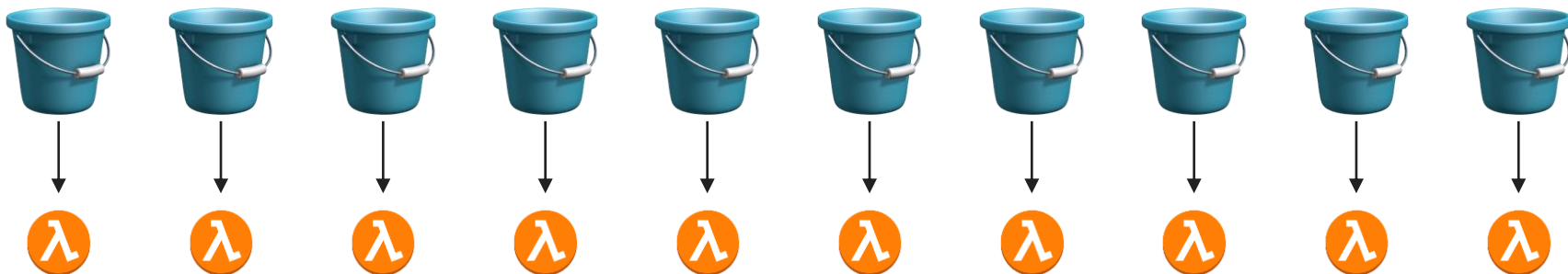


Scaling for large amounts of disbursements

Goal: ~100,000 citizens' payout to be processed at midnight

What's the best way to scale this?

Check the bucket every 1 minute & process records



Cater for errors and retry attempts

Mistakes can happen which might cause delay in the processing of the payouts.

Once everything is fixed, we want the citizens to receive their payouts as soon as possible with minimal config.

Having a retry period of 1 minute allows us to retry in the next minute and quickly process them.

Other technical approaches considered

- We considered Dynamodb's TTL feature
 - Benefits:
 - Part of database feature
 - Concerns:
 - TTL can take up to 24 hours to be removed, and that may cause quirks in terms of balance computation

Other technical approaches considered

- We considered pushing all events to an SQS queue
 - Benefits:
 - Async method
 - Can make use of DLQ
 - Concerns:
 - There's a maximum number of messages that the queue can store
 - Additional lambda calls needed to create the processing events at time of validity to push it to the queue
 - Additional infrastructure, might not be required for this use case

Technical deep dives

Expiration of payouts



Immanuella Lim

Software Engineer
Government Digital Services, GovTech

Modelling multiple government schemes

- Different government schemes have different expiration dates, & some do not expire at all
- Expiration caters for financial reconciliation & year-end closing of accounts
- A citizen can receive the same scheme twice, but with two different expiry dates

**How does credits expiry
work on GovWallet?**

Modelling for one disbursement

Assumptions made:

1. The smallest factor of time for expiries is a day
2. The only time that credits can expire is at the end of the day, i.e. at 23:59:59

 Tom

Initial Balance: \$100

Available Balance: \$90

Expires on 2022-06-03T23:59:59.999+08:00

Modelling multiple disbursements

 Tom

Initial Balance: \$100

Available Balance: \$90

Expires on **2022-06-03T23:59:59.999+08:00**

 Tom

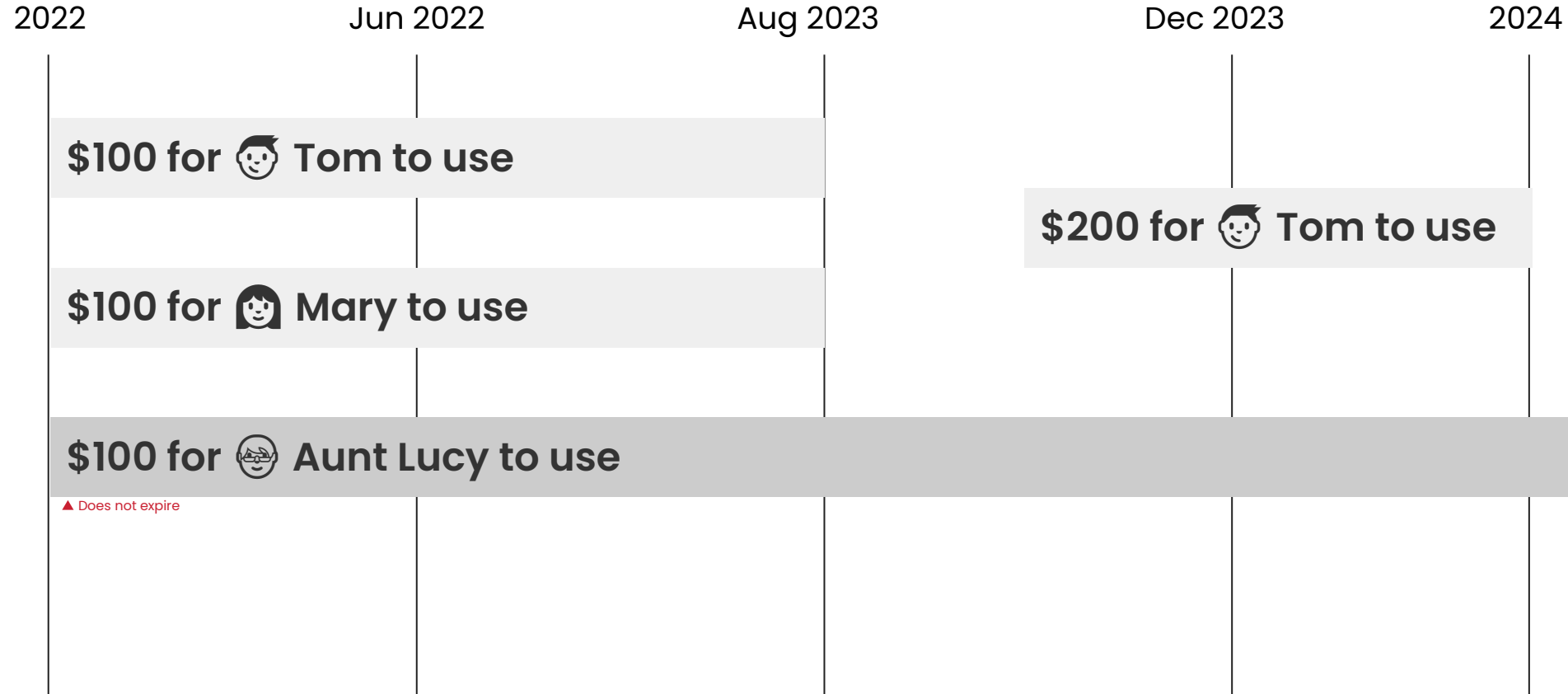
Initial Balance: \$500

Available Balance: \$500

Expires on **2023-01-31T23:59:59.999+08:00**

Total Balance for Tom: \$590

A simple scenario of credits expiry



Order of deduct/top-ups for expiry

Assumptions made:

- Spending will always deduct from the earliest expiring bucket
- Refunds will go back to the earliest expiring bucket

An example of credits spending #1

Initial Balance: \$100

Available Balance: \$90

Expires on 2022-06-03T23:59:59.999+08:00

 Tom spends \$20 on lunch

Initial Balance: \$100

Available Balance: \$70

Expires on 2022-06-03T23:59:59.999+08:00

Initial Balance: \$500

Available Balance: \$500

Expires on 2023-01-31T23:59:59.999+08:00

Initial Balance: \$500

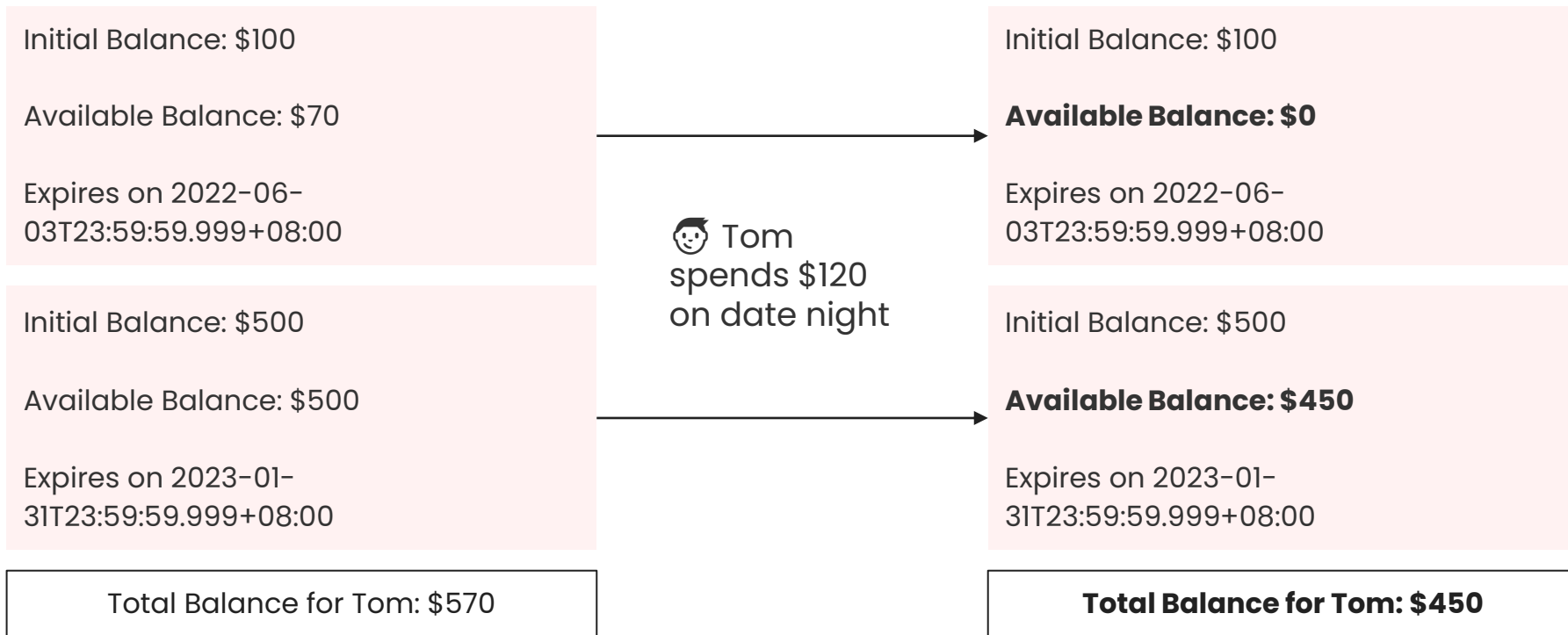
Available Balance: \$500

Expires on 2023-01-31T23:59:59.999+08:00

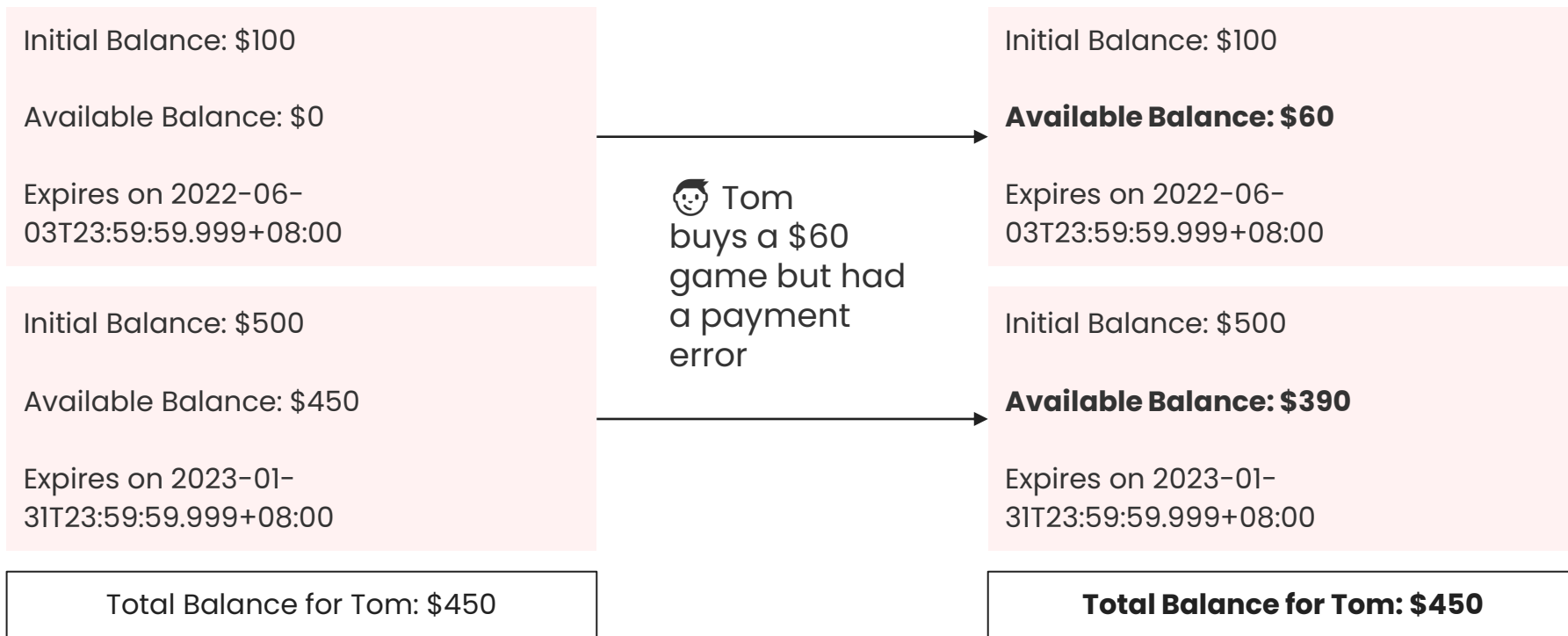
Total Balance for Tom: \$590

Total Balance for Tom: \$570

An example of credits spending #2



An example of credits refund



Lessons learnt from
modelling & implementing
expiration of credits

Plan ahead while delivering in milestones

- Forgetting to plan a few steps ahead means more complexities down the line when we work on new requirements & improvements
- Learning how to quantify & define a milestone better

It's ok to make assumptions

- Credits expiration can be very complex, and setting boundaries helps to make the problem more solvable
- Assumptions also helps guide our understanding of the feature

Balancing user needs & minimum data records

- We made a wrong assumption that viewing expired payouts is of a lower priority for agencies & end users, but this was not the case
- We prioritised keeping a minimum set of data records for fast retrieval of data
- This resulted in a more complex way of retrieval of these values, while we work on the enhancements to make retrieval more ergonomic

Technical deep dives

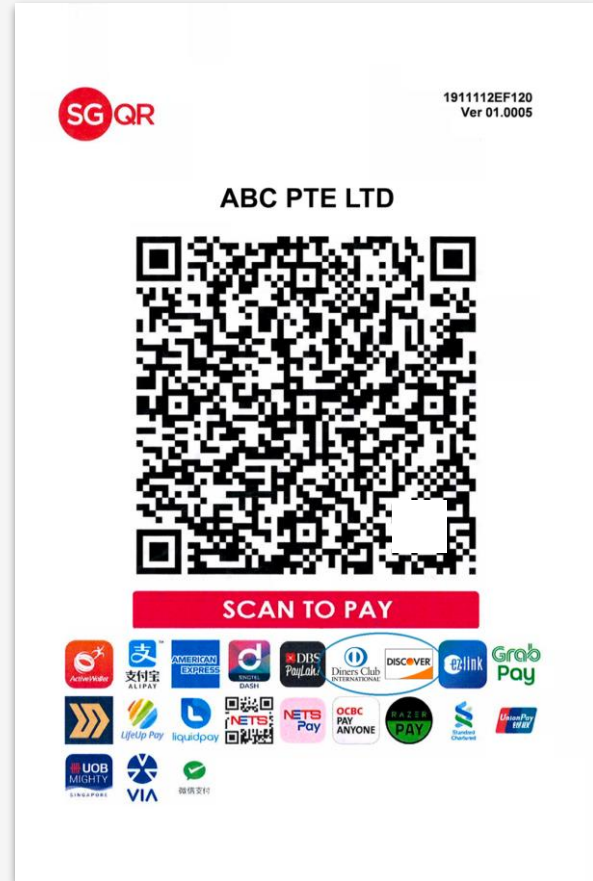
SGQR Payment Parser



Yong Jia Jie

Associate Software Engineer
Government Digital Services, GovTech

Singapore Quick Response (SGQR) Code



Why SGQR?

Make electronic payments by scanning a single QR



Reduce the need of multiple POS terminals



**What's *inside*
an SGQR?**



Payment Scheme Information

Information
necessary to make
payment to a
specific scheme.

- Examples include UEN, mobile number, reference numbers, etc.
- Each SGQR can contain multiple instances of payment scheme information.
- Consumers can choose which payment scheme they wish to pay with via their application of choice.



Merchant and Transaction Information

Information
necessary to make
payment to a
specific merchant.

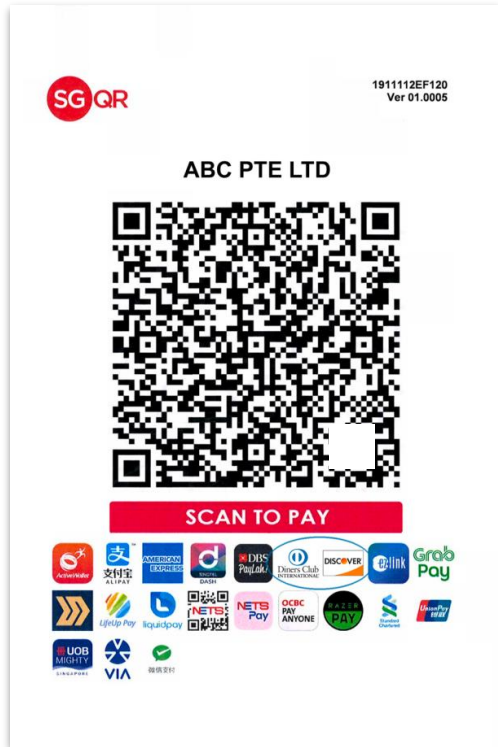
Information about a specific merchant or transaction:

- Transaction amount
- Transaction currency
- Merchant name
- Postal code
- etc.

Resolving SGQRs

**We want to
parse, validate, and transform SGQRs
to obtain their contents
and serve it to our consumers.**

Parse



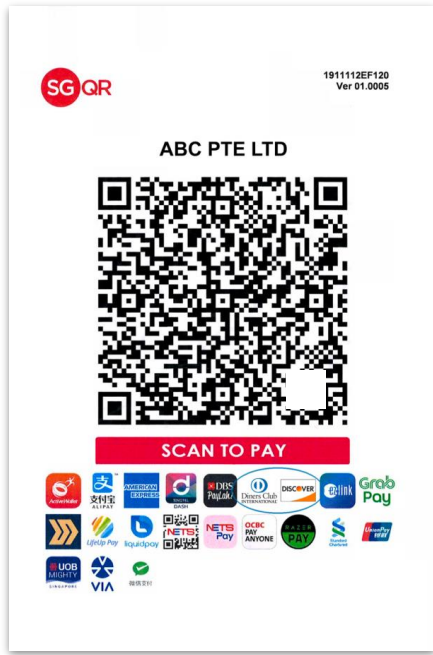
In this step, we want to resolve the SGQR into parts that we can meaningfully describe.

Pre-requisites

- SGQR Specifications
- Payment Scheme Specific SGQR Specifications

We convert the SGQR payload into JSON objects, as per the specification.

Parse



```
[  
  {  
    "id": 0,  
    "length": 2,  
    "value": "01"  
  },  
  {  
    "id": 1,  
    "length": 2,  
    "value": "11"  
  },  
  ...  
]
```

```
{  
  "schemes": [  
    "paynow": ...,  
    "nets": ...,  
  ]  
  "amount": 10.00,  
  "merchantName": "Stack-X",  
  ...  
  "checksum": "ABCD"  
}
```


Validate

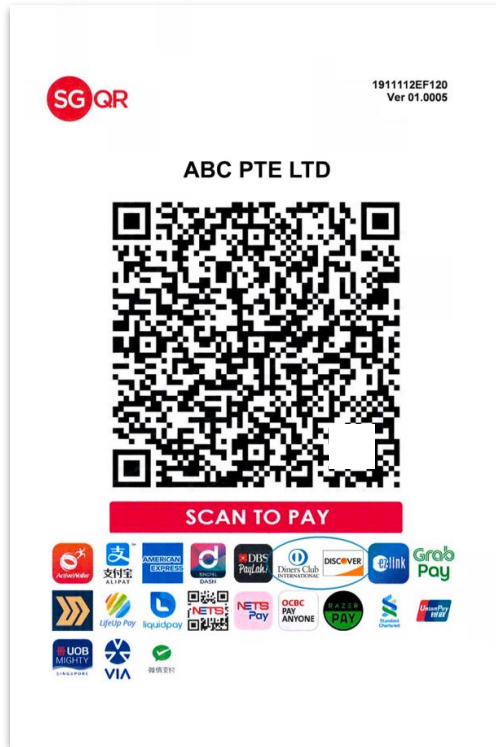
JSON Schemas

- JSON objects that defines the structure of JSON data.
- Clear, human- and machine-readable documentation.
- Helps us validate information as per specifications.

Functions

- Functions with custom validation logic.
- Helps us validate information that cannot be covered by JSON schemas.
- Examples include expiry dates, and SGQR checksum.

Transform

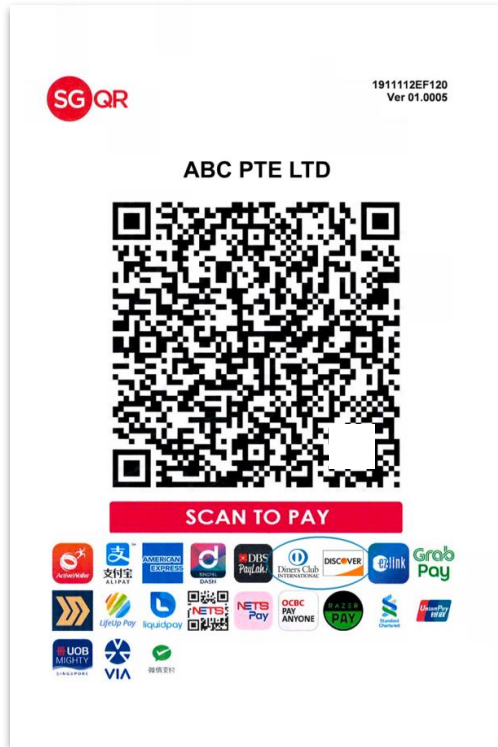


In this step, we want to alter the contents of the SGQR as per our opinions.

Opinions

- Transaction amounts are represented in cents.
- All truthy and falsy values shall be represented in boolean.
- Implicit defaults shall be present explicitly.

Eligibility



In this step, we want to determine if the SGQR is eligible for payment for a particular campaign.

This is an extra step on top of resolving SGQRs.

Eligibility Conditions

- Type of transaction (e.g. C2B, C2C, etc.)
- Merchant
- Payment schemes

Using SGQRs

Using SGQRs



Display transaction information



Facilitate checks



Facilitate transaction changes



Used for payment API calls

Lessons Learnt

Lessons learnt



Trust the specifications, but **verify** on real world data.



Always consider the **customer**.



Log as much as you can to facilitate **troubleshooting**.

Summary

It's been an action-packed ride so far, and we'll love to share more about our learnings as we continue building GovWallet.

We hope that our efforts so far have helped to make the experience better for everyone 😊.

If you have any questions, feel free to drop us a note at govwallet_support@tech.gov.sg

P.S. We're also hiring!

Question & Answer Time

Patricia

Product Owner

Sebastian

Tech Lead

Imma

Software Engineer

Jia Jie

Associate Software

Engineer

Dalson

Associate Software

Engineer

...and the rest of the team who is here too!

Thanks for joining us!
Have a good night!

