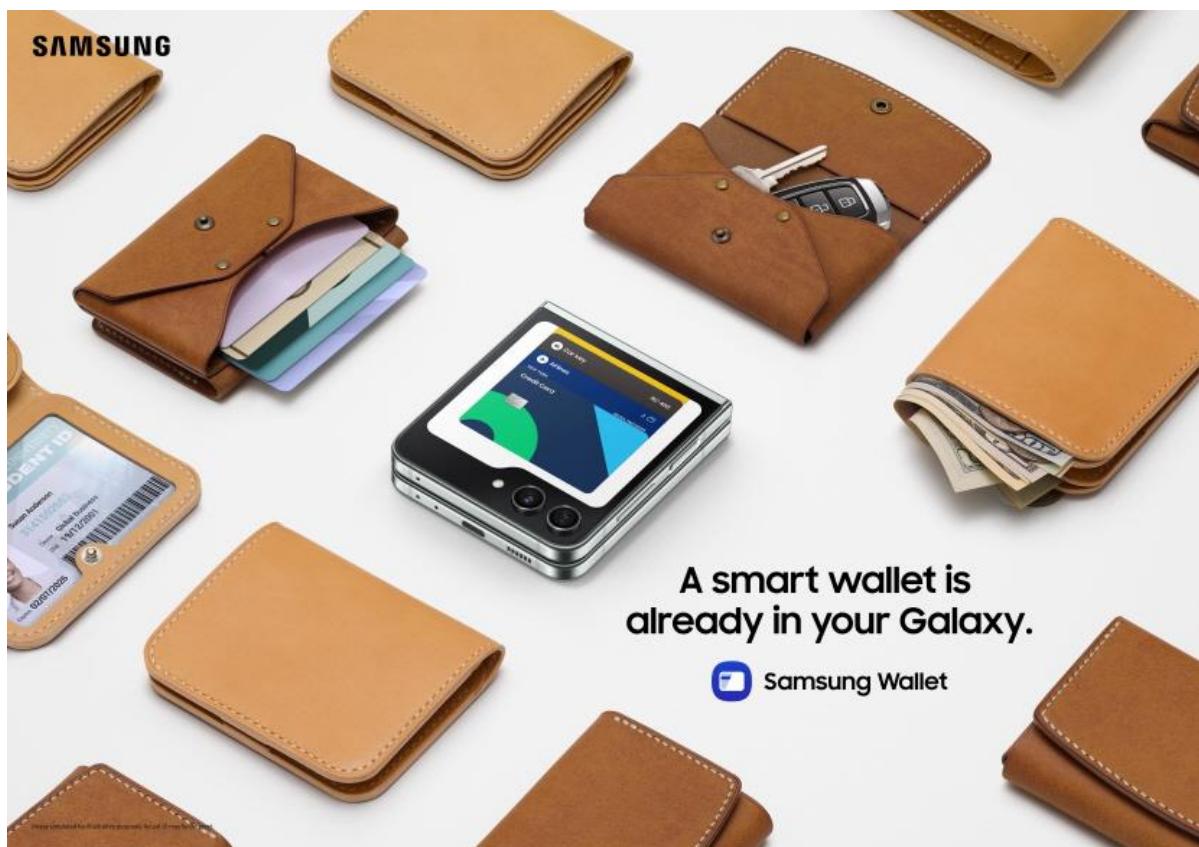


SAMSUNG Wallet

Samsung Wallet Cards API Specification

Wallet R&D Group
Samsung Electronics



A smart wallet is
already in your Galaxy.

 Samsung Wallet

Table of Contents

Table of Contents.....	2
Revision History	5
Preface.....	7
1. Overview.....	8
1.1 Getting started	8
Account.....	8
Onboarding.....	8
Add To Wallet.....	9
Management.....	9
1.2 Service flow	10
Preparation.....	10
Add to wallet	11
Update wallet cards.....	11
1.3 Card States	12
State diagram.....	12
2. Add to Samsung Wallet.....	13
2.1 Service flow	13
2.2 Interfaces.....	14
Data Transmit Link	14
Data Fetch Link	15
3. Wallet Cards.....	16
3.1 Boarding Pass.....	17
3.2 Event Ticket.....	28
3.3 Coupon.....	38
3.4 Gift Card	42
3.5 Loyalty	46
3.6 ID Card	50
3.7 Generic Card.....	57

4. Implementing ATW button.....	62
4.1 Samsung Wallet on the Web.....	62
Web Button Reference with importing API Javascript.....	62
4.2 Samsung Wallet on the App.....	66
App Button Reference.....	66
4.3 Samsung Wallet on the MMS/Email.....	68
Link to 'Add to Wallet' on the MMS/Email.....	68
4.4 Statistics Service	69
5. Server interaction	71
5.1 Partner Server API.....	72
Get Card Data	72
Send Card State.....	75
5.2 Samsung Server API	77
Update Notification.....	77
Cancel Notification	80
6. Security.....	82
6.1 Card Data Token.....	82
JWT Details.....	82
6.2 Authorization Token.....	85
JWT Details.....	85
7. Image Resources	88
7.1 Button Image resources	88
Add to Samsung Wallet Button	88
8. References.....	91
8.1 Barcodes.....	91
8.2 Locations.....	93
Details.....	93
8.3 Additional information	94
Details.....	94
8.4 Links.....	96

Samsung Wallet Cards – API Specification

Details.....	96
8.5 Classification.....	98
Details.....	98
8.6 Check service available devices.....	99
8.7 Security factors.....	101
Service Flow	101
How to generate key factors.....	102
8.8 Contact points	104

Revision History

Version	Description	Date
1.0.0	Released.	June 2022
1.0.2	New card type, 'Digital Key' is defined Some API path resource 'card' have been replaced by 'cards'. (Still supports backward compatibility.)	July 2022
1.0.7	Defined new reference chapter 'Actions'.	July 2022
1.0.10	Minor Updates	October 2022
1.0.12	The 'csinfo' has been updated from Required to optional. The 'Gift Card' has been added to 2.3 Attributes.	November 2022
1.0.13	A new chapter 'Wallet Cards' has been added. 'Event Ticket' has been replaced with 'Ticket'. Descriptions on some attributes have been modified. More attributes have been defined on 'Boarding Pass' and 'Event Ticket'.	November 2022
1.0.14	Card Sub-type defined Samsung Server API paths have been modified. 'Send Card State' response payload has been added. Wallet Cards details has been updated.	March 2023
1.1.0	logoImage.lightUrl has been added to 3. Wallet Cards > Boarding Pass, Event Ticket, Coupon, Gift Card, Loyalty. 3.4 Gift Card and 3.5 Loyalty have been added.	April 2023
1.1.1	Attributes has been added/changed	May 2023
1.1.2	Defined a new wallet card type 'ID Card' 1.3 Card States > State diagram updated 5.1 Partner Server API updated 5.2 Samsung Server API updated	July 2023
1.1.3	'ID Card' attributes has been updated 'Event Ticket' has been updated	July 2023
1.1.4	Expressions have been corrected Sample data and UI have been added	August 2023
1.1.5	New attributes has been updated Guide description has been updated	September 2023
1.1.6	Guide description has been updated 'ID Card' attributes has been updated 'Event Ticket' has been updated Chapter titles and positions has been adjusted	November 2023

Samsung Wallet Cards – API Specification

Version	Description	Date
1.1.7	'ID Card' attributes has been update 'Generic Card' attributes has been updated 'Links' chapter has been defined 'token' has been defined in 'Authorization Token'	December 2023
1.2.0	'Generic Card' attributes has been updated 'privacyModeYn' has been defined as a common attribute	January 2024
1.2.1	'Generic Card' decription and attributes has been updated Matters related to Samsung Pay branding has been excluded	February 2024

Preface

Copyright Notice

Copyright © 2021 Samsung Electronics Co. Ltd. All rights reserved. Samsung is a registered trademark of Samsung Electronics Co. Ltd. Specifications and designs are subject to change without notice.

Though every care has been taken to ensure the accuracy of this document, Samsung Electronics Co, Ltd. cannot accept responsibility for any errors or omissions or for any loss occurred to any person, whether legal or natural, from acting, or refraining from action, as a result of the information contained herein. Information in this document is subject to change at any time without obligation to notify any person of such changes.

Samsung Electronics Co, Ltd. may have patents or patent pending applications, trademarks copyrights or other intellectual property rights covering subject matter in this document. The furnishing of this document does not give the recipient or reader any license to these patents, trademarks copyrights or other intellectual property rights.

No part of this document may be communicated, distributed, reproduced or transmitted in any form or by any means, electronic or mechanical or otherwise, for any purpose, without the prior written permission of Samsung Electronics Co, Ltd.

All brand names and product names mentioned in this document are trademarks or registered trademarks of their respective owners.

Contact Information

Samsung Electronics Co., Ltd
416, Maetan-3dong, Yeongtong-gu Suwon-City, Gyeonggi-do, 443-742
Wallet R&D Group
Mobile Communications Business
Contact: jinwan2.choi@samsung.com

1. Overview

Samsung Wallet Cards is an e-wallet service that allows customers using Samsung devices to clip various digitized contents on Samsung Wallet. Users can add their ticket, coupon, boarding pass and other types of data into Samsung Wallet using an 'Add to Wallet' link via multiple online channels like App, Web, E-mail, or Social media messages.

This document describes how to implement Samsung Wallet Cards Service features from integration partners' point of view.

1.1 Getting started

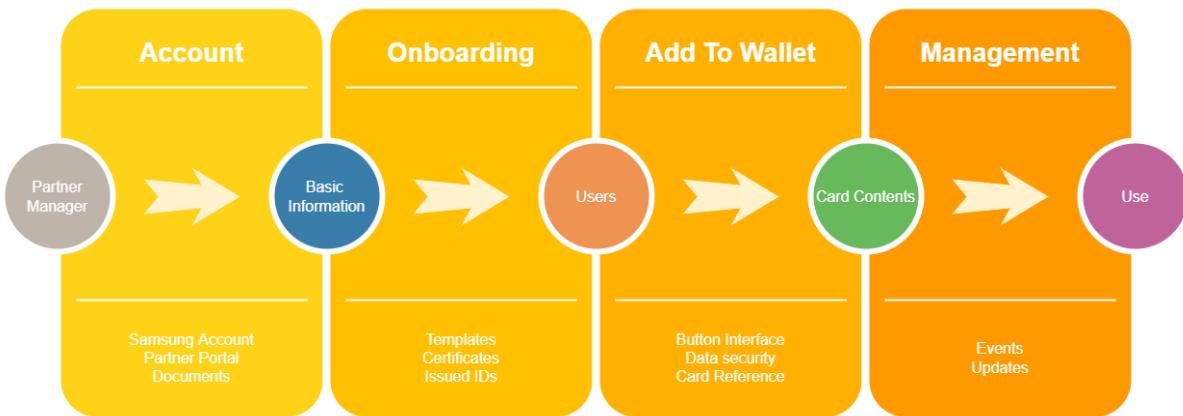


Figure 1-1 Integration Steps

Account

Samsung Account is necessary to manage your own service for Samsung Wallet. The account is used to sign-up/sign-in on Partner portal.

Onboarding

The interaction keys are issued from a service onboarding step. The partner manager should fill basic information of the company and card details on partner portal, choose the card type and template that fit their contents, which will then provide keys such as Card ID and Partner ID. To ensure safe data communication, make sure to prepare security key factors. Samsung will sign the provided CSR then deliver certificates used to create the secured data.

See the document '*Samsung Wallet Partner Onboarding Guide*' for onboarding details.

See the chapter [Security](#) for more details.

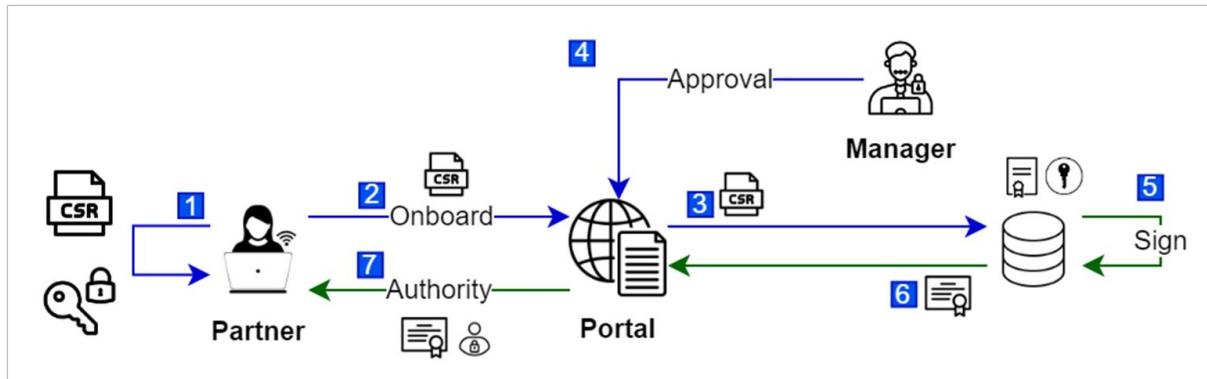


Figure 1-2 Service Onboarding

Add To Wallet

Partner managers have to configure 'Add to Samsung Wallet' button on their contents. The button must configure the link to the promised web page with the specific format. The linked page will initiate a basic process of storing cards on Samsung Wallet service.

See the chapter [Implement the button](#) for more details.

Every digitized contents on 'Add-to-wallet' link is identified by a reference ID (refId). If the same reference ID is reused, the existing card will be updated with new data without notice.

See the chapter ['Add to Wallet' Interfaces](#) for more details.

Samsung Wallet app won't cache every resource. No-cache web contents must be used for interfaces such as CDN. In other words, the app won't consider work with '304 Not Modified'.

Management

The stored wallet card contents on Samsung Wallet can be updated to predefined states. If partners want to manage the contents on Samsung Wallet, interaction between Samsung and partner servers is necessary. See the chapter [Server interaction](#) for more details.

1.2 Service flow

This chapter describes how to bring users' digitized content to Samsung Wallet. Once partner service onboarding is complete and templates are ready, it is possible to add contents such as boarding passes, coupons and tickets into Samsung Wallet. See the service flow below.

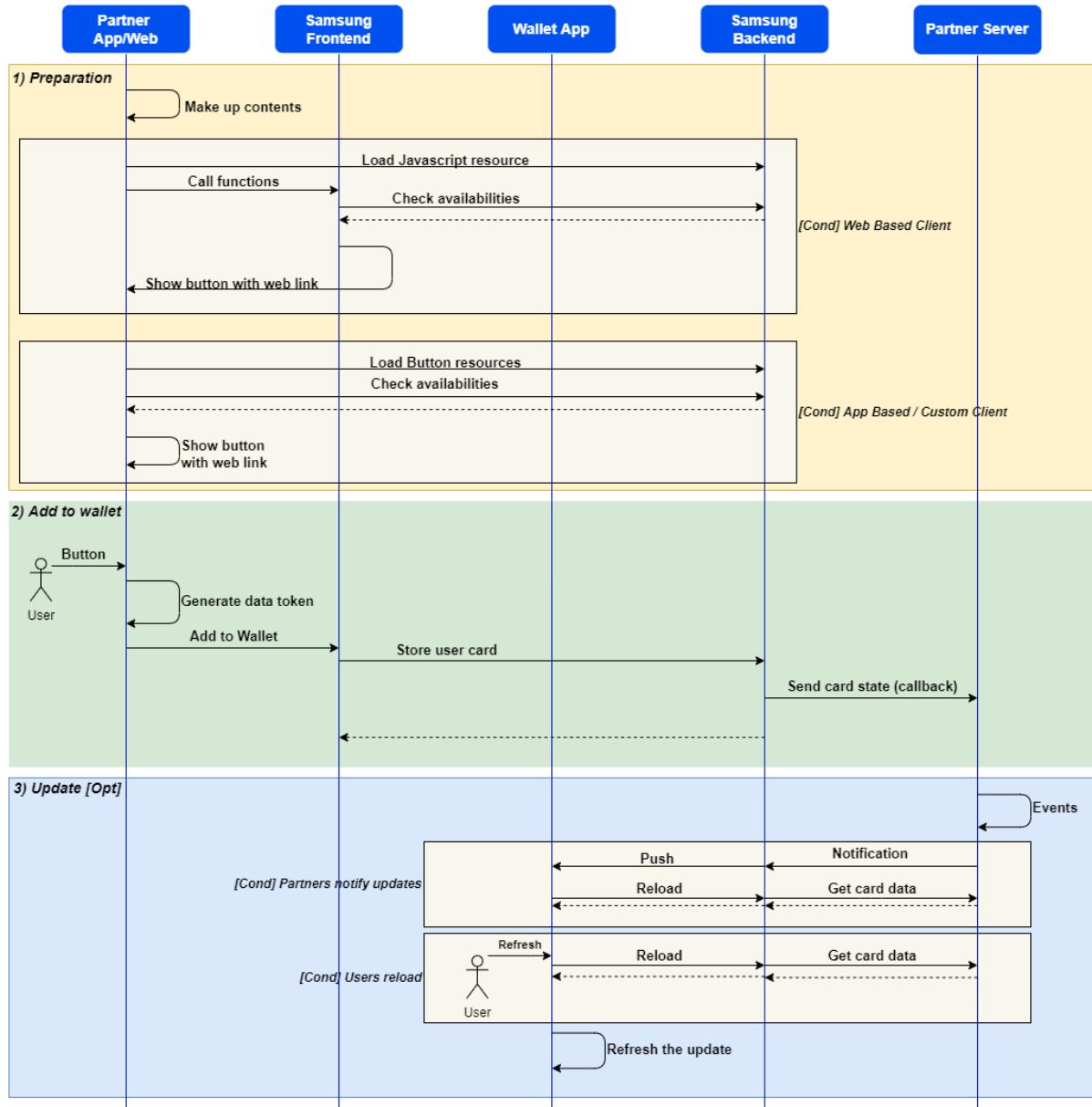


Figure 1-3 General Service Flow

Preparation

- The contents intended to store on Samsung Wallet must be tokenized as JWT (JSON Web Token)

when using [Data Transmit Link](#).

- Considering what environments the button will be displayed in such as App, Web, MMS or Email when you plan to integrate this service.
- See the chapter [Implement the button](#) for more details.

Add to wallet

- Once users who want to store contents click or tab the linked button, it initiates the process of adding to Samsung Wallet. See '[Add to Wallet' Interfaces](#) for more details.
- Partners can consider the option [Data Fetch Link](#) when they provide full content data.
- Samsung will notify the result through server interface named [Send Card State](#). This works as a callback for 'Add to Wallet' process delivering events and country code for the user wallet card.
- For interactions after storing contents on Samsung Wallet, server interfaces are needed which is defined on the chapter [Server interaction](#).

Update wallet cards

- If the wallet card contents should be updated by partners, they must call Samsung server API.
- Path parameter 'cc2' must be set to the value brought from callback named 'Send Card State' API which includes 'ADDED' event. See a chapter [Samsung Server API](#) for more details.
- [Update Notification](#) is for single content.
- [Cancel Notification](#) is for calling off every user's wallet card contents related to a specific event.
- [Get Card Data](#) payload has the same format of [card information](#) as the one of 'Add to wallet'.
- [Send Card State](#) can also have an event 'DELETED' to notify a wallet card deletion on Samsung Wallet by users.

1.3 Card States

Wallet cards have several states for each content provider's notifications and user actions. The state diagram below describes Wallet cards' states from the beginning to the end.

State diagram

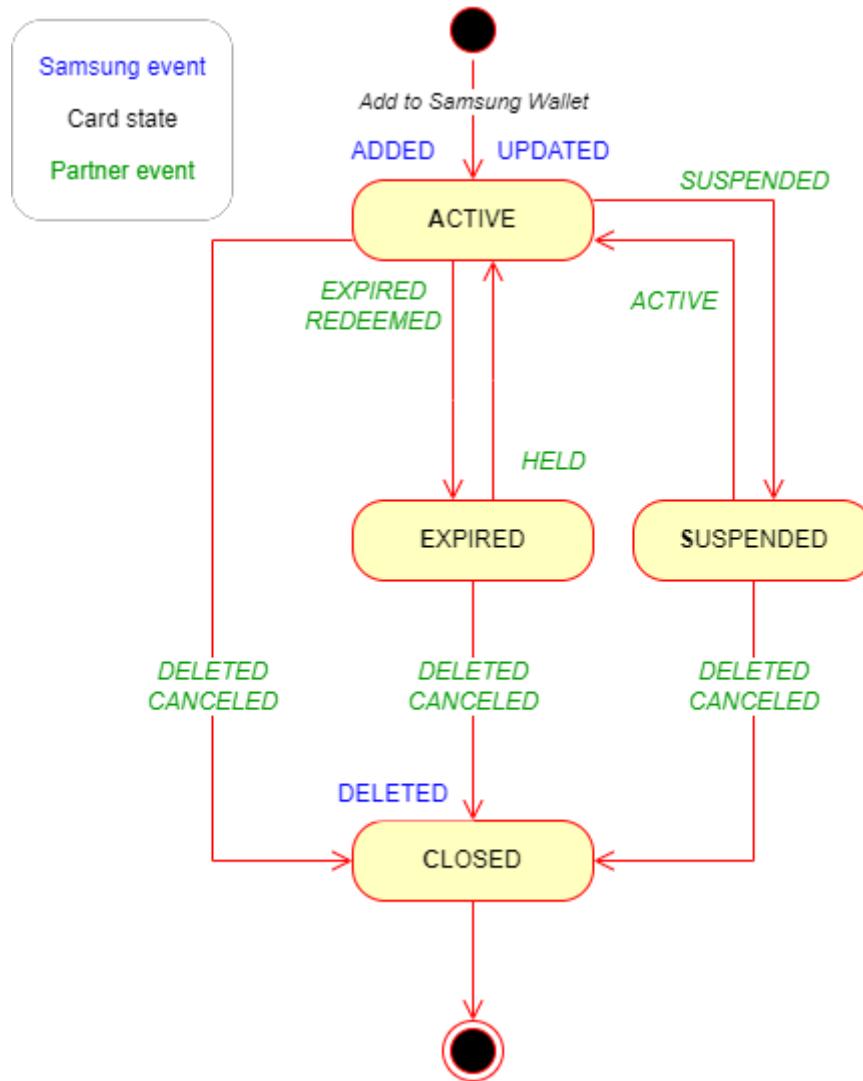


Figure 1-4 Card state

2. Add to Samsung Wallet

'Add to Samsung Wallet' service also known as ATW, defines interfaces for users to conveniently add digital contents to Samsung Wallet.

2.1 Service flow

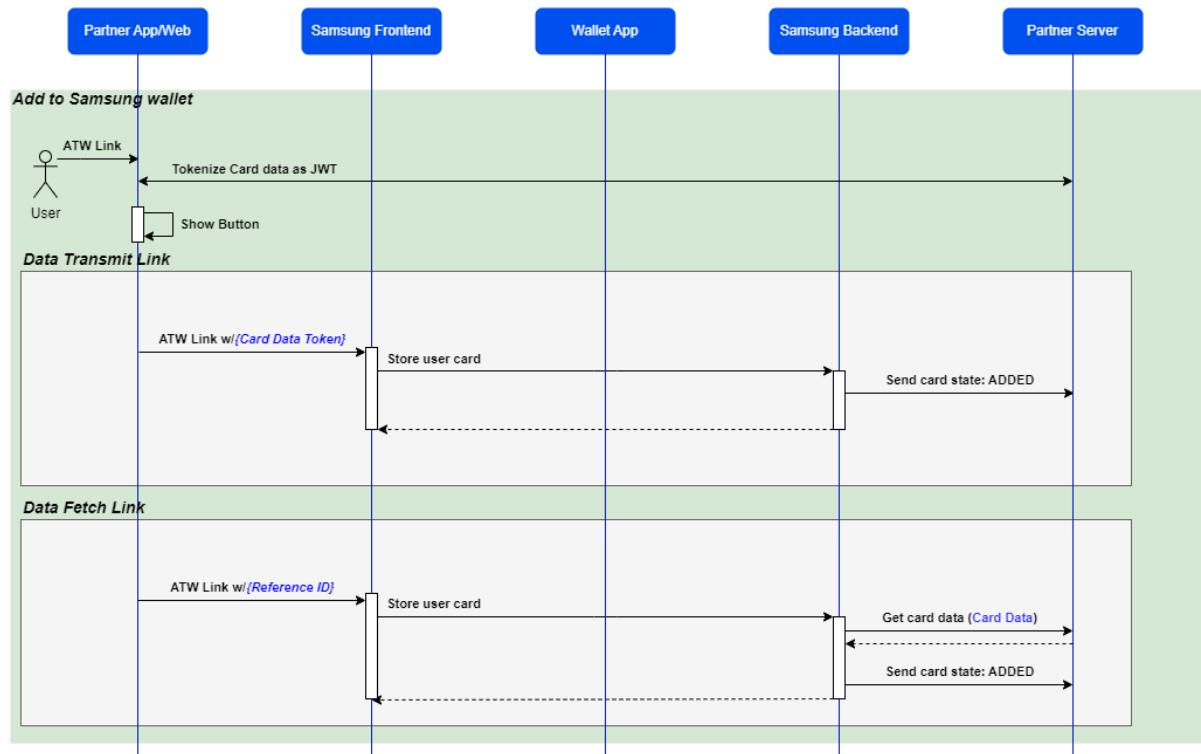


Figure 2-1 Add to Samsung Wallet

2.2 Interfaces

Data Transmit Link

The most common and straightforward method is the Data Transmit Link approach, which securely includes tokenized data in the ATW link. The ATW link format for this method is as follows.

* The name *Data Transmit Link* has been changed from *Typical flow*.

Type	Value			Description
URL	https://a.swallet.link/atw/v1/{Card Id}#Clip?cdata={Card Data Token}			
Path Parameters	Card Id	String(32)	Required	Wallet card identifier issued from Partner portal when the partner manager signs up for partner services and registers the wallet card they want to service.
Hash Path Parameters	#Clip	String(5)	Required	Parameters for the Hash link * The first letter is capitalized
Query Parameters	cdata	Object	Required	Actual payload data in basic JSON format to communicate between partners and Samsung Wallet. This must be secured in JWT(JSON Web Token) format. * See the chapter Security for more details.
Example	https://a.swallet.link/atw/v1/1656147182764415319#Clip?cdata=eyJjdHkiOiJKV1QiLCJhbGciOiJsInRpBWWzdGFtcCI6ImNyZWF0ZWQgdGltZSIzInBhcnRuZXJJRCI6InBhcnRuZXIgSUQifQ. Dn0_oZ3xcr0JuQ3mlSzLIUTxFoTewnZ0MQj7kiNjysNm5Xfwqt5vcN20PeebeLgUx8VJXLy4_9G4BHQ-hd4O9POYuTuAWew.YzdIMTFh0 - NYCeL3T0YzNzAD2KcK_HrtwIGEErHLGn6ydaq_fpFdSlxsA3ZJtNpg3wcuqEw5cldpbPFswbQLropqEpNawg5nlm3DKAA4a1daZMbSR1BGZHrH_vIKnx3CY5MO0jNBexl_YIZ5_wB379UYSwumQiPiTZVg2ljYvfht17i4			

Data Fetch Link

In cases involving sensitive data or when providing static links, Data Fetch Link method is highly recommended. Links using this approach include only a unique reference ID, and Wallet Cards are added by querying data through [Get Card Data](#) path as specified in Partner portal.

* The name *Data Fetch Link* has been changed from *Slim Data Flow*.

- Please be aware that if the link is exposed to unintended users, it can be exploited. Please prepare the integration with this in mind.
- It is crucial to ensure that the refId, used for a reference value, is generated in a manner that is not easily deducible by potential attackers.

Type	Value			Description
URL	https://a.swallet.link/atw/v1/{Card Id}#Clip?pdata={Reference ID}			
Path Parameters	Card Id	String	Required	Wallet card identifier * Must be generated from Partner Portal
Hash Path Parameters	#Clip	String	Required	Parameters for the Hash link.
Query Parameter	pdata	String	Required	Unique ID defined by content providers. This has identification for each user's Wallet Card contents. * For secure transactions, a Reference ID(refId) must be in a form that cannot be inferred.
Example	* Example: web link https://a.swallet.link/atw/1656147182764415319#Clip?pdata=slgHCzIwM9g			

3. Wallet Cards

This chapter defines Wallet Card data fields for attributes object by each Wallet card type.

The structure for configuring wallet cards follows the defined specification. Configuring the card data in the specified formatted JSON structure is required. See the details for each card type.

Type	Value			Description
Card object	card	Object	Required	Card information.
	card.type	String(16)	Required	Wallet Card type
	card.subType	String(16)	Required	Wallet Card sub type
	card.data[]	Array of Object	Required	Wallet card data container Allows up to 6 objects at once
	data[].refId	String(32)	Required	A unique content identifier defined by the content provider
	data[].createdAt	Long(13)	Required	Data creation timestamp. Epoch timestamp in milliseconds. *UTC±00:00
	data[].updatedAt	Long(13)	Required	Data update timestamp. Epoch timestamp in milliseconds. *UTC±00:00
	data[].language	String(8)	Required	Default content language code. e.g., en, ko
	data[].attributes	Object	Required	Attributes of card data. *Refer to the following chapters for each type
	data[].attributes.{fields}			Attribute fields by card type.
	data[].localization[]	Array of Object	Optional	Information for multilingual support.
	localization[].language	String(8)	Required	Multilingual content language code. e.g., en, ko
	localization[].attributes.{fields}			For displaying a given language, 'data[].attributes' can be replaced by localized versions. *Refer to the following chapters for each type

<p>Example</p> <pre>* Example: Card object { "card": { "type": "ticket", "subType": "movies", "data": [{ "refId": "ref-20230304-001", "createdAt": 1612660039000, "language": "en", "attributes": { "title": "Samsung Wallet", "mainImg": "https://..../main.png" } }], "localization": [{ "language": "ko", "attributes": { "title": "삼성 월렛" } }] } }</pre>	<p>* Example: Card object</p> <pre>{ "card": { "type": "ticket", "subType": "movies", "data": [{ "refId": "ref-20230304-001", "createdAt": 1612660039000, "language": "en", "attributes": { "title": "Samsung Wallet", "mainImg": "https://..../main.png" } }], "localization": [{ "language": "ko", "attributes": { "title": "삼성 월렛" } }] } }</pre> <p style="text-align: center;">*Refer to the following chapters for each type</p>
--	--

To ensure secure transmission of card data, it must be tokenized in JWT format. For this purpose, you will require the certificate obtained using the partner's email account when signing up for the partner portal.

For detailed information on secure data tokenization, please refer to the [Security](#) chapter.

*Image resources provided by URLs can be cached. Therefore, in order for the image resource to be replaced immediately, the corresponding URL path must be changed.

3.1 Boarding Pass

'Boarding pass' cards support one-off transit types such as *airlines*, *trains*, *buses*, *ferries*, and *etc.*

Boarding passes in Wallet can remind the user of boarding time and help during check-in. Depending on the integration level, it can provide status information updates related to flights, and journeys, such as boarding gate changes or cancellations.

Wallet Card Type	Wallet Card Sub Type
<i>boardingpass</i>	<i>airlines</i> , <i>trains</i> , <i>buses</i> , <i>others</i>

Samsung Wallet Cards – API Specification

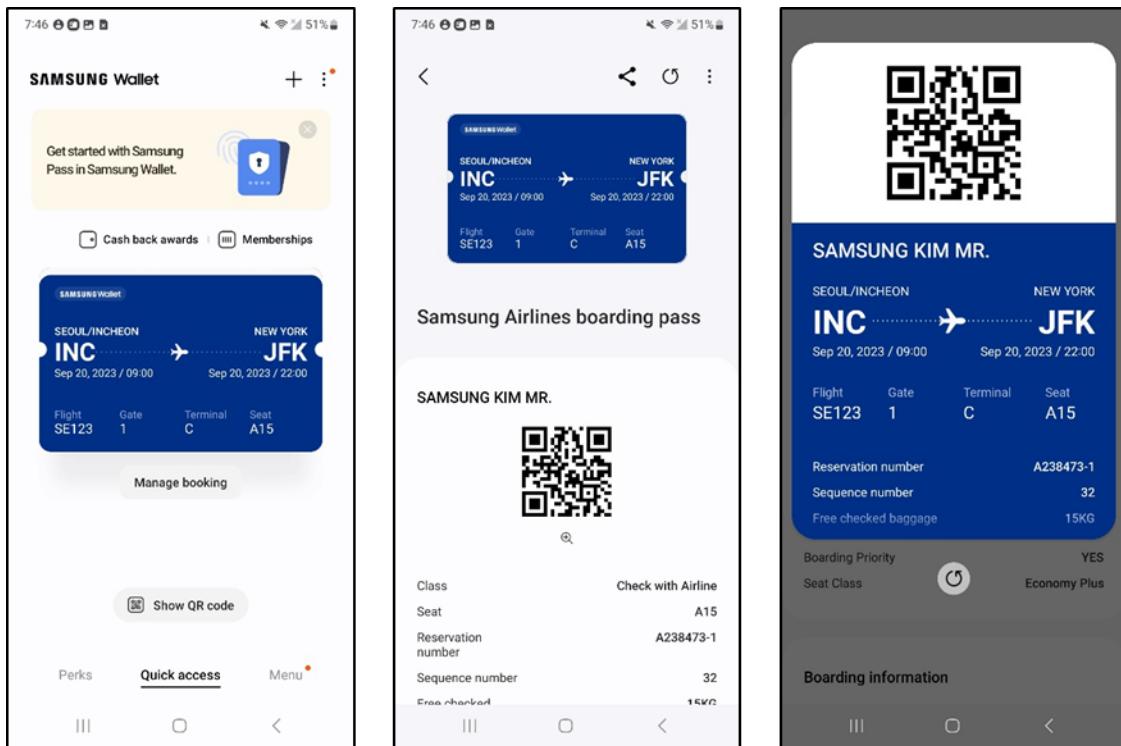


Figure 3-1 Sample: Boarding Pass: Airlines

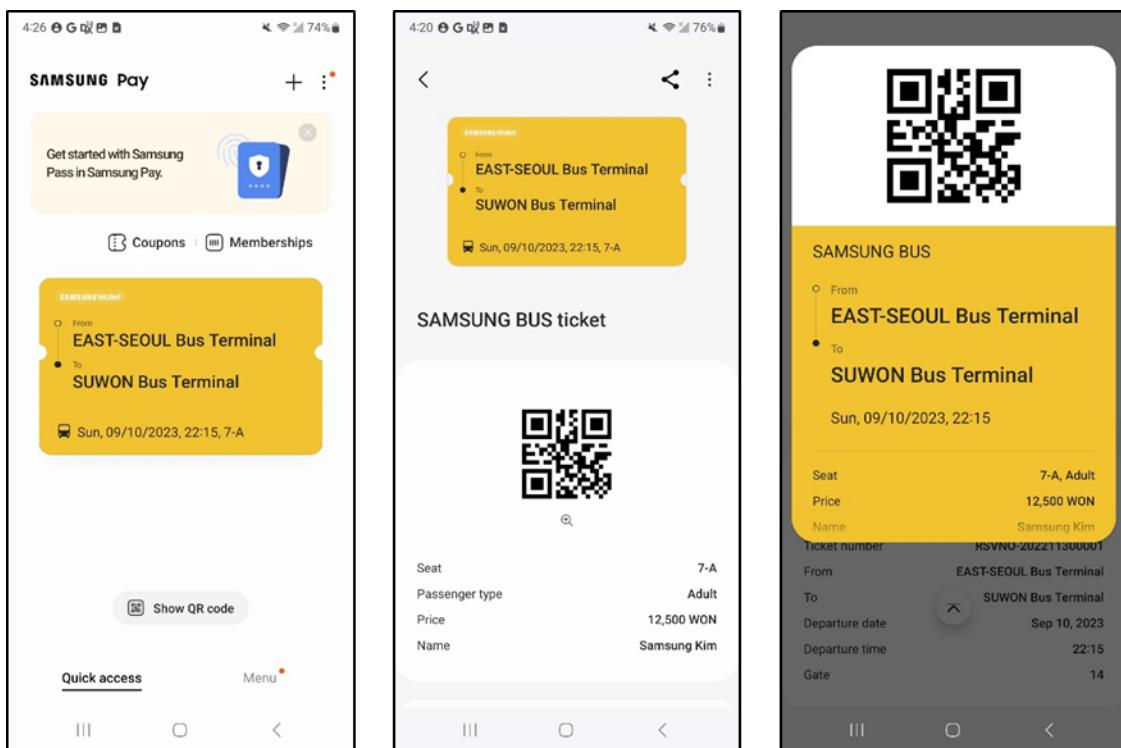


Figure 3-2 Sample: Boarding Pass: Buses

Samsung Wallet Cards – API Specification

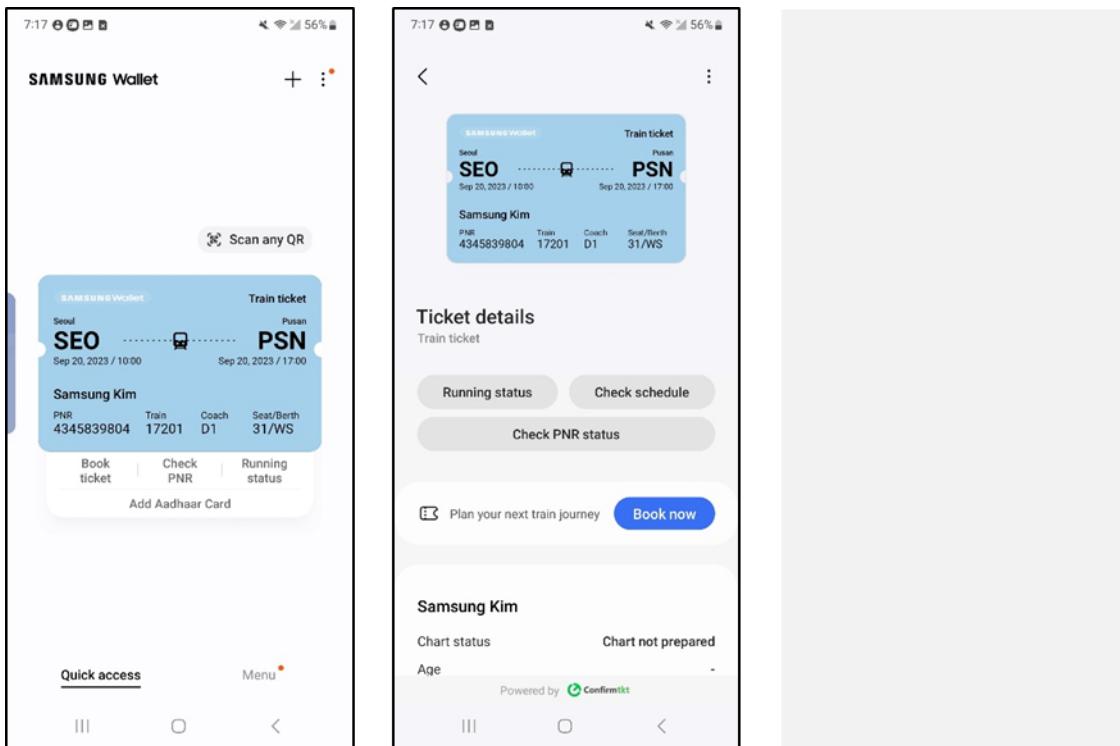


Figure 3-3 Sample: Boarding Pass: Trains

Specification: Boarding Pass

Type	Value			Description
attributes.{fields}	title	String(32)	Required	Main title. e.g., 'BOARDING PASS'.
	transitType	String(16)	Optional	Transit type * This will be deprecated
	groupingId	String(32)	Optional	Identifier used to group related cards.
	orderId	String(32)	Optional	Unique identifier for an order.
	providerLogo	String(256)	Required	URL of the logo image. The file size should not exceed than 256 kB.
	providerLogo.darkUrl	String(256)	Required	URL of the logo image in dark mode. The file size should not exceed 256 kB.

	providerLogo.lightUrl	String(256)	Required	URL of the logo image in light mode. The file size should not exceed 256 kB.
	providerName	String(32)	Required	Display name of the boarding pass provider.
	user	String(64)	Conditional	Passenger Name * Required if subType is <i>airlines</i>
	userType	String(32)	Optional	Passenger type i.e., Classification of passengers like <i>Child, Adult</i>
	transitOperator	String(64)	Optional	Operator name
	vehicleNumber	String(32)	Conditional	Transit or route number i.e., Flight number on airlines * Required if subType is <i>airlines</i>
	transitClass	String(32)	Optional	Transit class i.e., <i>Premium</i> class for buses
	transitFare	String(32)	Optional	General fare or purchase fare
	coachNumber	String(16)	Optional	Identifier assigned to each individual carriage. e.g., B4
	seatClass	String(32)	Conditional	Seat class. * Required if subType is <i>airlines</i>
	seatNumber	String(16)	Required	Individual number of the passenger seat. e.g., A-9, Free
	seatFacing	String(16)	Optional	Direction in which the seat is oriented within a carriage. It indicates whether the seat is positioned to face the direction of travel or face the opposite direction. e.g., Forward, Backward
	reservationNumber	String(32)	Required	Number of the transit reservation.
	boardingPriority	String(32)	Optional	Information on whether entitles the passenger to board before others. i.e., if the passengers are <i>pregnant</i> .
	boardingSeqNo	String(32)	Optional	Boarding sequence number
	boardingGroup	String(8)	Optional	Value of boarding group or zone. e.g., B.

	boardingTime	Long(13)	Optional	Boarding time. Epoch timestamp in milliseconds.
	boardingTime.utcOffset	String(8)	Conditional	UTC offset of boarding time at the departure point. * Required if boardingTime exist
	gateClosingTime	Long(13)	Optional	Boarding gate closing time. Epoch timestamp in milliseconds.
	gateClosingTime.utcOffset	String(8)	Conditional	UTC offset of gate closing time at the departure point. * Required if gateClosingTime exist
	baggageAllowance	String(16)	Optional	Baggage allowance.
	departName	String(32)	Conditional	Name of the departure point. * Required if subType is <i>airlines</i>
	departCode	String(8)	Conditional	Code for the departure point. * Required if subType is <i>airlines</i>
	departTerminal	String(8)	Conditional	Terminal name of the departure point. * Required if subType is <i>airlines</i>
	departGate	String(8)	Optional	Gate name of the departure point.
	estimatedOrActualStartDate	Long(13)	Conditional	Departure time. Epoch timestamp in milliseconds. i.e., the estimated time the aircraft plans to pull from the gate, or the actual time the aircraft already pulled from the gate. * Required if subType is <i>airlines</i>
	estimatedOrActualStartDate.utcOffset	String(8)	Conditional	UTC offset of time at the departure point. * Required if estimatedOrActualStartDate exist
	arriveName	String(32)	Conditional	Name of the arrival point. * Required if subType is <i>airlines</i>
	arriveCode	String(8)	Conditional	Code for the arrival point. * Required if subType is <i>airlines</i>
	arriveTerminal	String(8)	Optional	Terminal name of the arrival point.
	arriveGate	String(8)	Optional	Gate name of the arrival point.

	estimatedOrActualEndDate	Long(13)	Conditional	Arrival time Epoch timestamp in milliseconds. i.e., the estimated time the aircraft plans to reach the destination gate (not the runway), or the actual time it reached the gate. * Required if subType is <i>airlines</i>
	estimatedOrActualEndDate.utcOffset	String(8)	Conditional	UTC offset of time at the arrival point. * Required if estimatedOrActualEndDate exist
	locations	String(1024)	Optional	List of locations where the card can be used. * See Location format.
	bgColor	String(8)	Required	Color of the card art. e.g., #00FFFF.
	fontColor	String(8)	Optional	Color of the font on the card art Acceptable values: <i>dark, light</i>
	blinkColor	String(8)	Optional	Color of the blinking effect in the Indicator area. e.g., #00FFFF.
	appLinkLogo	String(256)	Required	App link image URL. The file size should not exceed 256 kB.
	appLinkName	String(32)	Required	App link name.
	appLinkData	String(256)	Required	Information about the partner app link.
	extraInfo	String(512)	Optional	Additional information to be delivered to customers. * See Additional Information format.
	noticeDesc	String(1024)	Optional	Text of the notice. * Long content is allowed. * See Additional Information format.
	csInfo	String(512)	Optional	Set value to JSON Object (<i>call</i> or <i>email</i> or <i>website</i>) * See below for an example.
	displayTSAPreCheckYn	String(1)	Optional	Flag whether to display the TSA PreCheck image. Either 'Y' or 'N'. Image size: 68*20 / 140*20

	membershipStatus Level	String(256)	Optional	Image URL for the status level of the airline alliance or own membership. i.e., Airline: SkyTeam, Star Alliance. Image size: 68*20 / 140*20
	barcode.value	String(4096)	Optional	Barcode data, serial number
	barcode.serialType	String(32)	Optional	Presentation Type e.g., SERIALNUMBER, BARCODE, ... * See Barcode format
	barcode.ptFormat	String(32)	Optional	Presentation Format e.g., BARCODE, QRCODE, SERIAL, ... * See Barcode format
	barcode.ptSubFor mat	String(32)	Optional	Presentation Sub-format e.g., CODE_128, QR_CODE, ... * See Barcode format
	barcode.errorCorre ctionLevel	String(4)	Optional	Amount of redundancy or error correction data included in the code. There are four error correction levels available in QR codes. - Code options: L/M/Q/H

Example	<u>Airline boarding pass</u> <pre>{ "card": { "type": "boardingpass", "subType": "airlines", "data": [{ "refId": "SE16138353212584800001", "createdAt": 1612660039000, "updatedAt": 1612660039000, "language": "en", "attributes": { "title": "OO AIR BOARDING PASS", "transitType": "Airline", "groupId": "SE867132687321", "providerLogo": "https://..../logoImage.png", "providerLogo.darkUrl": "https://..../logoImage.png", "providerName": "OO AIR", "user": "GIL DONG HONG", "vehicleNumber": "SE123", "seatClass": "Economy Plus", "seatNumber": "A15", "reservationNumber": "A238473-1", "boardingSeqNo": "32", "boardingTime": 1612660039000, "boardingTime.utcOffset": "UTC+09:00", "baggageAllowance": "15KG", "departName": "SEOUL/INCHEON", "departCode": "INC", "departTerminal": "C", "departGate": "1", "estimatedOrActualStartDate": 1612660039000, "estimatedOrActualStartDate.utcOffset": "UTC+09:00", "arriveName": "SAN FRANCISCO", "arriveCode": "SFO", "arriveTerminal": "A", "arriveGate": "11", "estimatedOrActualEndDate": 1612660039000, "estimatedOrActualEndDate.utcOffset": "UTC-08:00", "bgColor": "#FF00FF", "fontColor": "dark", "appLinkLogo": "https://..../applinklogo.png", "appLinkName": "OO Airline", "appLinkData": "https://www.ooairline.com", "csInfo": "{\"call\": \"555) 123-4567\", \"email\": \"cs@email.com\", \"website\": \"https://homepage.com/cs\"}", "barcode.value": "CS16138353212584806754FG1802", "barcode.serialType": "BARCODE", "barcode.ptFormat": "QRCODESERIAL", "barcode.ptSubFormat": "QR_CODE" }, "localization": [</pre>
---------	---

```
{
    "language": "ko",
    "attributes": {
        "title": "OO 항공 탑승권",
        "providerName": "OO 항공",
        "user": "홍길동"
    }
}
]
}
]
}
}

Grouped Bus boarding passes
{
    "card": {
        "type": "boardingpass",
        "subType": "buses",
        "data": [
            {
                "refId": "ref-202211300001",
                "createdAt": 1669782394000,
                "updatedAt": 1669782394000,
                "language": "en",
                "attributes": {
                    "title": "Samsung Bus Boarding Pass",
                    "transitType": "Bus",
                    "groupingId": "grp-20221130001",
                    "providerLogo": "https://../logoImage.png",
                    "providerLogo.darkUrl": "https://../logoImageDark.png",
                    "providerName": "SAMSUNG BUS",
                    "user": "GALAXY KIM",
                    "userType": "Adult",
                    "transitOperator": "Samsung Transport Co.",
                    "seatNumber": "7-A",
                    "reservationNumber": "RSVNO-202211300001",
                    "transitClass": "Premium",
                    "transitFare": "12,500 WON",
                    "boardingTime": 1671926400000,
                    "boardingTime.utcOffset": "UTC+09:00",
                    "departName": "EAST-SEOUL Bus Terminal",
                    "departGate": "14",
                    "estimatedOrActualStartDate": 1671928200000,
                    "estimatedOrActualStartDate.utcOffset": "UTC+09:00",
                    "arriveName": "SUWON Bus Terminal",
                    "bgColor": "#F1C232",
                    "fontColor": "dark",
                    "appLinkLogo": "https://../applinklogo.png",
                    "appLinkName": "Samsung Bus",
                    "appLinkData": "https://www.samsung-bus.com",
                    "csInfo": "{\"call\": \"555) 123-4567\", \"email\": \"cs@email.com\"}"
                }
            }
        ]
    }
}
```

<pre> \"website\": \"https://homepage.com/cs\", \"barcode.value\": \"161383532125848067541802\", \"barcode.serialType\": \"BARCODE\", \"barcode.ptFormat\": \"QRICODESERIAL\", \"barcode.ptSubFormat\": \"QR_CODE\" }, \"localization\": [{ \"language\": \"ko\", \"attributes\": { \"title\": \"삼성버스 탑승권\", \"providerName\": \"삼성버스\", \"user\": \"김 은하\", \"transitFare\": \"12,500 원\" } }] }, { \"refId\": \"ref-202211300002\", \"createdAt\": 1669782394000, \"updatedAt\": 1669782394000, \"language\": \"en\", \"attributes\": { \"title\": \"Samsung Bus Boarding Pass\", \"transitType\": \"Bus\", \"groupId\": \"grp-20221130001\", \"providerLogo\": \"https://..../logoImage.png\", \"providerLogo.darkUrl\": \"https://..../logoImageDark.png\", \"providerName\": \"SAMSUNG BUS\", \"user\": \"SAMSUNG SAM\", \"userType\": \"Child\", \"transitOperator\": \"Samsung Transport Co.\", \"seatNumber\": \"7-B\", \"reservationNumber\": \"RSVNO-202211300002\", \"transitClass\": \"Preminum\", \"transitFare\": \"9,000 WON\", \"boardingTime\": 1671926400000, \"boardingTime.utcOffset\": \"UTC+09:00\", \"departName\": \"EAST-SEOUL Bus Terminal\", \"departGate\": \"14\", \"estimatedOrActualStartDate\": 1671928200000, \"estimatedOrActualStartDate.utcOffset\": \"UTC+09:00\", \"arriveName\": \"SUWON Bus Terminal\", \"bgColor\": \"#F1C232\", \"fontColor\": \"dark\", \"appLinkLogo\": \"https://..../applinklogo.png\", \"appLinkName\": \"Samsung Bus\", \"appLinkData\": \"https://www.samsung-bus.com\", \"csInfo\": \"{\"call\": \"555) 123-4567\", \"email\": \"cs@email.com\"}\", \"website\": \"https://homepage.com/cs\", \"barcode.value\": \"161383532125848067541802\", </pre>

```
        "barcode.serialType": "BARCODE",
        "barcode.ptFormat": "QRICODESERIAL",
        "barcode.ptSubFormat": "QR_CODE"
    },
    "localization": [
        {
            "language": "ko",
            "attributes": {
                "title": "삼성버스 탑승권",
                "providerName": "삼성버스",
                "user": "사만다 삼성",
                "transitFare": "9,000 원"
            }
        }
    ]
}
```

3.2 Event Ticket

'Event Ticket' cards support event commodities for *performances, sports, movies, entrances, and etc.*

Event tickets in Wallet can provide additional information about the event and an alarm before the event time and expiration. Ticket cards support event commodities for Performances, Sports, Movies, and Entrance.

Wallet Card Type	Wallet Card Sub Type
<i>ticket</i>	<i>performances, sports, movies, entrances, others</i>

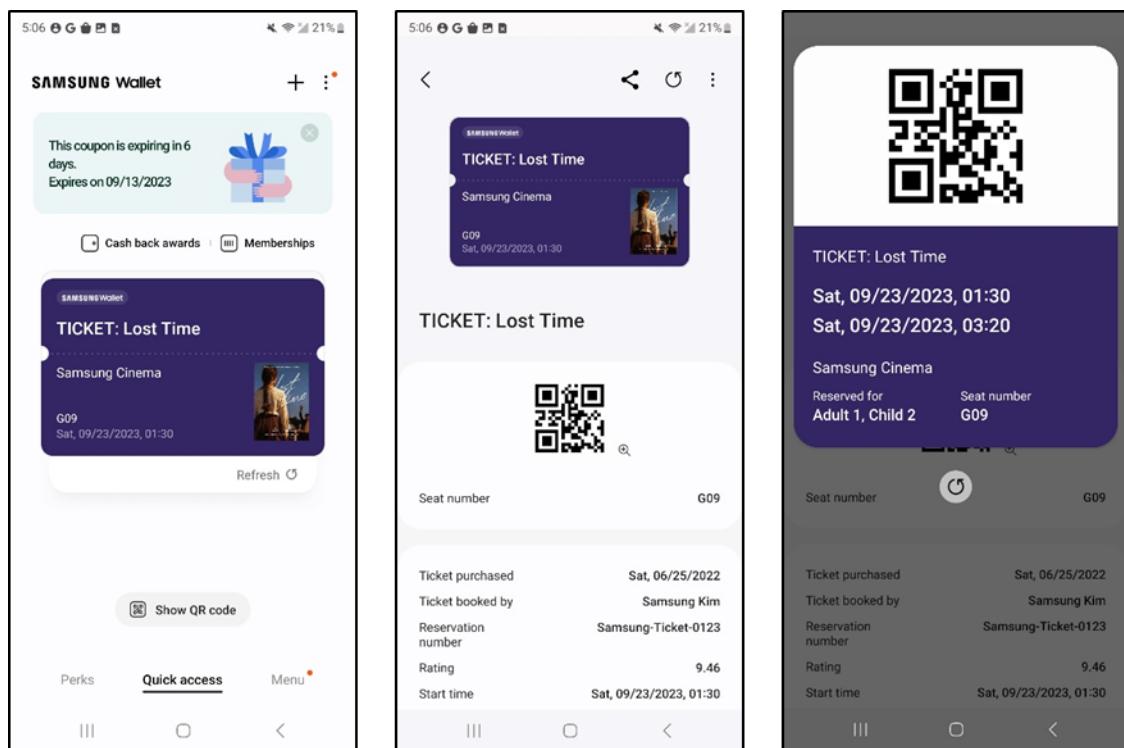


Figure 3-4 Sample: Event Ticket: Movies

Samsung Wallet Cards – API Specification

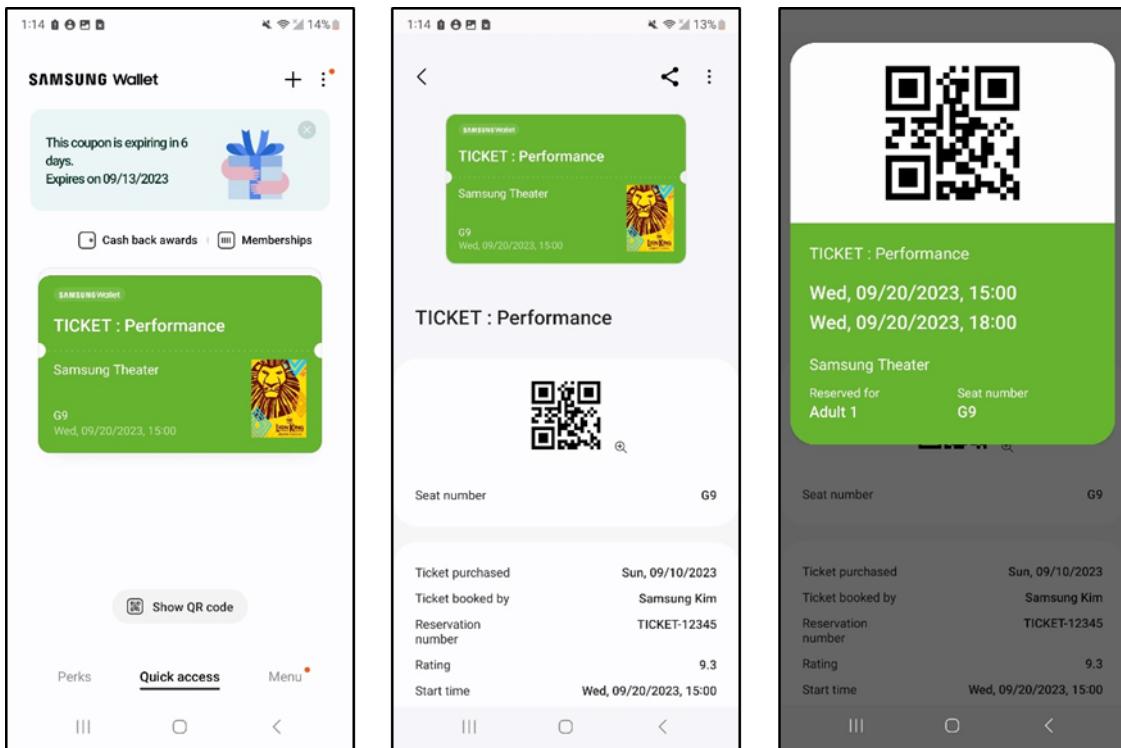


Figure 3-5 Sample: Event Ticket: Performances

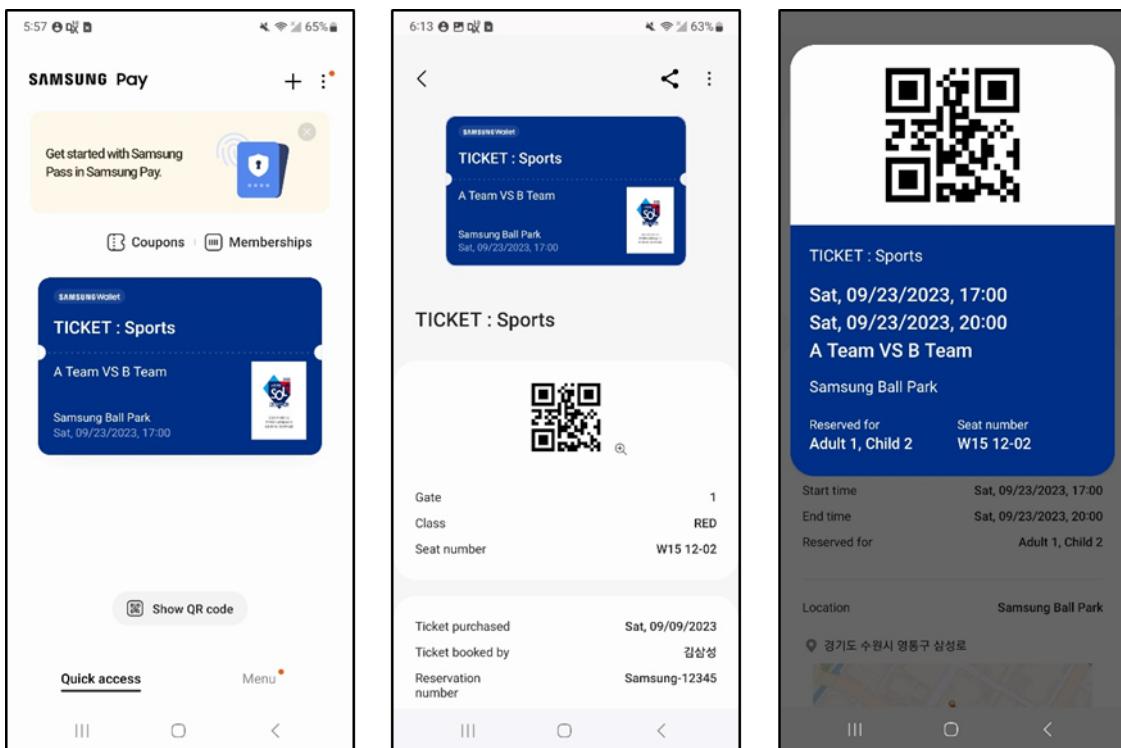


Figure 3-6 Sample: Event Ticket: Sports

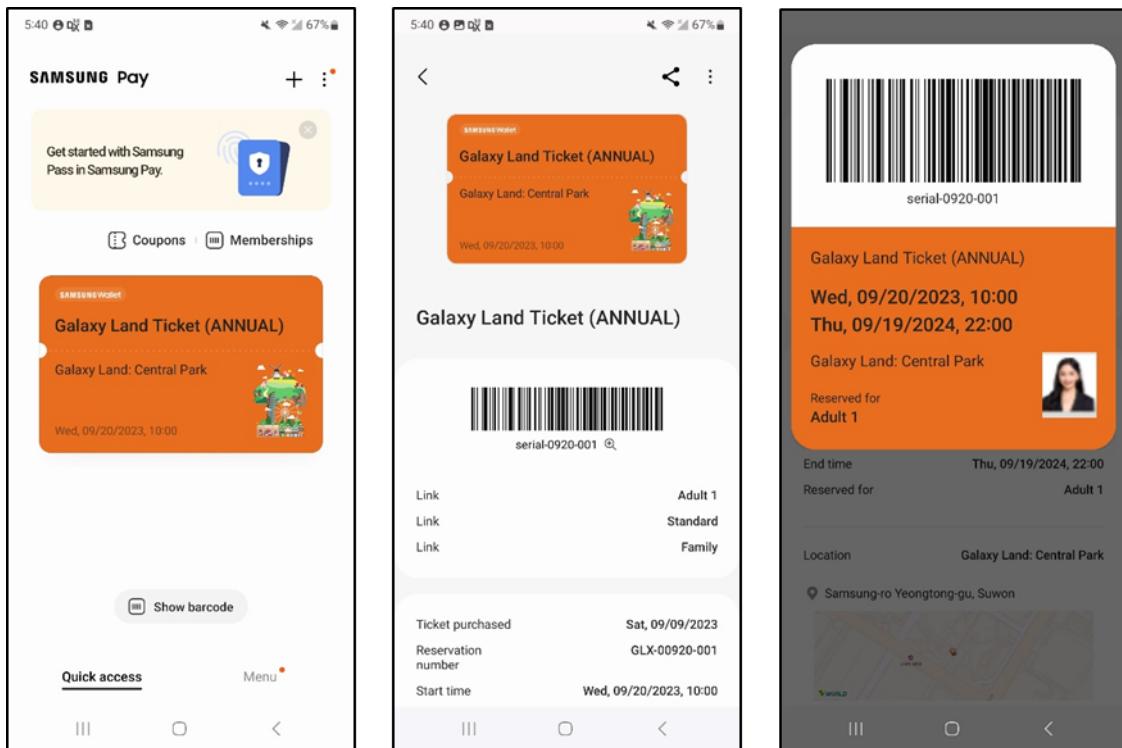


Figure 3-7 Sample: Event Ticket: Entrances

Specification: Ticket

Type	Value			Description
attributes.{fields}	title	String(32)	Required	Main title. e.g., MLB Ticket.
	category	String(16)	Optional	Ticket category * This will be deprecated
	eventId	String(32)	Optional	If full cancelation of the event occurs, find and process all tickets with this ID.
	groupId	String(32)	Optional	Identifier used to group related cards.
	orderId	String(32)	Optional	A unique identifier for an order.
	mainImg	String(256)	Required	URL for main ticket image. The file size should not exceed 512 kB.
	subtitle1	String(32)	Optional	The auxiliary field which displays supporting information.

	logoImage	String(256)	Required	Logo image URL to be displayed in the Card item. The file size should not exceed 256 kB.
	logoImage.darkUrl	String(256)	Required	Logo image URL in dark mode. The file size should not exceed 256 kB.
	logoImage.lightUrl	String(256)	Required	Logo image URL in light mode. The file size should not exceed 256 kB.
	wideImage	String(256)	Optional	Wide horizontal image URL displayed on the Card information screen. The file size should not exceed 256 kB.
	providerName	String(32)	Required	Ticket provider name.
	providerViewLink	String(512)	Optional	Link to additional information from the provider * See Links format
	classification	String(16)	Optional	Classifications of tickets Use <i>ONETIME</i> or <i>REGULAR</i> or <i>ANNUAL</i> * Default: <i>ONETIME</i>
	holderName	String(64)	Optional	Name of card holders
	idPhoto	String(20k)	Optional	Holder's photo image data encoded Base64
	idPhoto.format	String(32)	Optional	Image file format e.g., <i>jpeg</i> , <i>png</i> * Unsupported image formats may exist
	idPhoto.status	String(16)	Optional	Status of the data Allowed value: <i>UNCHANGED</i>
	grade	String(32)	Optional	Ticket grade
	seatClass	String(32)	Optional	Seat class.
	entrance	String(64)	Optional	Entrance gate.
	seatNumber	String(256)	Optional	Seat location.
	seatLayoutImage	String(256)	Optional	URL of the seat layout image. The file size should not exceed 512 kB.
	issueDate	Long(13)	Required	Issue date. Epoch timestamp in milliseconds.
	reservationNumber	String(32)	Required	Reservation number.

	user	String(64)	Optional	Name of person who made the reservation.
	certification	String(16)	Optional	Ticket certification. e.g., R as a film rating.
	reactivatableYn	String(1)	Optional	Flag whether the card is able to activate an expired ticket temporarily. Either 'Y' or 'N' * Default: 'N'
	preventCaptureYn	String(1)	Optional	Flag whether this wallet card view prevents screen capture Either 'Y' or 'N' * Default: 'N'
	noNetworkSupportYn	String(1)	Optional	Sets whether to support to open the wallet card under 'No Network' Status. Either 'Y' or 'N' * Default: 'N'
	startDate	Long(13)	Required	Start date (Display Start date) Epoch timestamp in milliseconds.
	endDate	Long(13)	Optional	End date (Display End date) Epoch timestamp in milliseconds. * If null, the card will expire in 10 hours from <i>startDate</i>
	person1	String(512)	Optional	Number of persons by category. * See Classification format.
	locations	String(1024)	Optional	List of locations where the card can be used. * See Location format.
	noticeDesc	String(1024)	Required	Text of the notice. * Long content is allowed. * See Additional Information format.
	groupInfo1	String(64)	Optional	The first common information with the same <i>groupId</i> .
	groupInfo2	String(64)	Optional	The second common information with the same <i>groupId</i> . * It is recommended to set <i>groupInfo1</i> first.
	groupInfo3	String(64)	Optional	The third common information with the same <i>groupId</i> . * It is recommended to set <i>groupInfo1, groupInfo2</i> first.

	csInfo	String(512)	Optional	Partners' customer service Information. Set value to JSON Object. (call, email, website, sns). * See the example below.
	privacyModeYn	String(1)	Optional	Whether or not to require user authentication when using the card Either 'Y' or 'N' * Default: 'N'
	appLinkLogo	String(256)	Required	App link image URL. The file size should not exceed 256 kB.
	appLinkName	String(32)	Required	App link name.
	appLinkData	String(256)	Required	Information about the partner app link.
	bgColor	String(8)	Optional	Color of the card art. e.g., #00FFFF.
	fontColor	String(8)	Optional	Color of the font on the card art Acceptable values: <i>dark, light</i>
	blinkColor	String(8)	Optional	Color of the blinking effect in the Indicator area e.g., #00FFFF
	barcode.value	String(4096)	Optional	Barcode data, serial number
	barcode.serialType	String(32)	Optional	Presentation Type e.g., SERIALNUMBER, BARCODE, ... * See Barcode format
	barcode.ptFormat	String(32)	Optional	Presentation Format e.g., BARCODE, QRCODE, SERIAL, ... * See Barcode format
	barcode.ptSubFormat	String(32)	Optional	Presentation Sub-format e.g., CODE_128, QR_CODE, ... * See Barcode format
	barcode.errorCorrectionLevel	String(4)	Optional	Amount of redundancy or error correction data included in the code. There are four error correction levels available in QR codes. - Code options: L/M/Q/H
* If coupons are provided together. Samsung Wallet can show up to two coupons per ticket.				
	relCoupon1.title	String(32)	Conditional	1 st coupon title. * Required if this ticket has a related coupon

	relCoupon1.subtitle	String(32)	Optional	1 st coupon subtitle
	relCoupon1.providerName	String(32)	Conditional	1 st coupon provider name. * Required if this ticket has a related coupon
	relCoupon1.imageFileSrc	String(256)	Optional	1 st coupon image URL. The file size should not exceed 256 kB.
	relCoupon1.noticeDescription	String(1024)	Optional	Text of the notice. * Long content is allowed. * See Additional Information format.
	relCoupon1.notificationTime	Long(13)	Optional	Coupon exposure time. Epoch timestamp in milliseconds.
	relCoupon1.value	String(4096)	Conditional	Barcode data, serial number
	relCoupon1.serialType	String(32)	Required	Presentation Type e.g., SERIALNUMBER, BARCODE, ... * See Barcode format
	relCoupon1.ptFormat	String(32)	Conditional	Presentation Format e.g., BARCODE, QRCODE, SERIAL, ... * See Barcode format
	relCoupon1.ptSubFormat	String(32)	Conditional	Presentation Sub-format e.g., CODE_128, QR_CODE, ... * See Barcode format
	relCoupon1.errorCorrectionLevel	String(4)	Optional	Amount of redundancy or error correction data included in the code. There are four error correction levels available in QR codes. - Code options: L/M/Q/H
	relCoupon2.title	String(32)	Conditional	2 nd coupon title. * Required if this ticket has a related 2 nd coupon
	relCoupon2.subtitle	String(32)	Optional	2 nd coupon subtitle
	relCoupon2.providerName	String(32)	Conditional	2 nd coupon provider name. * Required if this ticket has a related 2 nd coupon
	relCoupon2.imageFileSrc	String(256)	Optional	2 nd coupon image URL. The file size should not exceed 256 kB.
	relCoupon2.noticeDescription	String(1024)	Optional	Text of the notice. * Long content is allowed. * See Additional Information format.

	relCoupon2.notificationTime	Long(13)	Optional	Coupon exposure time. Epoch timestamp in milliseconds.
	relCoupon2.value	String(4096)	Conditional	Barcode data, serial number
	relCoupon2.serialType	String(32)	Required	Presentation Type e.g., SERIALNUMBER, BARCODE, ... * See Barcode format
	relCoupon2.ptFormat	String(32)	Conditional	Presentation Format e.g., BARCODE, QRCODE, SERIAL, ... * See Barcode format
	relCoupon2.ptSubFormat	String(32)	Conditional	Presentation Sub-format e.g., CODE_128, QR_CODE, ... * See Barcode format
	relCoupon2.errorCorrectionLevel	String(4)	Optional	Amount of redundancy or error correction data included in the code. There are four error correction levels available in QR codes. - Code options: L/M/Q/H

Example	<pre>{ "card": { "type": "ticket", "subType": "entrances", "data": [{ "refId": "ent-ticket-0613001", "createdAt": 1686657600000, "updatedAt": 1686657600000, "language": "en", "attributes": { "title": "Galaxy Land Entrance Ticket", "subtitle1": "Standard", "classification": "ANNUAL", "groupId": "group-0613001", "orderId": "ent-0613001", "mainImg": "https://..../main.png", "logoImage": "https://..../logo.png", "providerName": "Galaxy Entertainment", "logoImage.darkUrl": "https://..../logo-dark.png", "issueDate": 1686657600000, "reservationNumber": "GLX-0613-001", "startDate": 1686657600000, "endDate": 1718280000000, "holderName": "Kim Eunha", "idPhoto.data": "Base64-encoded{image-file-data}", "idPhoto.format": "png", "grade": "Family", "person1": "{\"person\": [{\"category\": \"Adult\", \"count\": 1}]}", "locations": "[{\"lat\": 37.256518, \"lng\": 127.053516, \"address\": \"Samsung-ro Yeongtong-gu, Suwon\", \"name\": \"Galaxy Land: Central Park\"}]", "noticeDesc": "{\"count\": 2, \"info\": [{\"title\": \"NOTICE 1\", \"content\": \"DESCRIPTION 1-1\", \"description\": \"DESCRIPTION 1-2\"}, {\"title\": \"NOTICE 2\", \"content\": \"DESCRIPTION 2-1\"}]}", "groupInfo1": "Adult 1", "groupInfo2": "Standard", "groupInfo3": "Family", "csInfo": "{\"call\": \"555) 123-4567\", \"email\": \"cs@email.com\", \"website\": \"https://homepage.com/cs\", \"facebook\": \"https://facebook.com/cs\", \"instagram\": \"https://instagram.com/cs\", \"youtube\": \"https://youtube.com/cs\"}", "appLinkName": "Galaxy ticket", "appLinkLogo": "https://..../applinklogo.png", "appLinkData": "https://www.applinkdata.com", "bgColor": "#E86D1F", "fontColor": "light", "blinkColor": "#E86D1F", "barcode.value": "serial-0613-001", "barcode.serialType": "QRcode", "barcode.ptFormat": "QRcodeSerial", "barcode.ptSubFormat": "QR_CODE" }, "localization": [</pre>
---------	---

```
    "language": "ko",
    "attributes": {
        "title": "갤럭시 랜드 입장권",
        "holderName": "김은하",
        "person1": "{\"person\": [{\"category\": \"어른\", \"count\": 1}]}",
        "locations": "[{\"lat\": 37.256518, \"lng\": 127.053516, \"address\": \"Samsung-ro Yeongtong-gu, Suwon\", \"name\": \"갤럭시 랜드 센트럴 파크\"]}",
        "noticeDesc": "{\"count\": 2, \"info\": [{\"title\": \"공지사항 1\", \"content\": [\"설명 1-1\", \"설명 1-2\"]}, {\"title\": \"공지사항 2\", \"content\": [\"설명 2-1\"]}]}",
        "groupInfo1": "어른 1"
    }
}
```

3.3 Coupon

'Coupon' cards support digitized redeemable voucher.

Coupons in Wallet for various channels can provide alarms before expiration and update usage status.

Wallet Card Type	Wallet Card Sub Type
<i>coupon</i>	<i>others</i>

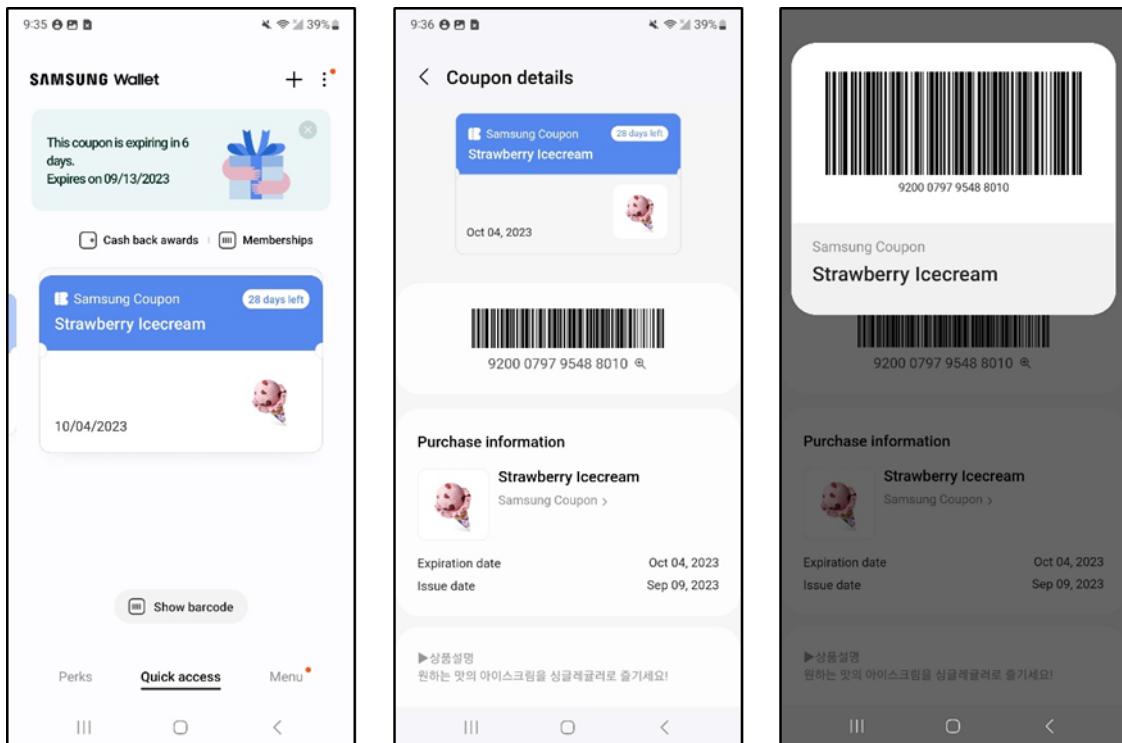


Figure 3-8 Sample: Coupon

Specification: Coupon

Type	Value			Description
attributes.{fields}	title	String(32)	Required	Main title. e.g., Free coupon.
	orderId	String(32)	Optional	Unique identifier for an order.
	mainImg	String(256)	Required	URL for the main coupon image. The file size should not exceed 512 kB.

	brandName	String(32)	Optional	Brand name.
	expiry	Long(13)	Required	Expiry date in timestamp format. Issued data is cleaned up after it expires. Epoch timestamp in milliseconds.
	issueDate	Long(13)	Required	Issue date. Epoch timestamp in milliseconds.
	redeemDate	Long(13)	Optional	Date when the coupon is used. Epoch timestamp in milliseconds.
	noticeDesc	String(1024)	Optional	Text of notice. *HTML supported
	editableYn	String(1)	Required	Flag whether the coupon can be modified. Either 'Y' or 'N'.
	deletableYn	String(1)	Required	Flag whether the coupon can be deleted. Either 'Y' or 'N'.
	displayRedeemButtonYn	String(1)	Required	Flag whether the use completion button is displayed. Either 'Y' or 'N'.
	notificationYn	String(1)	Required	Flag whether a notification related to the coupon is delivered. Either 'Y' or 'N'.
	appLinkLogo	String(256)	Required	App link image URL. The file size should not exceed 256 kB.
	appLinkName	String(32)	Required	App link name.
	appLinkData	String(256)	Required	Information about the partner app link.
	preventCaptureYn	String(1)	Optional	Flag whether this wallet card view prevents screen capture Either 'Y' or 'N' * Default: 'N'
	barcode.value	String(4096)	Conditional	Barcode data, serial number. * Required if serialType isn't 'None'.
	barcode.value2	String(4096)	Conditional	The secondary barcode data. * Required if <i>ptFormat</i> is <i>DUALBARCODE</i> or <i>DUALBARCODESERIAL</i>
	barcode.serialType	String(32)	Optional	Presentation Type e.g., SERIALNUMBER, BARCODE, ... * See Barcode format

Samsung Wallet Cards – API Specification

	barcode.ptFormat	String(32)	Optional	Presentation Format e.g., BARCODE, QRCODE, SERIAL, ... * See Barcode format
	barcode.ptSubFormat	String(32)	Optional	Presentation Sub-format e.g., CODE_128, QR_CODE, ... * See Barcode format
	barcode.errorCorrectionLevel	String(4)	Optional	Amount of redundancy or error correction data included in the code. There are four error correction levels available in QR codes. - Code options: L/M/Q/H
	balance	String(50)	Optional	Initial balance. This is going to be shown as received. It is recommended to use a one letter currency symbol. e.g., \$ 1,000, 1,000 P
	summaryUrl	String(256)	Optional	Webpage url that show details, such as balance
	userMessage	String(256)	Optional	Message to forward
	sender	String(64)	Optional	Name of sender

Example	<pre>{ "card": { "type": "coupon", "subType": "others", "data": [{ "refId": "ref-230712-0001", "createdAt": 1612660039000, "updatedAt": 1612660039000, "language": "en", "attributes": { "title": "Free Coupon", "mainImg": "https://..../main.png", "brandName": "Samsung Coupon", "expiry": 1637802725000, "issueDate": 1637457125000, "editableYn": "N", "deletableYn": "N", "displayRedeemButtonYn": "Y", "notificationYn": "Y", "appLinkName": "OO Voucher", "appLinkLogo": "https://..../applinklogo.png", "appLinkData": "https://www.oocoupon.com", "barcode.value": "CS16138353212584806754FG1802 ", "barcode.serialType": "BARCODE", "barcode.ptFormat": "QRCODESERIAL", "barcode.ptSubFormat": "QR_CODE" } }] } }</pre>
---------	--

3.4 Gift Card

'Gift Card' cards support enrolling prepaid cards also known as gift certificate, gift voucher or gift token.

Links (URLs) to get balance and transactions history in real time is provided in the [Partner portal](#). If a partner needs to integrate communication between Samsung Wallet server and the partner's server to support the feature, the partner has to set the links in [Partner portal](#).

Wallet Card Type	Wallet Card Sub Type
<i>giftcard</i>	<i>others</i>

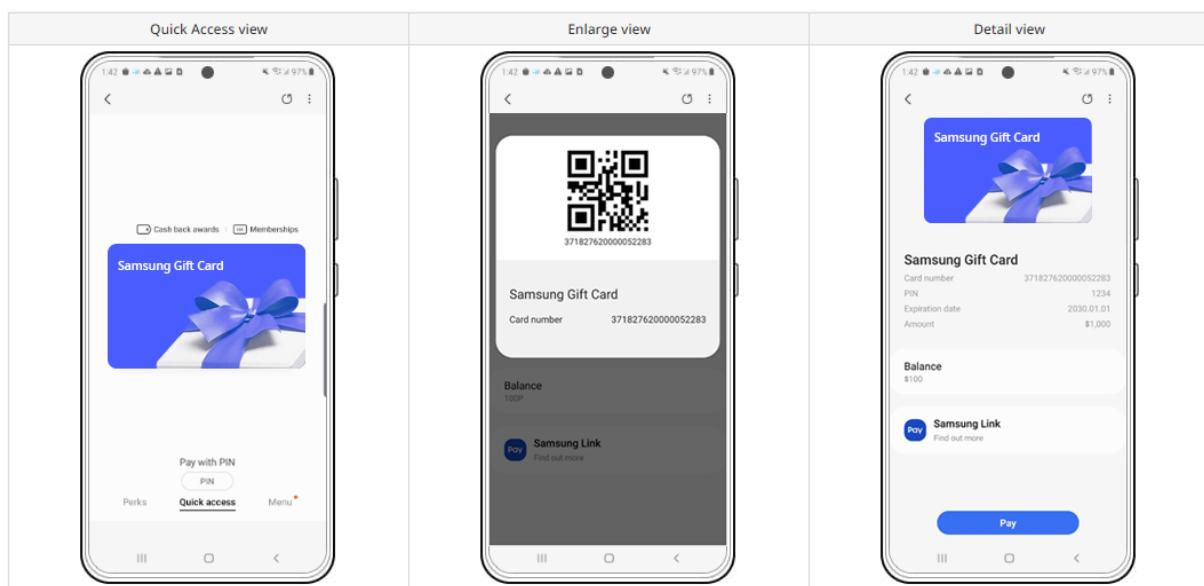


Figure 3-9 Sample: Gift Card

Specification: Gift card

Type	Value			Description
attributes.{fields}	title	String(32)	Required	Main title. e.g., Samsung Gift Card.
	eventId	String(36)	Optional	If full cancelation of the event occurs, find and process all gift cards with this ID.
	orderId	String(36)	Optional	A unique identifier for an order.
	subtitle1	String(32)	Optional	The auxiliary field which displays supporting information.

	logoImage	String(256)	Optional	Logo image URL to be displayed in the Card item. The file size should not exceed 256 kB.
	logoImage.darkUrl	String(256)	Optional	Logo image URL in dark mode. The file size should not exceed 256 kB.
	logoImage.lightUrl	String(256)	Optional	Logo image URL in light mode. The file size should not exceed 256 kB.
	providerName	String(32)	Required	Gift Card provider name.
	user	String(64)	Optional	Name of person who holds the gift card.
	preventCaptureYn	String(1)	Optional	Flag whether this wallet card view prevents screen capture. Either 'Y' or 'N', the default value is 'N'.
	startDate	Long(13)	Optional	Start date (Display Start date). Epoch timestamp in milliseconds.
	endDate	Long(13)	Optional	End date (Display End date). Epoch timestamp in milliseconds.
	locations	String(1024)	Optional	List of locations where the card can be used. * See Location format.
	noticeDesc	String(1024)	Optional	Text of the notice. * Long content is allowed. * See Additional Information format.
	csInfo	String(512)	Optional	Set value to JSON Object. (call, email, website). * See the example below.
	appLinkLogo	String(256)	Required	App link image URL. The file size should not exceed 256 kB.
	appLinkName	String(32)	Required	App link name.
	appLinkData	String(256)	Required	Information about the partner app link.
	bgImage	String(256)	Optional	URL for card art (background) image.
	mainImg	String(256)	Optional	URL for gift card image. The file size should not exceed 512 kB.

	bgColor	String(8)	Optional	Color of the card art. e.g., #00FFFF.
	fontColor	String(8)	Optional	Color of the font on the card art Acceptable values: <i>dark, light</i>
	blinkColor	String(8)	Optional	Color of the blinking effect which indicates that a card cannot be captured in the Indicator area. e.g., #00FFFF.
	barcode.value	String(4096)	Optional	Barcode data, serial number
	barcode.serialType	String(32)	Optional	Presentation Type e.g., SERIALNUMBER, BARCODE, ... * See Barcode format
	barcode.ptFormat	String(32)	Optional	Presentation Format e.g., BARCODE, QRCODE, SERIAL, ... * See Barcode format
	barcode.ptSubFormat	String(32)	Optional	Presentation Sub-format e.g., CODE_128, QR_CODE, ... * See Barcode format
	barcode.pin	String(16)	Optional	Pin to show with a barcode.
	barcode.errorCorrectionLevel	String(4)	Optional	Amount of redundancy or error correction data included in the code. There are four error correction levels available in QR codes. - Code options: L/M/Q/H
	merchantId	String(36)	Optional	Merchant Identifier.
	merchantName	String(32)	Optional	Merchant Name to display.
	amount	String(32)	Optional	Initial balance. This is going to be shown as received. e.g., \$ 1,000
	balance	String(32)	Optional	Remaining balance. This is going to be shown as received. e.g., \$ 1,000
	summaryUrl	String(256)	Optional	Web url that show details, such as balance or transactions history.

Example	<pre>{ "card": { "type": "giftcard", "subType": "others", "data": [{ "refId": "b3fdc982-28c9-47a3-b02f-d484779698a7", "createdAt": 1672574400000, "updatedAt": 1672574400000, "language": "en", "attributes": { "title": "Samsung Gift Card", "eventId": "event-001", "logoImage": "https://gpp.walletsvc.samsung.com/mcs/images/contents/wallet_intro_logo.png", "logoImage.darkUrl": "https://gpp.walletsvc.samsung.com/mcs/images/contents/wallet_intro_logo.png", "providerName": "Samsung Gift Card Provider", "user": "MS. Jane Doe", "noticeDesc": "Gift Card Test", "csInfo": "{\"call\":\"555) 123-4567\", \"email\":\"cs@email.com\", \"website\":\"https://homepage.com/cs\"}", "appLinkLogo": "https://play-lh.googleusercontent.com/ZnFa1roZ7hpv9j-jIAcBjmjuDl2x-FnuwTE0Yvbbcwf5VPzOQQiKBXGK7d-APTvag=w240-h480-rw", "appLinkName": "Gift Card Link", "appLinkData": "https://www.samsung.com/", "bgColor": "#0A1A4F", "fontColor": "light", "blinkColor": "#00FFFF", "barcode.value": "CS16138353212584806754FG1802", "barcode.serialType": "QRCode", "barcode.ptFormat": "QRCodeSerial", "barcode.ptSubFormat": "QR_CODE" } }] } }</pre>
---------	---

3.5 Loyalty

'Loyalty' cards support enrolling loyalty cards also known as membership.

Links (URLs) to get points in real time can be provided in the [Partner portal](#). If a partner needs to integrate communication between Samsung Wallet server and the partner's server to support the feature, the partner has to set the links in the [Partner portal](#).

Wallet Card Type	Wallet Card Sub Type
<i>loyalty</i>	<i>others</i>

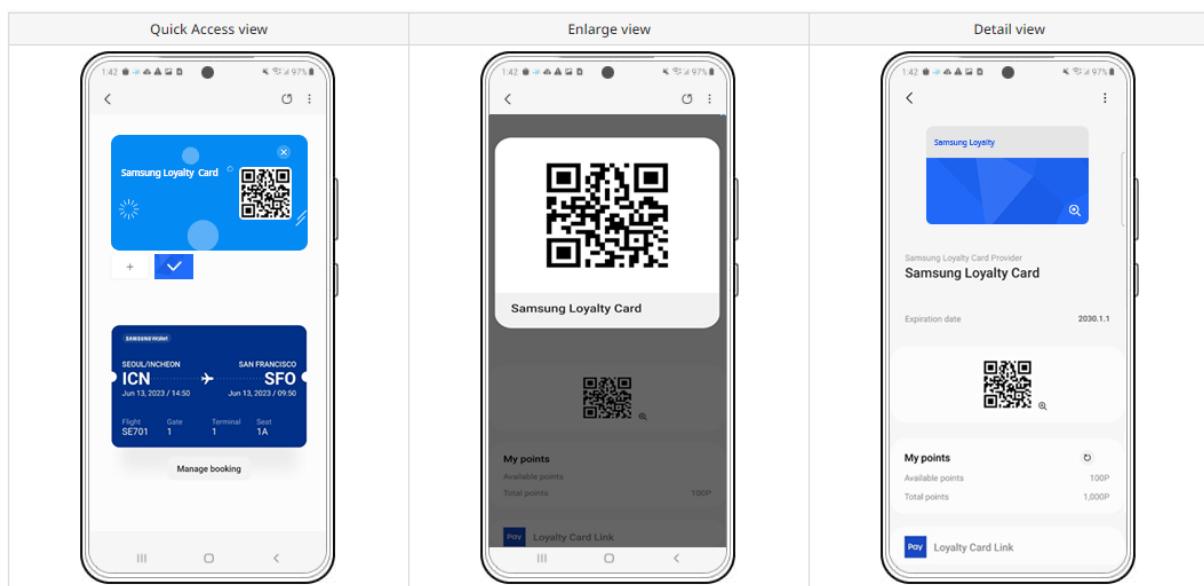


Figure 3-10 Sample: Loyalty

Specification: Loyalty

Type	Value			Description
attributes.{fields}	title	String(32)	Required	Main title. e.g., Samsung Loyalty Card.
	eventId	String(36)	Optional	If full cancelation of the event occurs, find and process all loyalty cards with this ID.
	groupId	String(36)	Optional	Identifier used to group related cards.
	orderId	String(36)	Optional	A unique identifier for an order.

	mainImg	String(256)	Optional	URL for main loyalty card image. The file size should not exceed 512 kB.
	subtitle1	String(32)	Optional	The auxiliary field which displays supporting information.
	logoImage	String(256)	Optional	Logo image URL to be displayed in the Card item. The file size should not exceed 256 kB.
	logoImage.darkUrl	String(256)	Optional	Logo image URL in dark mode. The file size should not exceed 256 kB.
	logoImage.lightUrl	String(256)	Optional	Logo image URL in light mode. The file size should not exceed 256 kB.
	providerName	String(32)	Required	Loyalty Card provider name.
	preventCaptureYn	String(1)	Optional	Flag whether this wallet card view prevents screen capture. Either 'Y' or 'N', the default value is 'N'.
	startDate	Long(13)	Optional	Start date (Display Start date). Epoch timestamp in milliseconds.
	endDate	Long(13)	Optional	End date (Display End date). Epoch timestamp in milliseconds.
	locations	String(1024)	Optional	List of locations where the card can be used. * See Location format.
	noticeDesc	String(1024)	Optional	Text of the notice. * Long content is allowed. * See Additional Information format.
	csInfo	String(512)	Optional	Set value to JSON Object. (call, email, website or social media which supports Facebook, Pinterest, X(Twitter) and Instagram). * See the example below.
	appLinkLogo	String(256)	Required	App link image URL. The file size should not exceed 256 kB.
	appLinkName	String(32)	Required	App link name.
	appLinkData	String(256)	Required	Information about the partner app link.
	bgImage	String(256)	Optional	Background image for a card art.

	bgColor	String(8)	Optional	Color of a card art. e.g., #00FFFF.
	fontColor	String(8)	Optional	Color of the font on the card art Acceptable values: <i>dark, light</i>
	blinkColor	String(8)	Optional	Color of the blinking effect which indicates that a card cannot be captured in the Indicator area. e.g., #00FFFF.
	barcode.value	String(4096)	Optional	Barcode data, serial number
	barcode.serialType	String(32)	Optional	Presentation Type e.g., SERIALNUMBER, BARCODE, ... * See Barcode format
	barcode.ptFormat	String(32)	Optional	Presentation Format e.g., BARCODE, QRCODE, SERIAL, ... * See Barcode format
	barcode.ptSubFormat	String(32)	Optional	Presentation Sub-format e.g., CODE_128, QR_CODE, ... * See Barcode format
	barcode.errorCorrectionLevel	String(4)	Optional	Amount of redundancy or error correction data included in the code. There are four error correction levels available in QR codes. - Code options: L/M/Q/H
	merchantId	String(36)	Optional	Merchant Identifier
	merchantName	String(32)	Optional	Merchant Name to display.
	amount	String(32)	Optional	Total amount of points or initial balance. This is going to be shown as received. It is recommended to use a one letter currency symbol. e.g., \$ 1,000, 1,000P
	balance	String(32)	Optional	Available points or remaining balance. This is going to be shown as received. It is recommended to use a one letter currency symbol. e.g., \$ 1,000, 1,000P
	summaryUrl	String(256)	Optional	Webpage url that show details, such as balance or transactions history.

Example	<pre>{ "card": { "type": "loyalty", "subType": "others", "data": [{ "refId": "b3fdc982-28c9-47a3-b02f-d484779698a8", "createdAt": 1672574400000, "updatedAt": 1672574400000, "language": "en", "attributes": { "title": "Samsung Loyalty Card", "eventId": "event-001", "logoImage": "https://gpp.walletsvc.samsung.com/mcs/images/contents/wallet_intro_logo.png", "logoImage.darkUrl": "https://gpp.walletsvc.samsung.com/mcs/images/contents/wallet_intro_logo.png", "providerName": "Samsung Loyalty Card Provider", "noticeDesc": "Loyalty Card Test", "csInfo": "{\"call\": \"555) 123-4567\", \"email\": \"cs@email.com\", \"website\": \"https://homepage.com/cs\"}", "appLinkLogo": "https://play-lh.googleusercontent.com/ZnFa1roZ7hpv9j-jIAcBjmjuDl2x-FnuwTE0OYvbbcwf5VPzOQQjKBXGK7d-APTvag=w240-h480-rw", "appLinkName": "Loyalty Card Link", "appLinkData": "https://www.samsung.com/", "bgColor": "#0A1A4F", "barcode.value": "CS16138353212584806754FG1802", "barcode.serialType": "QRCODE", "barcode.ptFormat": "QRCODESERIAL", "barcode.ptSubFormat": "QR_CODE", "amount": "1,000P", "balance": "500P" } }] } }</pre>
---------	--

3.6 ID Card

'ID Card' cards are used to present identification for employees, students, drivers, guests, etc.

Wallet Card Type	Wallet Card Sub Type
<i>idcard</i>	<i>employees, nationals, students, drivers, guests, others</i>

- TBD -

Figure 3-11 Sample: ID Card

Specification: ID Card

Type	Value			Description
nattributes.{ fields}	title	String(32)	Required	Main title of cards e.g., <i>Commercial Access, Employee badge</i>
	holderName	String(64)	Required	Name of card holders
	secondHolderName	String(64)	Optional	Second Name of card holders
	organization	String(64)	Optional	Organization of card holders belonging i.e., name of department, division, affiliation, association or team
	position	String(64)	Optional	Position of card holders
	identifier	String(64)	Required	Unique ID value
	idNumber	String(64)	Optional	Representative value for an ID i.e., ID number, Document number
	address	String(256)	Optional	Address of card holders
	placeOfLocation	String(64)	Optional	Place of location associated with the cardholder
	idPhoto	String(128k)	Optional	Holder's photo image data encoded Base64 The file size should not be greater than 128 kB

	<code>idPhoto.format</code>	<code>String(32)</code>	Optional	Image file format e.g., <code>jpeg, png</code> * Unsupported image formats may exist
	<code>idPhoto.status</code>	<code>String(16)</code>	Optional	Status of the data Allowed value: <code>UNCHANGED</code>
	<code>document</code>	<code>String(1024k)</code>	Optional	First document of identity
	<code>document.format</code>	<code>String(32)</code>	Optional	Document format - Allowed value: <code>pdf, jpeg, png</code>
	<code>document.status</code>	<code>String(16)</code>	Optional	Status of the data Allowed value: <code>UNCHANGED</code>
	<code>provisioningType</code>	<code>String(16)</code>	Optional	A value indicating how to provision content that affects card features Allowed value: <code>sdk, ese, usim, token</code>
	<code>provisioningFactor</code>	<code>String(512)</code>	Optional	Elements to complete provisioning
	<code>issueDate</code>	<code>Long(13)</code>	Required	Issue date Epoch timestamp in milliseconds.
	<code>birthdate</code>	<code>String(16)</code>	Optional	Date of birth
	<code>gender</code>	<code>String(16)</code>	Optional	Gender of card holders
	<code>classification</code>	<code>String(16)</code>	Optional	Classified identity type
	<code>expiry</code>	<code>Long(13)</code>	Optional	Expiry date in timestamp format Issued data is cleaned up after it expires Epoch timestamp in milliseconds
	<code>contacts</code>	<code>String(32)</code>	Optional	Personal contact information such as phone number
	<code>uiccId</code>	<code>String(32)</code>	Optional	Universal IC Card ID * deprecated
	<code>logoImage</code>	<code>String(256)</code>	Optional	Logo image URL to be displayed in card item The file size should not exceed 256 kB
	<code>logoImage.darkUrl</code>	<code>String(256)</code>	Optional	Logo image URL in dark mode The file size should not exceed 256 kB * If this value does not exist, <code>logoImage</code> will be substituted

	logoImage.lightUrl	String(256)	Optional	Logo image URL in light mode The file size should not exceed 256 kB * If this value does not exist, <i>logoImage</i> will be substituted
	logoText	String(16)	Optional	Text as an alternative to <i>logoImage</i>
	providerName	String(32)	Required	Content provider name
	issuerName	String(32)	Optional	Place of issue or Issuing authority name
	extraInfo	String(512)	Optional	Additional information * See Additional Information format
	noticeDesc	String(1024)	Optional	Text of the notice * Long content is allowed * See Additional Information format
	csInfo	String(512)	Required	Set value to JSON Object (call, email, website) * See the example below
	privacyModeYn	String(1)	Optional	Whether or not to require user authentication when using the card Either 'Y' or 'N' * Default: 'N'
	appLinkLogo	String(256)	Optional	App link image URL The file size should not exceed 256 kB
	appLinkName	String(32)	Optional	App link name
	appLinkData	String(256)	Optional	Information about the partner app link
	locations	String(1024)	Optional	List of locations where the card can be used * See Location format
	coverImage	String(256)	Optional	Card cover image URL The file size should not exceed 512 kB
	bgImage	String(256)	Optional	Card background image URL The file size should not exceed 512 kB
	bgColor	String(8)	Optional	Color of the card art Support Hex color code e.g., #015AAA
	fontColor	String(8)	Optional	Color of the font on the card art Allowed value: <i>black, white</i>

	blinkColor	String(8)	Optional	Color of the blinking effect in the Indicator area Support Hex color code e.g., #015AAA
	preventCaptureYn	String(1)	Optional	Flag whether this wallet card view prevents screen capture Either 'Y' or 'N' * Default: 'N'
	barcode.value	String(4096)	Optional	Barcode data, serial number
	barcode.serialType	String(32)	Optional	Presentation Type e.g., SERIALNUMBER, BARCODE, ... * See Barcode format
	barcode.ptFormat	String(32)	Optional	Presentation Format e.g., BARCODE, QRCODE, SERIAL, ... * See Barcode format
	barcode.ptSubFormat	String(32)	Optional	Presentation Sub-format e.g., CODE_128, QR_CODE, ... * See Barcode format
	barcode.errorCorrectionLevel	String(4)	Optional	Amount of redundancy or error correction data included in the code. There are four error correction levels available in QR codes. - Code options: L/M/Q/H * Default: L
	barcode.Interval	String(4)	Optional	Update interval if support for dynamic updates Epoch timestamp in milliseconds.

Example	<pre>* Employee ID Card { "card": { "type": "idcard", "subType": "employees", "data": [{ "refId": "20000130003", "createdAt": 1658385817000, "updatedAt": 1658385817000, "language": "en", "attributes": { "title": "Employee ID Card", "holderName": "Kim Samsung", "secondHolderName": "Samsung", "organization": "Digital Wallet, MX", "position": "Professional", "identifier": "2306070003", "idPhoto": "{base64 encoded image data}", "idPhoto.format": "jpeg", "issueDate": "1658385817000", "expiry": "1765855665000", "logoImage": "https://../logo.png", "providerName": "Mobile ID", "extraInfo": { "count": 1, "info": [{ "title": "shortCode", "content": "404457" }] }, "noticeDesc": { "count": 2, "info": [{ "title": "NOTICE1", "content": ["DESCRIPTION1", "DESCRIPTION2"] }, { "title": "NOTICE2", "content": ["DESCRIPTION1", "DESCRIPTION2"] }], "csInfo": { "call": "555) 123-4567", "email": "cs@email.com", "website": "https://homepage.com/cs" }, "appLink": { "name": "Mobile ID One", "logo": "https://../applinklogo.png", "data": "https://www.applinkOrWebLink.com", "coverImage": "https://../card/cover.png", "bgImage": "https://../card/background.png", "fontColor": "dark" } } }, "localization": [{ "language": "ko", "attributes": { "title": "\uacfc\ud55c\ud0d0", "holderName": "\uc0ac\uc131\ud55c", "secondHolderName": "Kim Samsung", "providerName": "\ub2e8\ub300 \uc131\ud55c\ud0dd", "extraInfo": { "count": 1, "info": [{ "title": "\uacfc\ud55c\ud0d0", "content": ["\uacfc\ud55c\ud0d0\ud55c\ud0d0", "\uacfc\ud55c\ud0d0\ud55c\ud0d0"] }] } } }] }] } }</pre>
---------	--

```

* National ID Card
{
    "card": {
        "type": "idcard",
        "subType": "nationals",
        "data": [
            {
                "refId": "IdentityCard-bpo1r3e5-3313-0991-z404-sq12994414u8",
                "createdAt": 1658385817000,
                "updatedAt": 1658385817000,
                "language": "en",
                "attributes": {
                    "title": "National ID Card",
                    "holderName": "Kim Samsung",
                    "secondHolderName": "Samsung",
                    "organization": "Digital Wallet, MX",
                    "position": "Professional",
                    "identifier": "2306070003",
                    "idPhoto": "{base64 encoded image data}",
                    "idPhoto.format": "jpeg",
                    "document": "{base64 encoded pdf data}",
                    "document.format": "pdf",
                    "issueDate": "1658385817000",
                    "expiry": "1765855665000",
                    "logoImage": "https://..../logo.png",
                    "provisioningType": "token",
                    "provisioningFactor": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWliOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvAG4gRG9IiwiWF0IjoxNTE2MjM5MDIyfQ.SflKxwRJSMeKKF2QT4fpMeJf36POk6yJV_adQssw5c",
                    "providerName": "Samsung Electronics",
                    "extraInfo": {
                        "count": 1,
                        "info": [
                            {
                                "title": "shortCode",
                                "content": "404457"
                            }
                        ]
                    },
                    "noticeDesc": [
                        {
                            "count": 2,
                            "info": [
                                {
                                    "title": "NOTICE1",
                                    "content": "DESCRIPTION1\\\"\\\"DESCRIPTION2\\\"\\\"",
                                    "csInfo": {
                                        "call": "(555) 123-4567",
                                        "email": "cs@email.com",
                                        "website": "https://homepage.com/cs"
                                    },
                                    "appLinkName": "Samsung Electronics",
                                    "appLinkLogo": "https://..../applinklogo.png",
                                    "appLinkData": "https://www.applinkOrWebLink.com",
                                    "coverImage": "https://..../card/cover.png",
                                    "bgImage": "https://..../card/background.png",
                                    "fontColor": "dark",
                                    "barcode.value": "5728306720836720763017601",
                                    "barcode.serialType": "QRCODE",
                                    "barcode.ptFormat": "QRIMAGE",
                                    "barcode.ptSubFormat": "QR_CODE",
                                    "barcode.interval": "300000"
                                }
                            ]
                        }
                    ]
                }
            }
        ]
    }
}

```

```
        "language": "ko",
        "attributes": {
            "title": "신분증",
            "holderName": "김 삼성",
            "secondHolderName": "Kim Samsung",
            "providerName": "삼성전자"
        }
    }]
}
```

3.7 Generic Card

'Generic Card' is defined for registering various forms of cards that aren't defined as other types. Partners can customize the items on the Generic Card to display by connecting them with card data.

For further information, please visit our partner portal.

Wallet Card Type	Wallet Card Sub Type
<i>generic</i>	<i>others</i>

TBD

Specification: Generic Card

Type	Value			Description
attributes.{fields}	title	String(32)	Required	Main title
	subtitle	String(32)	Optional	The auxiliary field which displays supporting information.
	providerName	String(32)	Required	Provider name.
	eventId	String(32)	Optional	Event Identifier
	groupId	String(32)	Optional	Identifier used to group related cards.
	startDate	Long(13)	Required	Start date (Display Start date) Epoch timestamp in milliseconds.
	startDate.relativeNotiTime	String(4)	Optional	The relative time from <i>startDate</i> in minutes to provide a notification to the user e.g., 5, 10, 15, 30, 60, and up to 2880
	endDate	Long(13)	Optional	End date (Display End date) Epoch timestamp in milliseconds.
	endDate.relativeNotiTime	String(4)	Optional	The relative time from <i>endDate</i> in minutes to provide a notification to the user e.g., 5, 10, 15, 30, 60, and up to 2880
	mainImg	String(256)	Required	URL for main image. The file size should not exceed 512 kB.

	logoImage	String(256)	Optional	Logo image URL to be displayed in card item The file size should not exceed 256 kB
	logoImage.darkUrl	String(256)	Optional	Logo image URL in dark mode The file size should not exceed 256 kB * If this value does not exist, <i>logoImage</i> will be substituted
	logoImage.lightUrl	String(256)	Optional	Logo image URL in light mode The file size should not exceed 256 kB * If this value does not exist, <i>logoImage</i> will be substituted
	coverImage	String(256)	Optional	Card cover image URL The file size should not exceed 512 kB
	bgImage	String(256)	Optional	Card background image URL The file size should not exceed 512 kB
	text{i}	String(64)	Optional	Text item i: 1~15
	image{i}	String(128k)	Optional	Image item i: 1~5
	image{i}.lightUrl	String(256)	Optional	Image item in light mode
	image{i}.darkUrl	String(256)	Optional	Image item in dark mode
	image{i}.status	String(16)	Optional	Image item status
	timestamp{i}	Long(13)	Optional	Timestamp item i: 1~5
	timestamp{i}.utcOffset	String(8)	Optional	UTC offset of timestamp
	serial{i}	String(4096)	Optional	Serial item i: 1~2
	serial{i}.serialType	String(32)	Optional	Presentation Type e.g., SERIALNUMBER, BARCODE, ... * See Barcode format
	serial{i}.ptFormat	String(32)	Optional	Presentation Format e.g., BARCODE, QRCODE, SERIAL, ... * See Barcode format
	serial{i}.ptSubFormat	String(32)	Optional	Presentation Sub-format e.g., CODE_128, QR_CODE, ... * See Barcode format

	serial{i}.errorCorrectionLevel	String(4)	Optional	Amount of redundancy or error correction data included in the code. There are four error correction levels available in QR codes. - Code options: L/M/Q/H
	noticeDesc	String(1024)	Required	Notice description * Long content is allowed. * See Additional Information format.
	csInfo	String(512)	Optional	CS Information Set value to JSON Object (<i>calls or emails or websites</i>) * See below for an example
	providerViewLink	String(512)	Optional	Link to additional information from the provider * See Links format
	privacyModeYn	String(1)	Optional	Whether or not to require user authentication when using the card Either 'Y' or 'N' * Default: 'N'
	bgColor	String(8)	Optional	Color of the card art. e.g., #00FFFF.
	fontColor	String(8)	Optional	Color of the font on the card art Acceptable values: <i>dark, light</i>
	blinkColor	String(8)	Optional	Color of the blinking effect in the Indicator area e.g., #00FFFF
	preventCaptureYn	String(1)	Optional	Flag whether this wallet card view prevents screen capture Either 'Y' or 'N' * Default: 'N'
	noNetworkSupportYn	String(1)	Optional	Sets whether to support to open the wallet card under 'No Network' Status. Either 'Y' or 'N' * Default: 'N'
	appLinkLogo	String(256)	Required	App link image URL. The file size should not exceed 256 kB.
	appLinkName	String(32)	Required	App link name.
	appLinkData	String(256)	Required	Information about the partner app link.
	locations	String(1024)	Optional	List of locations where the card can be used. * See Location format.

Example	<pre> "card": { "type": "generic", "subType": "others", "data": [{ "createdAt": 1661745824345, "updatedAt": 1661745824345, "language": "en", "refId": "REFID-012345", "attributes": { "title": "Samsung Generic Card", "subtitle": "Personal Members", "providerName": "Samsung", "startDate": 1661751274000, "startDate.utcOffset": "UTC+9", "endDate": 1761778000000, "endDate.utcOffset": "UTC+9", "endDate.relativeNotiTime": "1440", "text1": "13047623", "text2": "Silver", "text3": "Suwon Station Branch", "text4": "031)000-1235", "timestamp1": 1761751274000, "timestamp1.utcOffset": "UTC+9", "mainImg": "https://www.samsung.com/images/main.png", "image1": "https://www.samsung.com/images/image1.png", "image1.darkUrl": "https://www.samsung.com/images/dark.png", "image1.lightUrl": "https://www.samsung.com/images/light.png", "serial1.value": ">1180MM2241B7C 0000000000002980600000000000 0 090870907 ", "serial1.serialType": "QR CODE", "serial1.ptFormat": "QR CODE", "serial1.ptSubFormat": "QR_CODE", "bgColor": "#FF5000", "fontColor": "dark", "noticeDesc": " {\\"count\\":2,\\"info\\":[{\\\"title\\\":\\"NOTICE1\\",\\\"content\\\":["DESCRIPTION1\\","DESCRIPTION2\\"]},{\\\"title\\\":\\"NOTICE2\\",\\\"content\\\":["DESCRIPTION1\\","DESCRIPTION2\\"]}]}, "csInfo": "{ \"calls\": [{ \"key\": \"Emergency\", \"value\": \"82) 123-4567\" }, { \"key\": \"Customer Service\", \"value\": \"82) 123-9876\" }], \"emails\": [{ \"key\": \"CS Team\", \"value\": \"cs@atwsample.com\" }, { \"key\": \"FAQ\", \"value\": \"https://atwhomepage.com/faq\" }, { \"key\": \"Support\", \"value\": \"https://atwhomepage.com/support\" }] } }, "providerViewLink": " {\\"count\\":1,\\"info\\":[{\\\"link\\\":\"https://samsung.external.info.link/REFID-012345\\\",\\\"type\\\":\\\"web\\\",\\\"text\\\":\\\"See more information\\\"}]}", "appLinkData": "https://www.samsung.com/", "appLinkLogo": "https://www.samsung.com/logo.png", "appLinkName": "Samsung" }], "localization": [</pre>
---------	--

```
        "language": "ko",
        "attributes": {
            "title": "삼성 제네릭 카드",
            "subtitle": "개인 멤버스",
            "providerName": "삼성",
            "text2": "실버 등급",
            "text3": "수원역점"
        }
    }]
}
}
```

4. Implementing ATW button

For Wallet integration, you need to insert 'Add To Wallet' script into your system. We usually follow the procedures below to implement the 'Add to Wallet' button.

First, proceed with script composition with the sample script in [Partner portal](#) or refer to the [Integration sample code](#).

Second, create tokenized card data (Cdata) and insert the data token into the script above. Card data is the actual content data of wallet cards and has several formats based on card type. Please refer to [Generate_Cdata Sample Code](#) for detail.

Important: Cdata token should expire in 30 seconds after creation, so cdata token needs to be created right after users actually click "Add to Wallet" button.

To implement 'Add to Wallet' button, you may need some base data. You can find the base data and other necessary information on [Partner portal's Wallet card page](#).

4.1 Samsung Wallet on the Web

This section explains how to implement an 'Add to Wallet' button using JavaScript in a web view.

Web Button Reference with importing API Javascript

If you implement the 'Add to Wallet' button using this script, the button is shown only on the device that support Samsung Wallet.

To automatically parse <samsung:wallet> HTML tags on load, include the following standard JavaScript:

```
<script src="https://us-cdn-gpp.mcsvc.samsung.com/lib/wallet-card.js" type="text/javascript"></script>
```

You can use these tags or JavaScript functions for the web button if you're rendering HTML and you have proper partner permissions. You can also use the script by referring to the various attributes.

samsung:wallet HTML Tag

The 'samsung:wallet' namespace tag defines the placement and various attributes of the 'Add to Wallet' web button for Samsung Wallet.

```
<samsung:wallet  
cardid="CARD_ID" cdata="CDATA" partnercode="PARTNER_CODE" buttonid="BUTTON_ID"  
rdclickurl="RD_CLICK_URL" rdimpressionurl="RD_IMPRESSION_URL"  
></samsung:wallet>
```

- Button Attributes

Attributes	Type	Required	Description
cardid	String	Y	Wallet card identifier * the value granted from partner portal
cdata	String	Y	Encrypted Card object (JSON). * This field needs to be encrypted. * See Security .
partnercode	String	Y	Partner Code. * the value granted from partner portal
buttonid	String	Y	DOM Element ID for the 'Add to Wallet' web button's for Samsung Wallet.
buttontype	String	N	'Add To Wallet' button type. ["btnSW" / "btnATSW", default : btnSW] * See Image Resources
inline	String	N	Display one-line of 'Add To Wallet' image button. (default: true (one-line))
locale	String	N	locale of 'Add To Wallet' image button. * See Image Resources
rdclickurl	String	Y	URL of logging a button click event. * the value granted from partner portal
rdimpressionurl	String	Y	URL of logging a button impression event. * the value granted from partner portal
showforced	String	N	Flag to show "Add To Wallet" button forced. (default : false)
mediatheme	String	N	Load button's resources from media theme policy. ["default" / "inversion" / "lightonly" / "darkonly", default: default] *default : load button's theme by prefers-color-scheme policy *inversion : load inversion of default button's theme. *lightonly : load light theme of default button. *darkonly : load dark theme of default button.
style	String (CSSStyleDeclaration)	N	Load button with custom style
target	String	N	Option to choose button's target name * default : "WALLET"
onshowbutton	Function	N	Callback handler function for the button's on-show event.

onclickbutton	Function	N	<p>Callback handler function for the button's on-click event.</p> <p>If you register the handler function, you must return a callback or promise value.</p> <p>* See Usage of onClickButton Handler</p>
---------------	----------	---	---

samsungWallet.addButton function

This function allows you to explicitly render the Samsung Wallet API for the 'Add to Wallet' web button.

```
samsungWallet.addButton({
  cardId: "CARD_ID",
  cdata: "CDATA",
  partnerCode: "PARTNER_CODE",
  targetId: "TARGET_ID",
  buttonId: "BUTTON_ID",
  RDClickUrl: "RD_CLICK_URL",
  RDImpressionUrl: "RD_IMPRESSION_URL",
})
```

- Button Attributes

Unlike samsung:wallet HTML Tag case, you must use camelCase in the button attributes in function case.

Attributes	Type	Required	Description
cardId	String	Y	Wallet card identifier * the value granted from partner portal
cdata	String	Y	Encrypted Card object (JSON). * This field needs to be encrypted. * See Security .
partnerCode	String	Y	Partner Code. * the value granted from partner portal
targetId	String	Y	DOM (Document Object Model) Element ID to place the 'Add to Wallet' web button for Samsung Wallet.
buttonId	String	Y	DOM Element ID for the 'Add to Wallet' web button's for Samsung Wallet.
buttonType	String	N	'Add To Wallet' button type. ["btnSW" / "btnATSW", default : btnSW] * See Image Resources
inline	String	N	Display one-line of 'Add To Wallet' image button. (default: true (one-line))
locale	String	N	locale of 'Add To Wallet' image button. * See Image Resources

RDClickUrl	String	Y	URL of logging a button click event. * the value granted from partner portal
RDImpressionUrl	String	Y	URL of logging a button impression event. * the value granted from partner portal
showForced	String	N	Flag to show “Add To Wallet” button forced. (default : false)
mediaTheme	String	N	Load button's resources from media theme policy. [“default” / “inversion” / “lightonly” / “darkonly”, default: default] *default : load button's theme by prefers-color-scheme policy *inversion : load inversion of default button's theme. *lightonly : load light theme of default button. *darkonly : load dark theme of default button.
style	Object (CSSStyleDeclaration)	N	Load button with custom style
target	String	N	Option to choose button's target name * default : “WALLET”
onShowButton	Function	N	Callback handler function for the button's on-show event.
onClickButton	Function	N	Callback handler function for the button's on-click event. If you register the handler function, you must return a callback or promise value. * See Usage of onClickButton Handler

Usage of onClickButton Handler

You can choose whether to proceed with the next ‘Add To Wallet’ step using a Promise or a callback function, if you register a callback handler in onClickButton. The function' parameters are defined as following. We recommend that adding the process of generating JWT cdata (add cdata to options.cdata) in this handler because of the cdata expiration time.

Attributes	Type	Required	Description
options	Button Attributes	N	Attributes of the current button.
callback	Function	N	Callback function to pass the flag to proceed (default : false)
(Promise resolve)	Function	N	Promise resolved value to pass the flag to proceed (default : false)

2-1. Callback to web button process from callback attributes. (under ES5)

- By executing a callback function with flag, you can proceed to the next 'Add To Wallet' process.

```
onClickButton: function (options, callback) {
    // TODO partner's process
    callback(flag)
}
```

2-2. Callback to web button process from returning Promise. (ES6)

- By returning a promise with resolving flag, you can proceed to the next 'Add To Wallet' process.

```
onClickButton: async (options) => {
    return new Promise(async (resolve, reject) => {
        // TODO partner's process (await)
        resolve(flag)
    })
}
```

Important: Cdata token should expire in 30 seconds after creation, so cdata token needs to be created right after users actually click "Add to Wallet" button.

4.2 Samsung Wallet on the App

The following outline explains how to implement 'Add to Wallet' button on native app.

1. Get button graphic resources from a repository depending on your service environment.
 - A. See [Image Resources](#).
2. Check availability by calling API 'Check service available devices', then determine whether to show up 'Add to Wallet' button on the user device.
 - A. If 'available' on response has 'true' -> SUPPORT
 - B. If it has 'false' -> NOT SUPPORT
3. Implement JWT web link on the button triggered action

App Button Reference

App Button on Android

Sample code implementation

```
public class WalletCodeSample {
    protected final static String TAG = "SamsungWalletSample";
    protected static final String HOST = "https://api-us3.mpay.samsung.com";
    protected static final String PATH = "wallet/cmn/v2.0/device/available";

    /**
     * sample entry point of the usage
     */
    public static void main() {
        Executors.newSingleThreadExecutor().submit(() -> {
```

```

final String modelName = Build.MODEL;
final String countryCode = null; // (optional) country code (ISO_3166-2)
final String serviceType = "WALLET"; // (Required, fixed) for Samsung Wallet
final String partnerCode = null; // (Required)

try {
    WalletCodeSample sample = new WalletCodeSample();
    boolean isWalletSupported = sample.checkWalletSupported(
        modelName, countryCode, serviceType, partnerCode);
    String msg = String.format(
        "query for model(%s), countryCode(%s), serviceType(%s), partnerCode(%s) / wallet supported? (%s)",
        modelName,
        countryCode,
        serviceType,
        partnerCode,
        isWalletSupported);
    Log.d(TAG, msg);
} catch (Exception e) {
    // failed to check due to some reasons
    Log.e(TAG, e.getMessage(), e);
}
});

/**
 * please refer to the Wallet API Spec document > '6.6 Check service available devices' for more details
 *
 * @return true if wallet supported, otherwise false
 * @throws Exception throws exception when it's not possible to get status due to any reasons
 */
public boolean checkWalletSupported(@NonNull String modelName, @Nullable String countryCode,
    @NonNull String serviceType, @NonNull String partnerCode) throws Exception {
    if (modelName == null || modelName.isEmpty()) {
        Log.e(TAG, "modelName is Required parameter");
        throw new Exception("something went wrong (failed to get device model name)");
    }
    if (serviceType == null || serviceType.isEmpty()) {
        Log.e(TAG, "serviceType is Required parameter");
        throw new Exception("something went wrong (failed to get device serviceType)");
    }
    if (partnerCode == null || partnerCode.isEmpty()) {
        Log.e(TAG, "partnerCode is Required parameter");
        throw new Exception("something went wrong (failed to get device partnerCode)");
    }

    String urlString = makeUrl(modelName, countryCode, serviceType);
    Log.i(TAG, "urlString: " + urlString);

    try {
        URL url = new URL(urlString);

        HttpURLConnection connection = (HttpURLConnection) url.openConnection();
        connection.setRequestProperty("partnerCode", partnerCode);
        connection.setRequestMethod("GET");

        int responseCode = connection.getResponseCode();
        Log.i(TAG, "responseCode: " + responseCode);

        BufferedReader bufferedReader;
        if (responseCode == 200) {
            bufferedReader = new BufferedReader(new InputStreamReader(connection.getInputStream()));
        } else {
            bufferedReader = new BufferedReader(new InputStreamReader(connection.getErrorStream()));
        }

        StringBuilder sb = new StringBuilder();
        String inputline;
        while ((inputline = bufferedReader.readLine()) != null) {
            Log.i(TAG, inputline);
            sb.append(inputline);
        }
        connection.disconnect();
        bufferedReader.close();
    }
    // parse result
}

```

```
JSONObject jsonObject = new JSONObject(sb.toString());
String resultCode = jsonObject.getString("resultCode");
String resultMessage = jsonObject.getString("resultMessage");
if ("0".equals(resultCode) && "SUCCESS".equals(resultMessage)) {
    return jsonObject.getBoolean("available");
} else {
    throw new Exception("something went wrong, resultCode(" + resultCode + "),
                        resultMessage(" + resultMessage + ")");
}
} catch (IOException e) {
    Log.e(TAG, e.getMessage(), e);
    throw new Exception("something went wrong (IOException), " + e.getMessage());
} catch (JSONException e) {
    Log.e(TAG, e.getMessage(), e);
    throw new Exception("something went wrong, receive wrong formatted response, " + e.getMessage());
}
}

protected String makeUrl(@NonNull String modelName, @Nullable String countryCode,
                        @NonNull String serviceType) {
    StringBuilder sb = new StringBuilder();
    sb.append(HOST).append('/');
    sb.append(PATH);
    sb.append('?').append("serviceType").append('=').append(serviceType);
    sb.append('&').append("modelName").append('=').append(modelName);

    if (countryCode != null && !countryCode.isEmpty()) {
        sb.append('&').append("countryCode").append('=').append(countryCode);
    }
    return sb.toString();
}
}
```

Important: Cdata token should expire in 30 seconds after creation, so cdata token needs to be created right after users actually click “Add to Wallet” button.

4.3 Samsung Wallet on the MMS/Email

The following guides how to configure wallet code on Email and MMS.

1. Implement the process [Data Fetch Link](#) including Server APIs. You need to create a unique 'reference ID'. Please make sure 'reference ID' is complex enough, so that no important information can be inferred for security reasons.
2. Deliver messages including the web link via message platforms. For MMS, the designed link will show up as 'Smart Suggestion' on Samsung Devices. You can find a sample web link on **Partner portal's Wallet card page**.
3. As for card data, Samsung wallet will ask the partner system to provide card detail through server API.
4. Duplicate requests will be prohibited on the same device.

Link to 'Add to Wallet' on the MMS/Email

You can add an ‘Add to Wallet’ web button even in environments where the JavaScript API cannot be loaded, such as SMS or e-mail.

* These methods do not support controlling 'Add to Wallet' button visibility.

MMS Link

URL Link: [URL](#)

Attributes	Type	Required	Description
URL	String	Y	Add to wallet link URL * See Data Transmit Link * See Data Fetch Link

Email on Web Button Link

```
<a href="URL">
  
  
</a>
```

Attributes	Type	Required	Description
URL	String	Y	Add to wallet link URL * See Data Transmit Link * See Data Fetch Link
IMAGE_URL	String	Y	Button's image resource URL * See Image Resources
RD_IMPRESSION_URL	String	Y	Logging URL for impression * the value granted from partner portal

4.4 Statistics Service

Samsung Wallet serves useful statistics data of the integrated service on partner portal. The data configured by making simple API calls for each event such as button impression and click. These are necessary to provide better services as well.

Statistics API sample code: You can find actual code sample at wallet script guide section in '[Wallet Cards' menu on the partner portal](#).

[Event Notification API]

[https://us-
rd.mcsvc.samsung.com/statistics/{event}/addtowlt?{parameters}&utm_source=partner&utm_medium={channel}](https://us-rd.mcsvc.samsung.com/statistics/{event}/addtowlt?{parameters}&utm_source=partner&utm_medium={channel})

{event}:

For each event in situations:

- **impression:** When 'Add to Wallet' button has been shown
- **click:** When 'Add to Wallet' button has been clicked

{parameters}:

Includes key factors to figure out the service.

{channel}:

- **app:** Samsung Wallet button on the App
- **web:** Samsung Wallet button on the Web
- **email:** Samsung Wallet button on the Email

For details, please visit 'Wallet Cards' menu on the partner portal.

5. Server interaction

The added users' cards allow updating its data using server interactions. Find the card details to configure API on Partner portal if partners want to manage the added cards.

- 1) Samsung server will notify the result of 'Add to Wallet' via [Send Card State](#).
- 2) Partners get the callback URL for Samsung Server API from [Send Card State](#) payload.
- 3) Using the callback URL, partners can make actions for the added cards via [Samsung Server API](#).
- 4) Depending on the interfaces, Samsung Server triggers specific operations. For example, when [Update Notification](#) is called, Samsung server calls partners' server to look up the updated contents.

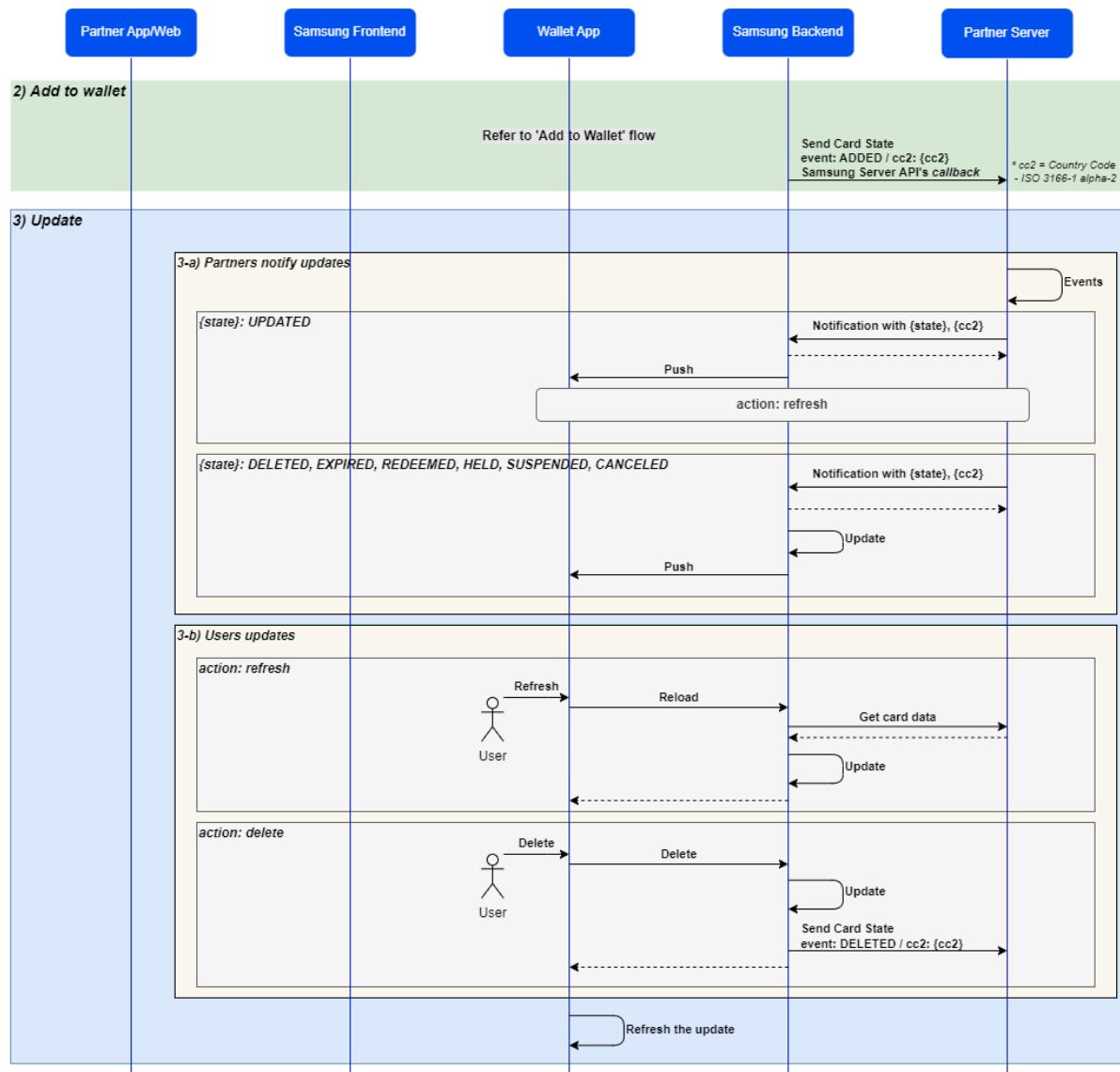


Figure 5-1 Server Interaction

5.1 Partner Server API

Samsung server can call the following API by using endpoint on the registered card information.
If the partner server manages an inbound allow list, contact us to register Samsung server IP address.

Get Card Data

Returns the current information of the card.

[Request]

Type	Value			Description
Method	GET			
URL	{Partner server URL}/cards/{Card Id}/{refId}?fields={fields}			
Headers	Authorization	String(1024)	Required	Credential token. The token can have prefix "Bearer" as an authorization type. e.g., Bearer <credentials> * See Authorization Token .
	x-request-id	String(32)	Required	Request identifier. Randomly generated UUID string.
Path Parameters	Card Id	String(32)	Required	Wallet card identifier * Refer to the 'Add to Wallet' Interfaces
	refId	String(32)	Required	A unique content identifier defined by the content provider
Query Parameter	fields	String(128)	Optional	Attributes which intended to retrieve. Can be specified using commas(,) as separators. e.g. <i>balance,barcode</i>
Payload	N/A			
Example	GET /cards/12584806754/ref-20230304-0003			

[Response]

Type	Value			Description
HTTP Status	200 OK 204 No Content			
Payload (Option1)	cdata	String	Conditional	Card object. (JSON) * This field needs to be encrypted. * See Security .
Payload (Option2)	card	Object	Conditional	Card information * Card object as an alternative to <i>cdata</i> * If cards include sensitive data, highly recommend using <i>cdata</i>
	card.type	String(16)	Required	Wallet Card type. * See Wallet Cards
	card.data[]	Array of Object	Required	Wallet card data container
	data[].refId	String(32)	Required	A unique content identifier defined by the content provider
	data[].createdAt	Long(13)	Required	Timestamp of data. Epoch timestamp in milliseconds.
	data[].updatedAt	Long(13)	Required	Timestamp of data. Epoch timestamp in milliseconds.
	data[].state	String(16)	Required	Wallet card state e.g., ACTIVE, UPDATED, EXPIRED, REDEEMED, HELD, DELETED, CANCELED, PENDING * See Card States for details
	data[].language	String(8)	Required	Default content language code. e.g., en, ko
	data[].attributes	Object	Required	Card data attributes.
	data[].attributes.{fields}			Attribute fields by card type. * See Wallet Cards
Localization	data[].localization[]	Array of Object	Optional	Information for multilingual support.
	localization[].language	String(8)	Required	Multilingual content language code. e.g., en, ko
	localization[].attributes.{fields}			For displaying a given language, 'data[].attributes' can be replaced by localized versions. * See Wallet Cards

Example (Option1)	<pre>{ "cdata": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4gRG9lIiwiaWF0IjoxNTE2MjM5MDIyfQ.SflKxwRJSMeKKF2QT4fwpMeJf36P0k6yJV_adQssw5c" }</pre>
Example (Option2)	<pre>{ "card": { "type": "ticket", "subType": "movies", "data": [{ "refId": "ref-20230304-0002", "createdAt": 1612660039000, "language": "en", "attributes": { "title": "Samsung Wallet", {*Refer to Wallet Cards} }, "localization": [{ "language": "ko", "attributes": { "title": "삼성월렛" } }] }] } }</pre>
Example (filtered using <i>select</i> parameter)	<p>GET /cards/12584806754/ref-20230304-0003?select=idPhoto</p> <pre>{ "card": { "type": "ticket", "subType": "entrances", "data": [{ "refId": "ref-20230304-0003", "createdAt": 1612660039000, "language": "en", "attributes": { "idPhoto": "idPhoto data" } }] } }</pre> <p>OR</p> <pre>{ "cdata": tokenize{data} }</pre>

[Result]

HTTP Status Code	Description
200 OK	Success
204 No Content	Card doesn't exist *The Card will be removed in the wallet service
400 Bad Request	Requests cannot or will not be processed the request due to something that is perceived to be a client error
401 Unauthorized	Authorization token is invalid or expired
500 Internal Server Error	Server encountered an unexpected condition that prevented it from fulfilling the request
503 Service Unavailable	Server is not ready to handle the request

Send Card State

Partners can manage the state or history of the card using this API.

If the Card state is changed on the Samsung device, Samsung calls this API using a refId.

[Request]

Type	Value			Description
Method	POST			
URL	{Partner server URL}/cards/{Card Id}/{refId}			
Headers	Authorization	String(1024)	Required	Credential token. The token can have prefix "Bearer" as an authorization type. e.g., Bearer <credentials> * See Authorization Token .
	x-request-id	String(32)	Required	Request identifier. Randomly generated UUID string.
Path Parameters	Card Id	String(32)	Required	Wallet card identifier * Refer to the 'Add to Wallet' Interfaces
	refId	String(32)	Required	A unique content identifier defined by the content provider
Query Parameters	cc2	String(2)	Required	Country code (cc2) for Samsung Server API
	event	String(16)	Required	Events on wallet card e.g., ADDED, UPDATED, DELETED * See Card States for details
Payload	callback	String	Optional	Callback URL for Samsung Server API

Samsung Wallet Cards – API Specification

Example	POST /cards/12584806754/ref-20230304-001?cc2=us&event=ADDED { "callback": "https://us-tsapi.walletsvc.samsung.com" }
---------	---

[Response]

Type	Value	Description
HTTP Status	200 OK	
Payload	N/A	
Example	200 OK	

[Result]

HTTP Status Code	Description
200 OK	Success
401 Unauthorized	Authorization token is invalid or expired
500 Internal Server Error	Server encountered an unexpected condition that prevented it from fulfilling the request
503 Service Unavailable	Server is not ready to handle the request

5.2 Samsung Server API

Partners can notify their contents changes with the following API.

[Service Domain]

Environment	Domain
Public domain	https://tsapi-card.walletsvc.samsung.com
Private domain	'callback' field from Send Card State API request payload

The domains can be selectively used depending on your service requirement.

- If the service needs to register static IPs on your system, we recommend using Private domain. In this case, use the domain received in the request 'callback' field from [Send Card State](#) API.
- If the service does not require IP registration, Public domain can be a good choice. In this case, country code(cc2) is required as a path parameter.

* To configure integration for each environments, register a new card service and get new Card ID.

* To guarantee safe communication, servers should configure Token-based Authentication. See [Authorization Token](#) for the details.

Update Notification

If wallet card data content is updated, send a notification to the Samsung server.

[Request]

Type	Value			Description
Method	POST			
URL	{cc2}/wltx/cards/{Card Id}/notification			
Headers	Authorization	String(1024)	Required	Credential token. The token can have prefix "Bearer" as an authorization type. e.g., Bearer <credentials> * See Authorization Token .
	x-smcs-partner-id	String(32)	Required	Partner ID.
	x-request-id	String(32)	Required	Request identifier. Random generated UUID string.
Path Parameters	cc2	String(2)	Conditional	Country code (cc2) from Send Card State * Required if using Public domain

Samsung Wallet Cards – API Specification

	Card Id	String(32)	Required	Wallet card identifier granted from Partner Portal
Payload	card	Object	Required	Wallet card object
	card.type	String(16)	Required	Wallet card type. *See Wallet Cards
	card.data[]	Array of Object	Required	Wallet card data container
	data[].refId	String(32)	Required	A unique content identifier defined by the content provider
	data[].state	String(16)	Required	Wallet card state e.g., ACTIVE, UPDATED, EXPIRED, REDEEMED, HELD, DELETED, SUSPENDED, PENDING * See Send Card State for details
Example	POST /wltx/cards/12584806754/notification [Headers] Authorization: eyJjdHkiOiJBVVRIIwidmVyljoxLCJwYXJ0bmVySWQiOlxMjg10... x-smcs-partner-id: partner-id-0001 x-request-id: req-202303140003 [Payload] { "card": { "type": "ticket", "data": [{ "refId": "ref-20230304-0003", "state": "UPDATED" }] } }			

[Response]

Type	Value	Description
HTTP Status	200 OK 204 No Content	
Payload	N/A	
Example	200 OK	

[Result]

HTTP Status Code	Description
------------------	-------------

Samsung Wallet Cards – API Specification

200 OK	Success
204 No Content	Card doesn't exist *The Card will be removed in the wallet service
400 Bad Request	Requests cannot or will not be processed the request due to something that is perceived to be a client error
401 Unauthorized	Authorization token is invalid or expired
500 Internal Server Error	Server encountered an unexpected condition that prevented it from fulfilling the request
503 Service Unavailable	Server is not ready to handle the request

Cancel Notification

If a cancelation happens for events such as performances, sports, movies, and journeys, partners can send a notification about it and set all of the related cards to expire.

*** This API does not support updates for specific attributes on the card.**

[Request]

Type	Value			Description
Method	POST			
URL	{cc2}/wltx/cards/{Card Id}/cancellation			
Headers	Authorization	String(1024)	Required	Credential token. The token can have prefix "Bearer" as an authorization type. e.g., Bearer <credentials> * See Authorization Token .
	x-smcs-partner-id	String(32)	Required	Partner ID.
	x-request-id	String(32)	Required	Request identifier. Randomly generated UUID string.
Path Parameters	cc2	String(2)	Conditional	Country code (cc2) from Send Card State * Required if using Public domain
	Card Id	String(32)	Required	Wallet card identifier granted from Partner Portal
Payload	card	Object	Required	Wallet card object
	card.type	String(16)	Required	Wallet Card type. *See Wallet Cards
	card.data[]	Array of Object	Required	Wallet card data container
	data[].eventId	String(32)	Conditional	Required if card.type has been set as 'ticket'.
	data[].vehicleNumber	String(32)		Required if card.type has been set as 'boardingpass'.
	data[].estimatedOr ActualStartDate	Long(13)		
	data[].state	String(16)	Required	Wallet card state e.g., CANCELED. * See Card States for details

Example	<pre> POST /wltx/cards/12584806754/cancellation [Headers] Authorization: eyJjdHkiOiJBVVRIIwidmVyIjoxLCJwYXJ0bmVySWQiOiIxMjg1O... x-smcs-partner-id: partner-id-0001 x-request-id: req-202303140004 [Payload] * A movie ticket has been canceled. { "card": { "type": "ticket", "data": [{ "eventId": "event-722164a1a7", "state": "CANCELED" }] } } </pre>
---------	---

[Response]

Type	Value	Description
HTTP Status	200 OK	
Payload	N/A	
Example	200 OK	

[Result]

HTTP Status Code	Description
200 OK	Success
204 No Content	Card doesn't exist *The Card will be removed in the wallet service
400 Bad Request	Requests cannot or will not be processed due to something that is perceived to be a client error
401 Unauthorized	Authorization token is invalid or expired
500 Internal Server Error	Server encountered an unexpected condition that prevented it from fulfilling the request
503 Service Unavailable	Server is not ready to handle the request

6. Security

The following contents describe how to generate JWT (JSON Web Token). It follows *RFC 7519* specification. For more details, refer to '<https://jwt.io>'

6.1 Card Data Token

For secure data inter-communication, the token must be encrypted and signed using security factors. See the chapter *Security factors* for details.

JWT Details

JWE Format

[JWE Header]

JWE Header	Requirement	Description
alg	Required	Cryptographic algorithm used to encrypt the Content Encryption Key (CEK). e.g., <i>RSA1_5</i>
enc	Required	Content encryption algorithm used to perform authenticated encryption on the plaintext to produce the ciphertext.

[JWE Payload]

JWE Payload	Requirement	Description
encrypted_key	Required	Contains the BASE64URL (JWE Encrypted Key) value. The Content Encryption Key is encrypted with the Public Key.
iv	Required	Contains the BASE64URL (JWE Initialization Vector) value. Initialization vector used in the encryption algorithm.
ciphertext	Required	Ciphertext value resulting from authenticated encryption of the 'cdata' object, which is encrypted using 'encrypted_key' and 'iv'.
authentication tag	Required	Contains the BASE64URL (JWE Authentication Tag) value, used for verifying the integrity of the ciphertext.

[JWE Example]

```
BASE64URL (UTF8 (JWE Header)) + '.' +
BASE64URL (JWE encrypted_key) + '.' +

```

BASE64URL (JWE iv) + '!' +
 BASE64URL (JWE ciphertext) + '!' +
 BASE64URL (JWE authentication tag)

JWE Header

```
{"enc":"A128GCM","alg":"RSA1_5"}
```

JWE Payload: ciphertext

Refer to sheets on '[Add to Wallet' interfaces, and Wallet Cards](#)

Result

```
eyJraWQi0iJXTFQuUFVCS0VZIwiWz5Ji0j0iTQTEy0EdDTSiSImFsZyI6I1JTQTFNSJ9.ABO_Ci81BtJ2d1a8TCgkFwBx9WpRI4TkhhZwmS8swct_2nNZHasI_nKlmj3wn
KM5gwaoUny14ZXC_6EoZhJ6TdIcIuQ-raIRs6woEsu8XA2dTsC5117wu9WdsgOK4anJ0K1UNII4PLeR3d-4foX1Hx1fok9sIwWqqFq14vnqg3hE-
i4J6cyWOYBphzNYBMKyyNkIqFc161bTTEhc4TDAOrPKWra3VMB0BBz5NyfZ1axzFk-
17tZ0GfhS82A7GL9REj1k5B10_2qfGmhTtfFVcyYTMKv3InMaHQ0b48L3SK1oPPmFCuqigYMLUDbg_QWdnB19eI1n0jjt8Ar2NUA.ZviyGHUSi5Fb2R12.gm5ivizrQQ
dr8NPK12qrEyaI4MD-FISfwtBBBgEbhnjhMnu-c_o1YUYRvdhCm0ki_rvcDNZKDLcP_g7shSkMroYin3bI92qgtkf2V4Y-
kCuG2DvgV9UIV3oxawl1IKfcNtmZ1z3tKhV_FUE77rNrbrwf2XVMiwsQo5b01moUsKbhHuHasq1lRE0rtC1fgn03qFE_E-B87vh75En2PnBYDJv-
6_8g3aesSyodVHydYaOnLxW_KWqif-i5AUwFiIgK5LgVmUz9DS16-QKgYiz5p19nyydJjjpIlib tuaLYvz1CH-gskwEUhM162ZR-
Chz2Ado8Vn0SRoCcJhcaX6PBS3x6fhyXhr65BjZAn41Mdf5SKN92bcFcyLx83_pgRM4VUI_-Kx1lwpaRkwRtyxmEbkMJ-
2w8NUMRNnpGt2ERLo_hvtZ8Xh1kopvqjLdjdg_Qqfu_OEW03HvuNKGQeu3QhI6EyWvbarb70ZsIsz-F9507K-KQtJhfbWz_YrA2NxD2Bcgc9uA966_9uQ4oMBwA-
8fccAWPxYYU4vZBz_ycv25j8GrdqHHtW6n9TK2y4Nu07jt4CcoFvU5N_Gsyn1qoWD11-_1Mk8aMF-L5dDipvrun7Dca1JD8Me4NsAAkESLQfKz_sddSu0-
05icfKm33quQP6fzn50cY6Dmn5KBVQXZHgHCG_A_KlxQ0Q1KsPL4JSAxCNCiUyVPTDqbc0HxxUiYrM3tCde6piCymGCPK205Ny1D-
6eN43dI4yK029YQx8rSLDOAA6RWvp-ZEHdKxyNyI18A-8Fw4i0qp6V9K9AjZ5T-ajqDK1h6n_opt-ZCjkJzz-7r2x07BSa_5ng7iwamBRsv1defxhIyV-EsCe1MEif-
_na_411h6pJA-GwCzp_w5wlqra0RPDq2081Y70XPPu4H_3EH_6a8cy5YHns93VfU00NSofN1Ker25zwNID39zoijy_de9gZjaWxa3k0TPrPn5MfdpXvTD0-
-ro4oqI34ab62-rUBCdYdsmTGGIHYZ3sHLGtyAfrBzHMPMDKAU0J9BUlRaSpqnr4NaHfq_S1m1Uy5pEEQ3j0b0zmc4uAsNQonrrreLqm8bKFQI41GjRJm9UvkCR-
PMFOnShEoqjmce6ZKua1qTOEFXYcd0ejBjqDcyCNqQGuGoDlnq2-3MkGGrpVQVYAO10MyKC_S18kpDvjXntKggzQ9B9vn1nQ9_fY3hmyor0ZEExyTajfua-
4I1sFKG3crQkx3scCSp-W6rf7VfzX5vDhqbHFZbhbuXYpFj1bDmIs_w-
xQDv1Kgb1zMw9gRbwM2Mk8t9QpzinNHcaQfv2dQaGQth4VycUccq0mJCS5QgnBwdiGGVxfk7BhwHK2jrw3K4egjQna9LSSNhCjhQZ69M16iVBffktnz5oT0L-
nPKCQEIfia-rjmwYy68EodZ18S75414YvLmjjiDuJkxB7zS0USVrvizp1jMk1RsbgwW7rf0JlcI9ed-
mHpsMxvepj2UxEZUX95Z_vx7i8xGSZxm1WrmSI0EEPA5TL7GQfxfIMtV4V_08RjIIPQtDJKMKENKF1INVkn8Wio7nosfyAk1gp1xkPR2SCIE1TCIRPweu_4y56Yq3wxvNwnc
AX7yyjyTuBIRk30zNw840MYY1JC67wntyBeqiq3TY1vz6Wxraenn_dlwIkru-RY_bvC98JwZPgDnpZdTdpnDhiJLeYk9zhZcwvJhvom7MS4CLjs6NdqvM9yeilkk55H-
E3n22-1n1u6pMjeYFBvTyzFr57sidCTsEEJmhb1QdSK3RpFxv2H0DYY90k7uAsdSifWhCKMue4QyUrd34WvZhjjAPRxqeOjfaVz-Vt-
331jVaJGKzIifMPuYFck-kYRqbQ193Uy9SY6-7ReId3C7GY1P9-TFSUgoBPUlNyDiEAmtoQ1izaepwJKjqS9Lj0Fd00jhxbqzy0-
spItgLnbMtuxpxQH6phH34vdB2FcgitTC8h1vp3_A0LvxxzuMdU3Jypc91tqMXxG7XZ4h4U0rvIS2q3XQwUB1uT09SyhfnPFf16h0-U8Bqd0FG-YYbA-
QN_AWN4UFS2FTFY-7yD1isp0G31LIFmORpLeHz0pcgEf0-
MayacxsVGIoptn67EnMwe_GrdwKzV27DeoczTmCn_Fb7QvTDsLE881RFz7LrhMiTUITdo4E0FkWuaZ1CoHrBpBhZmG30tLjbxYdB-
1rq74rXfdC1eOBjvPcdAxomyA9EOXCNt70ttI16FR31xjdYSGQv_iHtfkdaumMFYOjih_W9ZagIwB2uxMhNG0A3MpT8R80HZBPpvH3Hsb2uezWw8AqTlmkAlqF0g6nZqM
181z46gNKZ7w3h8a29-
ycI0yPz_m0POFIhnWjrEndjkew6aZoDEHwyUPsn07y930Dc8kHhPzb84bkAhBC25Ye8wGvgMrhFIwSigfht_G3M8Nlt3vfAsQe98two0Tzu3K72Kmod8khwd6Xq60a1Xob
A1M9wf151Wmjji8yr4T-7pqdC510mbxSQuRA00-6Puja5DUFU0Q3yZM0iWR1YJciqAoFp-XWN9CrH287vJzhW2s4Ges8S-
Wuda9yu61u3b1pwR0fysEUzQuaY_t3qkZiaGhvZ0A2nEFdY2wkTmaonidqtsku8rhPKnqaLRC_ydnvyQ00xnrdwJRxILuTVlaaQmYgT10zesSRvpkH4InkIU0ikBDCeQ
VnLC3qJuYMcm5u_DTIc-pb7E9H4zWxm3TA1MLz0C-v1u0shzaqo3tvIXA9uy913qvPz1reALwg7w1yqQuhPd-
6PGolbdddFqWXEkba3JtRy3wnxJICZCGZoqwiUpdHpm0CyzfLx71cBcpvC3Lkg_pDUWKb2qjV2HjodUSvStv8.bv9p-aoAIT1mfIJswZevSg
```

JWS Format

JWS Header

JWS Header	Requirement	Description
alg	Required	Cryptographic algorithm used to generate signature e.g., RS256

JWS Header	Requirement	Description
cty	Required	Payload content type Set as 'CARD'
ver	Required	Token version. Set as 2.
partnerId	Required	Partner identifier.
utc	Required	Creation time. To prevent repeated use, the token expires after a certain period of time. Unix timestamp in milliseconds. *Time offset from UTC of +00:00

[JWS Payload]

JWS Payload	Requirement	Description
JWE compact serialization	Required	Contains BASE64URL (JWE) value.

[JWS Signature]

JWS Signature	Requirement	Description
JWS Signature	Required	BASE64URL (Signature of (BASE64URL(UTF8(JWS Header)) + '.' + BASE64URL(JWS Payload))))

[JWS Example]

```
BASE64URL (UTF8 (JWS Header)) + '.' +
BASE64URL (JWS Payload) + '.' +
BASE64URL (JWS Signature)
```

JWS Header:

```
{"cty":"CARD","ver":2,"partnerId":"1234567890","utc":1631776245876,"alg":"RS256"}
```

JWS Payload:

JWE Result

Result:

```

eyJidHkiOiJQQVNTIwidmVyljoxLCJwYXJ0bmVvSWQiOixMjM0NTY3ODkwiwidXRjIjoxNjM10DQ10DU2MjQ0LCJhbGciOiJSUzI1NiisImtpZCI6IIBUTi5QUklRLVkfQ.ZX1K
cmFXUWlPaUpYVEZRdVVGvKNTMFzaSw13aVpXNlpJam9pUVRFerU9FZERUU01zSw1Gc1p5STZjbEpUVRGZk5TsJkuQJPX0Np0DFCdfEoyZDFh0FRDZ0tmV0J40VdwUkkr0VG
toSFp3bVM4c3djf8yb5kaSEfzSV9uS0tajnNbktNNw3dYn9VbnkxNFpYXzZfb1poSjZUZE1pY1VRLXjhSVJzNndvRVN10FbBmnRUMXNDNwxN3d10Vdkc2dPSzrhbkow
S01VTk1JNFBMViZC002m9YUh4MwZvaz1zSpxcFGClw0dm5xZzNorS1pNEo2Y31XT1CcGH6T11CTU5e05rSXFGY3psNmxiVFRFaGMbVERBT3QS1dyYTNTUWQk
J6Nu55ekYxXh6RmstMTd0WjBHzmhT0dJBN0dMOVFajfLNUiXMF8ycwZhbWhUdgZGvMn5wVRNS3YzSw5NUyHrmgi00EwzU0sxb1BqUZDdXFpZ11NVkxVRGJnX1FXZG5C
bd11Swxbk9qanQ40X1yT1VBL1p2aX1HSFVtaTVGyj3SbdIuZ281axZpenJRURSOE50SzF0mF5R1BSTRNRC1GSVm1v1RC0kjRNwoTmhqTw51LWNfbzFZVV1SdmRoQ2
0wS21fcnZjRE5aS0RMY1BfZzdaFnRtVjwWluM2JJOTjxz3RrRmgvJrZLwtDdUcyRHZhvjlVSvYzb3hhV3zSutmY050bVppWmozVghWx0ZVRTdKck5yYndmM1hNvk13
c1FvNWlwibG1vVXNLYkh1SGFzcU1sUkUwUnrDMwZnbjAzcUzFx0UtQjg3dmh0NUVJm1BuQ11EsNytn184ZzNhZNTeW9dkh5e11hwu9ThhXX0tXcw1GLWk1QVV3rm1z0
s1TGdWbV6OURtU0tWW16NXB055eRkmwSwxpYnR1YUx2dnpcMCUNILwdz3dFVwhpTw2M1pSLUNjRd1J0WxhtRWJLTuotMnc4T1vNUk5ucEd0MkvSTG9faZ0WjhYaDrb3B2cWpMZGpkR1
NjVCs1pBbjRsTRmU3NLTjkyYmGeWNWWDhKX3BnKuxNzVSVS1Tz3gbxFdwQXjRd1J0WxhtRWJLTuotMnc4T1vNUk5ucEd0MkvSTG9faZ0WjhYaDrb3B2cWpMZGpkR1
9rcwzVX09FV082SHZ1tkHuvV1M1FoS7FeVd2YXj1n09ac01ze160tPVN0stS1F05mhm01d6x11y0tj0e0y0mnnYz1QTK2N185dV0b0b1Cd0t0Ezjy0FXUHhZwv0
dlpCe195Y1YyNw04R3JkcUhIVCh2bj1lus1p5NE51MDdqSVQ0Y0NvR1Z1Nus5fr3N5bjFxb1dEMTETx2NazhhTUytTDVkrGlwdnJ1bjdERWFsSkQ4TNU0TrNBQwtFU0xRZk
t6X3NkZFN1MC0wNw1jZktMzNxVVFQNmZ6bjVPY1k2R61uNwtaQ1ZRFpIz0hDR19BX0sxeFFR0fhfa3VQ7DRKu0f4Q05DaVuZUFEcjMEh4WHDvaVlyTTN0Q0R1nNbp
Q31tR0NiUetDMa1Tm5BeQtNmVONDNKSTR50MmjZUxg4c1NMRRE9BQ7TSVzLwVpFGRLeH10e01s5kEtXzhGdzRpt3wznzL0tBa1o1C1ha1FESzfoNm5fb302LV
pDSmtKw0notN3IyeDA3Q1hXzvZdpd2FTQ1J7fdFKzU4e15V1fC0n1Mu1FaWtbfmNDEdxaSkeStkRt3DnT3V2xxcmewU1BecTiwOGxZnByUFb1Ehfm0V1
xZxOGNSNV1o5G5TOTNWZ1VPMESTUwZoAt11j1Iend0SU0z0Xpva1qX2R10wdaamFWGEzazBUUjh0bjVNznrwFZURDAtmc8b03fJMzRhyjyLXJzQnkNkwRzvRhz0
IIwlkzc0hMR35QWzQyNpITVBNEtBVU9KOUJVShjU3FwbnI0TmF1ZfNfUzFtMv5NxbFvREzajBiT3ptY2M0vUfzT1FrbnJyZuxxbThiS0zRSTQxR2dKUkpt0Vv2a0NS
LBVNrk9uU0h1T3FqbWN1N1pLdwExcVRPrUZYNNkrk91akJqcurcy31dTm9x3RvB0rscw4yLtnna0dHcnBwUvVzQw9sT015S0nfu2w4a3BEDmpYTrnrlz2d6UUI5Vm5sbm
xROv9mWtNobXvcjBarU4vEvRhamZys005WzxRktHm2NyUmt4M3NjQ1nwlvc2cm3dmZ6wDV2RghxYkhGwMjIYnVvWXBGaJfirG1c193LhRRFzYMuLnYmx6TxNxDws
QndNm1kLOHj00VQfem10SGhNw1ZwMmRryUdRdG0vN1vQ2NxMG1k1M1uWduQkt3Zg1H1R1Z4Zms3Qmhs3EseyanJxm0s0zWdqU5h0uTxU05oQ2poUvo20U0xNmlwQmzma
RuejVvVDBMLw50s2NRRU1maUetcmttd15NkjkB2raaThTNIM0dR2dKxtdmpqaUr15nhL0jdc1FvU12yVml6UGxqTwsxuNidmd3vdyZk9KTGNj0wVkkLw11chNNeH21
cGoyVxhFw1hV0TVAx3ZyN2k4eEdTwnhtbFdbyvNjMEVfUEE1vew3R1FmeGzJtzRwfTzHsak1JUFF0RepNa0V0S0zSt1zrbjhXaw83bm9z11BazFncGx4a1BSM1nDSU
VsVENJU1B3ZxVfNKh1N11xM3d4Y1zud0NBWdd5ewp5VhVCSVjRmB6T1c4NG9NwV1sSkM2N3dud1lCrXfpM1Rzmxz6N1d4cmF1bm5fZE53Sw1t1Vjzx2j2Qz1Cs1daUgdE
TnBaZFrkwc5Se1Ktev55z16Fpjz3ZqsfzvbtwvTm0Q8xkcczZOHFnbt15Zw1saz15C1f5MsYm10xjF1nNBnVsrKj2dFkwezyjzU3c21kvGnjdhNfrupnag3nmv
VxZfnM1jwRhnh2MkhDmErzWt1Pazd1QXv2hLq1pNdwu0U1vcmQzWTRxd1pIampBFYjCwRb0pMyVzalvZLw1MwpwyUpHap3pJawZnuvhzRmnrLwtZunfiute5
M1VzYt1TWTZFLtdfcmVzDndtDeDhNwxQs01UR1vNZ29CUFVmtn1EaUvBbxxRrstF6yWVwd0pLanFTOUxKb0zKt09qafh1cVp5Mc1zC1E0zexuQk10dxheweFF1nBoSDm0dm
RiMmzdZ2p0vEm4dF2cDnfQtbMdnh4enVnzfuZs11wYz1sdHfnwHhn1hNg0vVfydk1MtMf1hRd1vCmxvutz1TeWhmTbmtzoc1v0ejrZ9GrY1ZwWjBLvFox0Fx
Tjrvr1Myr1rgw0s3eUqxanWmEcZmuJr1m1PunBMzuh6MhbjZ0VmTy1Nyx1h3zVkdjt3B0bjy3RwN5d2vfr3kdt6v6j13RgvvY3pUbnuX0zIn1FwverzTEu40DFSRn
o3THj0tW1lusuVjVgrvNNeUwRmtXwF1muNvshjccfj0wm1M2z0Tgpief1kQ1i1Rn3HnjyZmrdmvpQkowd1bjZef4215Q1ft1hDtnQ3Mhr0sTE2R1iZbhqzf1Tr1F2
X211dGzrZGF1TU1GUW9KahfVzlaYwdjV2IydxNaE5HmEEztxbuof14MeHaQ1BwdkgzsfniMnV1el30EFxVGxta0FscuywzZuwnfNmtgxjejQ2Z05Lwj3d3M2g4YT15LX
1DaTB5UhpfbT0ZjaG5XanJfbmRqs2V3NmFab0RFShd5vBzbk83eTkzUUrj0GtiaF6gyj0ymtbaEJDM1NzZTh3R3zntxjor1k13u21nZnh00cztTh0bH02dmZbc1f1
0Th0d28wVhp1M0s3Mkttb0Q4a2hkdzYctZPWyb2JBMu05d2ZpNTFxbWpqah5cJruws03chFeyzuxT21ieFnrvxJbzatN1B1amE1RfvgvU1PUTNsWk0waVdsmv1ky2
1xQw9GcC1Y045Q3jIMjg3dkpaaFcyczRhzxm4Uy1xdwRhox11NjF1M2Ixchdsmgzzc0VpvxpRdwFzx3QzcWtaawfHaHzameeybkvgzfkyd2tubwfvm1kcxrza3u4cmhQ
S25xvuxsq1952G52eVfpT3hucr351jYel1MdRwbGfHwU1Z21RsMhplc1nsdnBrdrJbmtjvtbpa0jEq2vrvn5M00pxTrnZtw1M1dV9veve1jLxbi0uSSDR6V3htM1RBbE
1Mem9DLxydTBzSpHCW9rM3R2svhboxv50WkzcxZQjeqfyzUfmv2c3dz5cVFVaFBkLTZQ29sYmrkRnxFWEVrYjQzsnRseTN3bnhKsvpDR1pvcXdpvxZQZehwem0wQ31m
ekx4nZfj0mNweuMzTgtxn3BEVvdQjx3s1ySgpvZfVtd1N0djuYnY5C1hbf0fJd0FtZk1Kc1daZxZTz2BwgNQ5n8apKes9fb4htdQbterdKIAZTmphy6r_h7k7og4i3
gmgsd3FEP604CS6jTTUTOST6gDmuDWZ0zTpTwetj64P4oF1WlzKf6tX8alrkaQR2npTXh_ah87BkW69myzaKb4D9obNpg7qdk7lgkpQ180olmBtPxIV-
wkiN92F6n2fp0i5B1t1wS_h8wxGIA6Nkm0s-
RoAYL7GtvgBSogOHKhvGaXnhesQY7KZgQTE9OrCc_fliqyRABHtpgyBwb7Wp0hPodZQ0dPaduMKkprs05VidFzJUfxduYc7ZbZE-
g_tixrjK3LinfrNzXyloGohBw5GRPHu3wlTg

```

6.2 Authorization Token

Restful API needs to include an authentication token (JWT). Samsung and partners can use the token to authenticate API calls.

JWT Details

JWS Header

JWS Header	Requirement	Description
alg	Required	Cryptographic algorithm used to sign the payload. e.g., RS256
cty	Required	Payload content type *such as 'AUTH'

JWS Header	Requirement	Description
ver	Required	Token version. Set as 2.
partnerId	Required	Partner ID. Same as partnerCode.
utc	Required	Creation time. To prevent repeated use, the token expires after a certain period of time. Unix timestamp in milliseconds. *Time offset from UTC of +00:00

[JWS Payload]

JWE Payload	Requirement	Description
API	Required	Current API information
API.method	Required	API method
API.path	Required	API path
refId	Optional	A unique content identifier defined by the content provider
updatedAt	Optional	Data update timestamp. Epoch timestamp in milliseconds.
token	Optional	Token to verify the request
simList	Optional	List of Sim Card Information * deprecated
simList[].contacts	Optional	Personal contact information such as phone number * deprecated
simList[].uiccId	Optional	Universal IC Card ID * deprecated

[Authentication Token Example]

JWS Header:

```
{"cty":"AUTH","ver":1,"partnerId":"1234567890","utc":1631775948348,"alg":"RS256"}
```

JWS Payload:

Samsung Wallet Cards – API Specification

```
*Samsung Server API > Update Notification
{
    "API": {
        "method": "POST",
        "path": "/wltx/cards/12584806754/notification"
    },
    "refId": "ref-20230304-0003"
}
```

```
*Partner Server API > Get Card Data
{
    "API": {
        "method": "GET",
        "path": "/cards/12584806754/ref-20230304-0003"
    },
    "refId": "ref-20230304-0003"
}
```

JWS Result:

```
eyJjdHkiOiJBVRRIiwidmVjIjoxLCJwYXJ0bmVySWQiOixMjM0NTY30DkwIiwidXRjIjoxNjMxNzc10TQ4MzQ4LCJhbGciOiJSUzI1NiIsImtpZCI6IlldMVC5QUk1LRV
k1fQ.ewogICAgICAgIpkFQSSIGIHsKICAgICAi1W0aG9kIjogIkdfVCIsC1AgICAgICAgInBhdGgiOiaIL2NhcmtQ1MxNjEzODM1MzIxMjU4NDgwNjc1NCIKICAgIH0sc
iAgICAIcmVmSWQiOiaiQ1MxNjEzODM1MzIxMjU4NDgwNjc1NCIKfQo.AscAwII-aMbJKoly_AuZagxrwUUmKfUhBZnrLk0YkvBy0g2dSLJs-
_xyQ9tooh4cwsfpkeJ0VqkWBRYROKABkhwMRdkjrAjeAQ-87s-
b0p1RCBeLNzMFq66gCmbg9xpD6dmwWlnRAzySZjrcyZklLu9si5qYKrkyU0z34MCwzwdNeOs3z3Gl1xft42M2-
cDUxKQwi0wfrYAnxIEdwboIYu12SDnPsRBWlb7liw4oMM6fg01diRTbK6AYumbf7Zqj1_oygeLv9jFDY0zE0TQykLtTSHGdws7IMyamhA5nhaGP1hqIVzAQooSA14gBCm1
U0zDqw4JQa4-1Vgjr_i5XEA
```

Authorization Token:

```
Bearer
eyJjdHkiOiJBVRRIiwidmVjIjoxLCJwYXJ0bmVySWQiOixMjM0NTY30DkwIiwidXRjIjoxNjMxNzc10TQ4MzQ4LCJhbGciOiJSUzI1NiIsImtpZCI6IlldMVC5QUk1LRV
k1fQ.ewogICAgICAgIpkFQSSIGIHsKICAgICAi1W0aG9kIjogIkdfVCIsC1AgICAgICAgInBhdGgiOiaIL2NhcmtQ1MxNjEzODM1MzIxMjU4NDgwNjc1NCIKICAgIH0sc
iAgICAIcmVmSWQiOiaiQ1MxNjEzODM1MzIxMjU4NDgwNjc1NCIKfQo.AscAwII-aMbJKoly_AuZagxrwUUmKfUhBZnrLk0YkvBy0g2dSLJs-
_xyQ9tooh4cwsfpkeJ0VqkWBRYROKABkhwMRdkjrAjeAQ-87s-
b0p1RCBeLNzMFq66gCmbg9xpD6dmwWlnRAzySZjrcyZklLu9si5qYKrkyU0z34MCwzwdNeOs3z3Gl1xft42M2-
cDUxKQwi0wfrYAnxIEdwboIYu12SDnPsRBWlb7liw4oMM6fg01diRTbK6AYumbf7Zqj1_oygeLv9jFDY0zE0TQykLtTSHGdws7IMyamhA5nhaGP1hqIVzAQooSA14gBCm1
U0zDqw4JQa4-1Vgjr_i5XEA
```

7. Image Resources

7.1 Button Image resources

Add to Samsung Wallet Button

[Image Sample]

Type	Locale	Version	Image
btnSW	N/A	normal	
		normal, dark	
		vertical	
		vertical, dark	
btnATSW	en-kr, en-us, en-um, en-vi, en-gb, en-vg, en-io, en-fr, en-gf, en-tf, en-pf, en-de, en-it, en-es	normal	
		normal, dark	
		vertical	
		vertical, dark	
	fr-fr, fr-gf, fr-tf, fr-pf	normal	
		normal, dark	
		vertical	
		vertical, dark	
	de-de	normal	

		normal, dark	Zu SAMSUNG Wallet hinzufügen >
		vertical	Zu SAMSUNG Wallet hinzufügen >
		vertical, dark	Zu SAMSUNG Wallet hinzufügen >
it-it	normal	Aggiungi a SAMSUNG Wallet >	
	normal, dark	Aggiungi a SAMSUNG Wallet >	
	vertical	Aggiungi a SAMSUNG Wallet >	
	vertical, dark	Aggiungi a SAMSUNG Wallet >	
ko-kr	normal	SAMSUNG Wallet에 추가 >	
	normal, dark	SAMSUNG Wallet에 추가 >	
	vertical	SAMSUNG Wallet에 추가 >	
	vertical, dark	SAMSUNG Wallet에 추가 >	
es-es	normal	Añadir a SAMSUNG Wallet >	
	normal, dark	Añadir a SAMSUNG Wallet >	
	vertical	Añadir a SAMSUNG Wallet >	
	vertical, dark	Añadir a SAMSUNG Wallet >	

[Base URL]

US: <https://us-cdn-gpp.mcsvc.samsung.com/lib>

KR: <https://kr-cdn-gpp.mcsvc.samsung.com/lib>

[File Name]

Type	Version	File Name
btnSW	normal	wallet-card.png

Type	Version	File Name
	normal, dark	wallet-card-dark.png
	vertical	wallet-card-ver.png
	vertical, dark	wallet-card-ver-dark.png
btnATSW	normal	add-to-samsung-wallet_{locale}.png
	normal, dark	add-to-samsung-wallet_dark_{locale}.png
	vertical	add-to-samsung-wallet_ver_{locale}.png
	vertical, dark	add-to-samsung-wallet_dark_ver_{locale}.png

[Default value of each resource property]

- Type : btnSW
- Locale : en-us
- Version : normal

8. References

8.1 Barcodes

[Presentation type & formats]

The barcode display type defines how the serial number, barcode, and/or QR code is shown.

[Presentation Types (serialType)]

Code name	Description
SERIALNUMBER	Serial Number.
BARCODE	Barcode.
QRCODE	QR code.

[Presentation Formats (ptFormat)]

Code name	Description
BARCODE	Barcode only
BARCODESERIAL	Barcode with serial number
SERIAL	Serial number only
DUALSERIAL	Dual serial number
DUALBARCODE	Dual barcode
DUALBARCODESERIAL	Dual barcode for each serial numbers
BARCODEPIN	Barcode with PIN
QRCODE	QR code only
QRCODESERIAL	QR code with serial number.

[Barcode Formats (ptSubFormat)]

Below are commonly used barcode formats, supported by the ZXing barcode scanning library.

See also: <https://zxing.github.io/zxing/apidocs/com/google/zxing/BarcodeFormat.html>

Code name	Description
APIAZTEC	Aztec 2D barcode format.

Code name	Description
CODABAR	CODABAR 1D format.
CODE_39	Code 39 1D format.
CODE_93	Code 93 1D format.
CODE_128	Code 128 1D format.
DATA_MATRIX	Data Matrix 2D barcode format.
EAN_8	EAN-8 1D format.
EAN_13	EAN-13 1D format.
ITF	ITF (Interleaved Two of Five) 1D format.
MAXICODE	MaxiCode 2D barcode format.
PDF_417	PDF417 format.
QR_CODE	QR Code 2D barcode format.
RSS_14	RSS 14.
RSS_EXPANDED	RSS EXPANDED.
UPC_A	UPC-A 1D format.
UPC_E	UPC-E 1D format.
UPC_EAN_EXTENSION	UPC/EAN extension format. Not a stand-alone format.

8.2 Locations

Locations refer to place information that denotes where a card can be used.

Using this information, Samsung Wallet can show the map, a place name, and address. Additionally, location information can be used to provide Location-Based Services (LBS).

Location information can be represented by a JSON array and up to 10 locations can be specified.

* Notice: Map services are only available in certain countries.

Details

[JSON Format]

Key	Type	Requirement	Description
[] .lat	Double	Optional	Latitude.
[] .lng	Double	Optional	Longitude.
[] .address	String	Required	String containing the full address.
[] .name	String	Required	Branch name.

[Example]

```
* Location information for the entrance to Oracle Park.  
[  
  {  
    "lat": 37.779337,  
    "lng": -122.388755,  
    "address": "24 Willie Mays Plaza, San Francisco, CA 94107",  
    "name": "Willie Mays Plaza"  
  },  
  {  
    "lat": 37.77814,  
    "lng": -122.390836,  
    "address": "King St, San Francisco, CA 94107",  
    "name": "King St"  
  }  
]
```

8.3 Additional information

Additional information to be delivered to customers can be defined in the following format.

Be careful of the content string length. If an attribute does not allow long content, it will not be displayed on the device.

Details

[JSON Format]

Key	Type	Requirement	Description
count	Integer	Required	Size of information.
info[]	Array of Objects	Required	Container of information
info[].title	String	Required	Title.
info[].content[]	Array of Strings	Required	Content text

[Example]

```
* Extra information for a boarding pass.  
{  
    "count": 3,  
    "info": [  
        {  
            "title": "baggageAllowance",  
            "content": [  
                "15 KG"  
            ]  
        },  
        {  
            "title": "boardingPriority",  
            "content": [  
                "YES"  
            ]  
        },  
        {  
            "title": "seatClass",  
            "content": [  
                "Economy Plus"  
            ]  
        }  
    ]  
}
```

* An example case of long content being allowed: Movie ticket policy.

```
{  
    "count": 2,  
    "info": [  
        {  
            "title": "REFUNDS AND EXCHANGES",  
            "content": [  
                "Refunds and exchanges of movie ticket(s) are available in certain limited circumstances.",  
                "Movie tickets purchased through the Services include a non-refundable Convenience Fee.  
Before purchasing your movie ticket(s) we urge you to confirm the title, time, location and quantity of tickets for  
the movie you wish to see."  
            ]  
        },  
        {  
            "title": "CHANGES TO TICKET POLICY",  
            "content": [  
                "From time to time, we may revise this Ticket Policy. You can determine when this Ticket  
Policy was last revised by referring to the top of this page. Any changes to this Ticket Policy will become effective  
upon posting of the revised Ticket Policy on the Internet, accessible through the Services."  
            ]  
        }  
    ]  
}
```

* Using Dynamic QR Code

8.4 Links

Linkable data which provides additional information in the following format.

Details

[JSON Format]

Key	Type	Requirement	Description
count	Integer	Required	Size of links
info[]	Array of Objects	Required	Container of links
info[].link	String	Required	Link URL
info[].type	String	Required	View type that will run the link * Allowed values: web, app, browser
info[].text	String	Optional	Text of the link

[Example]

```
{  
    "count": 1,  
    "info": [  
        {"link": "https://samsung.external.info.link",  
         "type": "web",  
         "text": "See more information"  
     ]  
}
```


8.5 Classification

Classification defines different kinds of people who can use the cards.

Details

[JSON Format]

Key	Type	Requirement	Description
person[]	Array of Object	Required	Container of person list who can use the card
person[].category	String	Required	Category name
person[].count	String	Required	Number of person

[Example]

```
* 3 persons with a ticket.  
{  
    "person": [  
        {  
            "category": "Adult",  
            "count": 2  
        },  
        {  
            "category": "Child",  
            "count": 1  
        }  
    ]  
}
```

```
* 1 person with a ticket.  
{  
    "person": [  
        {  
            "category": "Adult",  
            "count": 1  
        }  
    ]  
}
```

8.6 Check service available devices

Check Samsung Wallet service availability for devices and countries.

[Request]

Type	Value			Description
Method	GET			
URL	https://api-us3.mpay.samsung.com/wallet/cmn/v2.0/device/available			
Header	partnerCode	String	Required	Partner Code - Partner Code or ID from Samsung Wallet Partner System
Query Parameters	modelName	String	Required	Device model name Ex) SM-G925K, SM-R730T
	serviceType	String	Required	Service type - WALLET : Samsung Wallet
Example				

[Response]

Type	Value	Description
body	resultCode	String
	resultMessage	String
	available	boolean
	supportKR	boolean
Example (Success)	HTTP/1.1 200 OK Content-Type: application/json; charset=UTF-8 Content-Length: xx { "resultCode" : "0", "resultMessage" : "SUCCESS", "available": "true", "supportKR": "true" }	Result Code Result Message Service availability KR service support device
Example (Error)	HTTP/1.1 400 Bad Request Content-Type: application/json; charset=UTF-8 Content-Length: xx { "resultCode" : "CMN5N9001",	

	"resultMessage" : " Internal Server Error" }
--	---

[Response]

Http Code	Status Code	DESCRIPTION
200	0	Success
400	CMN1N1001	Missing requisite parameter from API request – {0}
400	CMN1N1002	Invalid parameter from API request – {0}
400	CMN1N1003	Invalid parameter Length from API request – {0}
400	CMN2N2007	Invalid Country Code
500	CMN4N8003	DB Connection Failed

8.7 Security factors

Service Flow

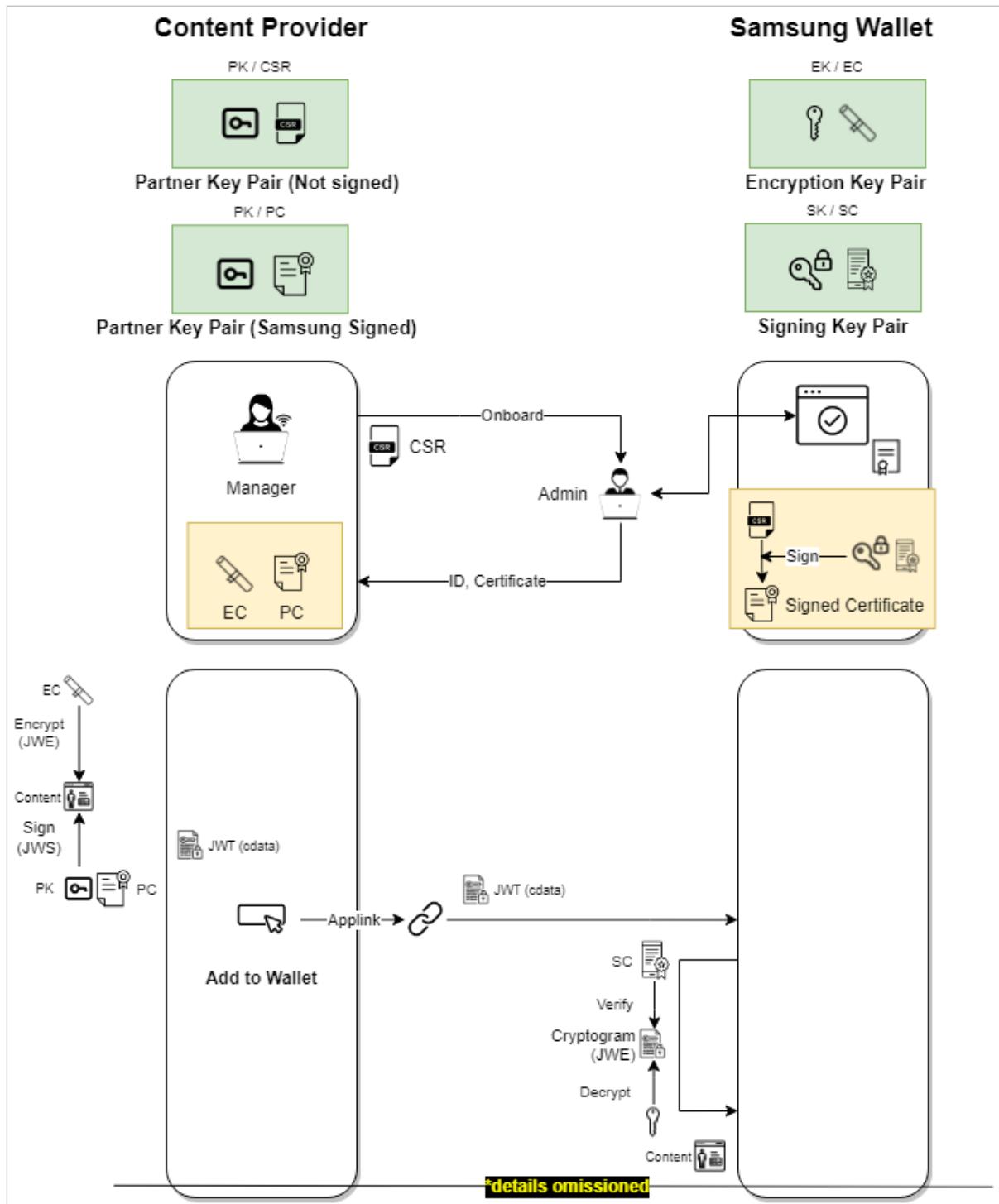


Figure 8-1 Data Security Flow

How to generate key factors

OpenSSL is an open-source command-line tool that allows you to perform various SSL-related tasks. This article will explain how to create key factors for security with OpenSSL.

Private Key

A private key enables encryption and is the most important component of certificates.

```
openssl genpkey -out domain.key -algorithm RSA -pkeyopt rsa_keygen_bits:2048
```

If you want to make it more secure, adding '*-des3*' on the command will encrypt with password.

Certificate Signing Request (CSR)

Certificate Signing Request (CSR) is a necessary factor to get a signed certificate. The CSR includes the public key and some additional information, such as organization and country.

Let's create a CSR (domain.csr) from the existing private key.

```
openssl req -out domain.csr -key domain.key -new -sha256
```

You'll enter CSR information to complete the process. The output will look like this:

```
You are about to be asked to enter information that will be incorporated  
into your certificate request.  
What you are about to enter is what is called a Distinguished Name or a DN.  
There are quite a few fields but you can leave some blank  
For some fields there will be a default value,  
If you enter ''; the field will be left blank.  
----  
Country Name (2 letter code) [XX]:KR  
State or Province Name (full name) []:Seoul  
Locality Name (eg, city) [Default City]:Sample City  
Organization Name (eg, company) [Default Company Ltd]:Sample Company  
Organizational Unit Name (eg, section) []:Sample section  
Common Name (eg, your name or your server's hostname) []:domain  
Email Address []:email@email.com  
  
Please enter the following 'extra' attributes  
to be sent with your certificate request  
A challenge password []:  
An optional company name []:
```

* 'password' is optional attribute.

* An important field is 'Common Name', which should be the exact Fully Qualified Domain Name (FQDN) of our domain.

Samsung Wallet Cards – API Specification

For more information, refer to '<https://www.openssl.org>'.

8.8 Contact points

[Dev Team]

Name	Address
Wallet Technical Support	https://developer.samsung.com/dashboard/support

< End of Document >