

The information in this document is confidential and meant to be used only by the intended or authorized owner, it contains confidential, proprietary or legally privileged information. No confidentiality or privilege is waived or lost by any miss-transmission. If you receive this document in error, please immediately delete it and all copies of it, destroy any hard copies of it and notify the  
TIMWETECH. You must not, directly, use, disclose, distribute, print, or copy any part of this message if you are not the intended or authorized owner.



Presented by:

TIMwe Asia Pacific Sdn Bhd

Suite 18-02, Level 18, GTower 199 Jalan Tun Razak, 50400, Kuala Lumpur, Malaysia

Integration Specification

Integration with Card Management System (TNM)

Version 0.3

February 4, 2020

Contents

[VERSION HISTORY 4](#_Toc33427181)

[BUSINESS SPECIFICATIONS & POLICIES 5](#_Toc33427182)

[INTEGRATION SPECIFICATION 6](#_Toc33427183)

[Application 6](#_Toc33427184)

[Sub-Systems 6](#_Toc33427185)

[Interfaces 6](#_Toc33427186)

[Authentication 6](#_Toc33427187)

[Offline interfaces 6](#_Toc33427188)

[Online interfaces 6](#_Toc33427189)

[STANDARDS 7](#_Toc33427190)

[Offline data file feed 7](#_Toc33427191)

[Control file 8](#_Toc33427192)

[Possible Rejection Cases 9](#_Toc33427193)

[File rejection reasons 9](#_Toc33427194)

[Record rejection reasons 9](#_Toc33427195)

[Rejection types & errors 10](#_Toc33427196)

[Online REST API interface 11](#_Toc33427197)

[Process flowchart 12](#_Toc33427198)

[PRODUCT DUMP (INTTNM001) 13](#_Toc33427199)

[Context goal 13](#_Toc33427200)

[Pre-conditions 13](#_Toc33427201)

[Post-conditions 13](#_Toc33427202)

[File naming 13](#_Toc33427203)

[Paths 13](#_Toc33427204)

[Data feed fields 14](#_Toc33427205)

[Processing result 14](#_Toc33427206)

[STARTERPACK STOCK DUMP (INTTNM002) 15](#_Toc33427207)

[Context goal 15](#_Toc33427208)

[Pre-conditions 15](#_Toc33427209)

[Post-conditions 15](#_Toc33427210)

[File naming 15](#_Toc33427211)

[Paths 15](#_Toc33427212)

[Data feed fields 16](#_Toc33427213)

[Processing result 16](#_Toc33427214)

[STARTERPACK ALLOCATION DUMP (INTTNM003) 17](#_Toc33427215)

[Context goal 17](#_Toc33427216)

[Pre-conditions 17](#_Toc33427217)

[Post-conditions 17](#_Toc33427218)

[File naming 17](#_Toc33427219)

[Paths 17](#_Toc33427220)

[Data feed fields 18](#_Toc33427221)

[Processing result 18](#_Toc33427222)

[VOUCHER STOCK DUMP (INTTNM004) 19](#_Toc33427223)

[Context goal 19](#_Toc33427224)

[Pre-conditions 19](#_Toc33427225)

[Post-conditions 19](#_Toc33427226)

[File naming 19](#_Toc33427227)

[Paths 19](#_Toc33427228)

[Data feed fields 20](#_Toc33427229)

[Processing result 20](#_Toc33427230)

[VOUCHER ALLOCATION DUMP (INTTNM005) 21](#_Toc33427231)

[Context goal 21](#_Toc33427232)

[Pre-conditions 21](#_Toc33427233)

[Post-conditions 21](#_Toc33427234)

[File naming 21](#_Toc33427235)

[Paths 21](#_Toc33427236)

[Data feed fields 22](#_Toc33427237)

[Processing result 22](#_Toc33427238)

[EXTEND EXPIRY DATE (INTTNM006) 23](#_Toc33427239)

[Context goal 23](#_Toc33427240)

[Pre-conditions 23](#_Toc33427241)

[Post-conditions 23](#_Toc33427242)

[File naming 23](#_Toc33427243)

[Paths 23](#_Toc33427244)

[Data feed fields 24](#_Toc33427245)

[Processing Result 24](#_Toc33427246)

[QUERY STOCK (INTTNM007) 25](#_Toc33427247)

[Actors, surrounding systems involved 25](#_Toc33427248)

[Context goal 25](#_Toc33427249)

[Pre-conditions 25](#_Toc33427250)

[Post-conditions 25](#_Toc33427251)

[Endpoint URL 25](#_Toc33427252)

[request & response 26](#_Toc33427253)

[Request 26](#_Toc33427254)

[RESPONSE 26](#_Toc33427255)

[Rejection types & errors 26](#_Toc33427256)

[RESERVE STOCK (INTTNM008) 28](#_Toc33427257)

[Actors, surrounding systems involved 28](#_Toc33427258)

[Context goal 28](#_Toc33427259)

[Pre-conditions 28](#_Toc33427260)

[Post-conditions 28](#_Toc33427261)

[Endpoint URL 28](#_Toc33427262)

[request & response 29](#_Toc33427263)

[Request 29](#_Toc33427264)

[RESPONSE 29](#_Toc33427265)

[Rejection types & errors 29](#_Toc33427266)

[APPROVAL SIGN-OFF 31](#_Toc33427267)

This document outlines the interface specification for integration

# VERSION HISTORY

|  |  |  |  |
| --- | --- | --- | --- |
| VERSION | DATE | CHANGE | BY |
| 0.1 | 16/01/2020 | Creating of document | Romet Roosalu |
| 0.3 | 03/02/2020 | Updated with Online Interfaces | Parthipan Rajagopal |
|  |  |  |  |
| 1.0 | 04/02/2020 | Final sign-off version | Romet Roosalu |

# BUSINESS SPECIFICATIONS & POLICIES

Indosat is replacing its FOSS system with TNM system. MOBII was integrated with FOSS system for allocating the stock to partners, sync product details and extending the product expiry date. MOBII must update its interface to support TNM – Majority of the offline interfaces are used same structure of FOSS interface. Few of them must be updated based on TNM data. Additionally, TNM must provide two online APIs for stock query and stock reservation to support primary sales operation.

# INTEGRATION SPECIFICATION

## Application

|  |  |  |  |
| --- | --- | --- | --- |
| ENVIRONMENT | LOCATION | IP | LOGICAL SERVER NAME |
| Production | Client Network DC1 | 10.34.19.136 | Batch Server |

## Sub-Systems

|  |  |  |  |
| --- | --- | --- | --- |
| ENVIRONMENT | LOCATION | IP | LOGICAL SERVER NAME |
| Production | Client Network DC1 | N/A | N/A |

## Interfaces

|  |  |
| --- | --- |
| INTERFACE ID | NAME |
| INTTNM001 | Product Dump |
| INTTNM002 | Starter Pack Stock Dump |
| INTNNM003 | Starter Pack Allocation Dump |
| INTTNM004 | Voucher Stock Dump |
| INTTNM005 | Voucher Allocation Dump |
| INTTNM006 | Extend Expiry Date |
| INTTNM007 | Get Product Stock Availability API |
| INTTNM008 | Reserve stock for a Dealer API |

## Authentication

### Offline interfaces

MOBII SFTP server username and password will be shared at deployment time with TNM team.

SFTP access (port 22) between TNM to MOBII has to be enabled by Indosat hardware team.

### Online interfaces

Access to TNM server from MOBII batch server has to be enabled by Indosat hardware team. Below http headers will be passed as part of each request call.

|  |  |  |
| --- | --- | --- |
| No | Header Name | Header Value |
| 1 | Accept | Application/json |
| 2 | Content-Type | Application/json |
| 3 | x-auth-type | None |

# STANDARDS

## Offline data file feed

MOBII follows ASCII file interface for offline file integration and REST API call for online interfaces with TNM system.

MOBII always follows few standards for its offline file integration with surrounding system the same is described as follows,

* Offline file has the format of ASCII text file and one record per line.
* First line of record is reserved for header.
* Even if there is no allocation/transaction in TNM still file must be shared to MOBII just with header.
* TNM must push the files to MOBII SFTP server path. File format and path are explained in corresponding interface section.
* Only agreed file names will be considered from SFTP path.
* Other files will not be considered for process and MOBII support team can delete from SFTP path by not informing to anyone.
* MOBII schedulers are configured to accept the files from 2AM daily, so file must be available in MOBII SFTP path before that.
* After that MOBII process starts validating on record level. If there are record level rejection while processing, System shall process all success records & generate rejected file for all rejected records with reason of rejection & copy this file to rejected folder path. System shall also send mail to operations team for intimation of rejected files.
* Both rejected and actual files are moved to backup and available to download from MOBII web UI for period after that that it will be deleted from system.
* Additionally, if MOBII identify any issue in one record and the complete transaction/allocation will be rejected. Problematic row will contain actual rejection reason and other row contains generic rejection reason like “Rejected due to dependent row”.
* All files (control, actual, rejected actual and rejected control) will be zipped (.gz format) & pushed to SFTP/FTP server. MD5 key will be generated before zipped.
* Fields delimiter:
  + Actual file: pipe – '|', field values that contain the pipe character
* Control file: the field separator will be TAB.
* Below status will be maintained for each file:
  + File Received
  + File Processed
  + File Completed
  + File Rejected
  + File Partially Rejected

### Control file

MOBII always accept the offline files along with control file, this file needs to be generated for every sync file.

Control file contains below details with <TAB> delimiter.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Field | Mandatory | Data Types | Description |
| 1 | Sync file gen Date | Yes | Date Time | Date of file generated. In the format YYYY-MM-DD HH:MM:SS in 24 hour format |
| 2 | Number of Row in Sync File | Yes | Number | Number of rows in the .csv file including header. |
| 3 | Size | Yes | Number | Sync file size in bytes |
| 4 | Hash Key | Yes | String | MD5 hash key of sync .csv file. /usr/bin/md5sum UNIX command will be used to generate the key. |
| 5 | Sync File name | Yes | String | File name of dump/sync file. |

* Both control file and actual file names must be match only file extension will be different.
* TNM must copy the actual file first to the SFTP path and then control file, when MOBII transfer the file from SFTP to local it searches the control file first and then actual file.
* MOBII will generate the MD5 hash key for actual file and compare this key with control file key and additionally it will validate with file size and number of rows between actual and control file specification. If any mismatch identified, then those file statuses will be changed to rejected and notified to MOBII support for further action.

### Possible Rejection Cases

### File rejection reasons

* If generated MD5 hash key does not match or other values from control file does not match.
* During file processing, if file is not as per expected format.
* Mandatory columns missing in file (Heading level validation first row).

### Record rejection reasons

* During file processing, if any record is rejected, all records will be processed except the erroneous one.
* Such rejected records details would be pushed to FTP path & alert will be send to operation team on email. User needs to download and correct such records & upload it back to SND again
* Common cases for record rejection:
  + Reference data not existing in system
  + Product ID does not exist in system
  + Dealer code does not exist in system
  + Serial Number Already existing in system.
  + Serial Number does not exist in system.
  + Serial Number does not exist with that dealer.
* All dependent transactions related to error record will also get rejected.

### Rejection types & errors

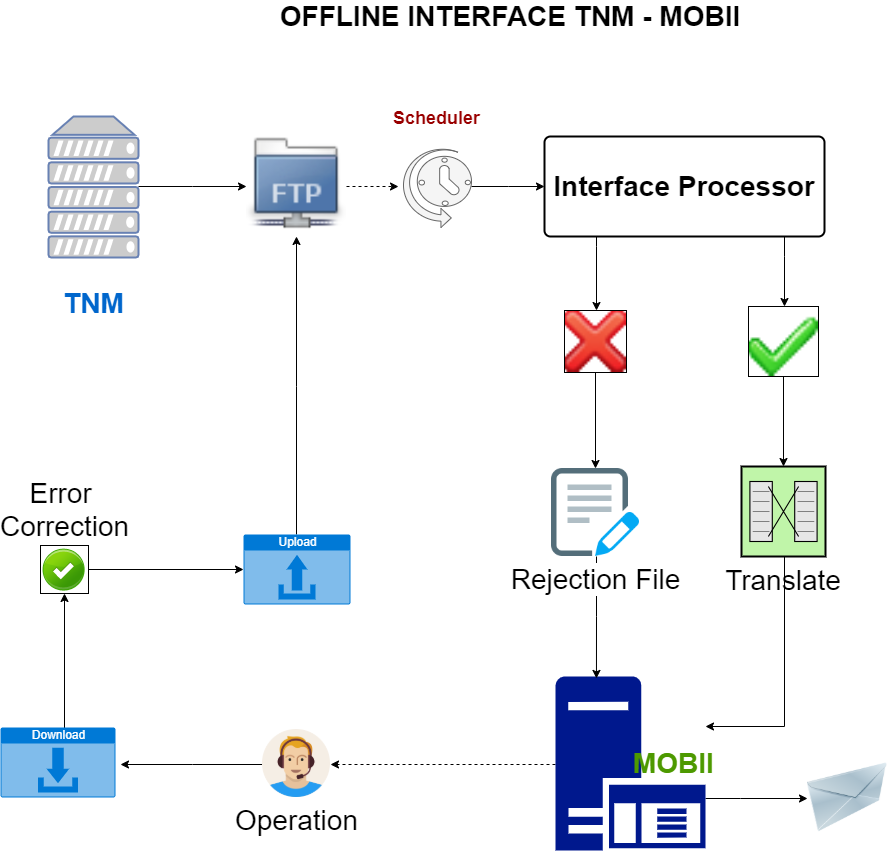
|  |  |
| --- | --- |
| ERROR ID | ERROR MESSAGE |
| 101 | Unknown error on validation |
| 1002 | No of columns not exist correctly--> <<Record in File>> |
| 1004 | Duplicate record exists in the shared file |
| 1006 | Mandatory Field <<Field Name>> is missing |
| 1009 | <<Field>> contains invalid data <<Field Value>> |
| 3001 | Invalid Organization : <<VALUE>> |
| 3002 | Organization is not in active status : <<VALUE>> |
| 3003 | Invalid Organization or not in Active status : <<VALUE>> |
| 30401 | Invalid Product Code : <<VALUE>> |
| 30402 | Product is not in active status : <<VALUE>> |
| 30403 | Invalid Product or not in Active status : <<VALUE>> |
| 30404 | Invalid Product Category : <<VALUE>> |
| 30405 | Product category not in active status : <<VALUE>> |
| 30406 | Invalid Product category or not in Active status : <<VALUE>> |
| 30407 | Invalid Sub Category for Category : <<VALUE>> |
| 30408 | Invalid Brand : <<VALUE>> |
| 30409 | Product ID not available for : <<VALUE>> |
| 30410 | Found duplicates for given Product ID : <<VALUE>> |
| 30411 | Material Code already exists : <<VALUE>> |
| 30501 | Serial Numbers not exist for the product : <<VALUE>> |
| 30502 | Serial Number Length should not be greater than : <<VALUE>> |

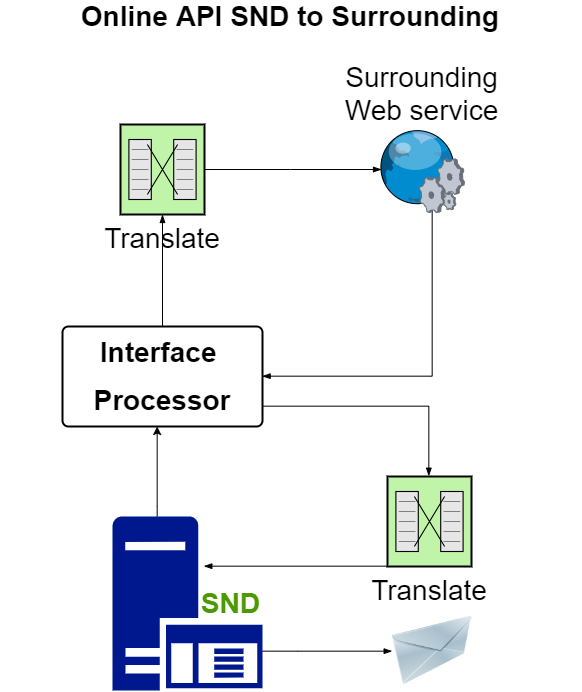
## Online REST API interface

Below standards will be applicable for REST API call between MOBII and TNM

* TNM will provide the end point URL to MOBII team for each REST APIs. The same will be configured in MOBII interface server for corresponding integration.
* TNM must share their server IP for SIT, UAT, PRE-PROD and PROD environment and the access must be enabled from MOBII server.
* Each API must be authenticated, and the required username and password must be shared with MOBII team for each environment.

## Process flowchart





# PRODUCT DUMP (INTTNM001)

|  |  |
| --- | --- |
| INTERFACE TYPE: | Off-line file feed |
| INTERFACE METHOD: | Batch file |
| INITIATOR SYSTEM: | TNM |
| RESPONDER SYSTEM: | MOBII |

## Context goal

1. TNM system will generate a file for latest updated or created products in TNM. In MOBII product and its parameters will be compared with file data, if any difference found that will be updated, if product code not exist in MOBII then it will be created newly.
2. Possible parameters will be updated, includes product name, price and end date. Other than these parameters modification MOBII will skip the record.
3. File will be pushed to MOBII path daily before 2AM with latest updated records. MOBII scheduler will pick pushed files for process.
4. Once file is copied to MOBII interface server then file will be deleted from SFTP path.

## Pre-conditions

1. PRODUCT CATEGORY and BRAND master data is created in MOBII and active.

## Post-conditions

1. If Product Code is new, then new Product is created in MOBII, in Active status
2. If Product Code is available, given file data will be updated for corresponding product in MOBII.

## File naming

PRODUCTS\_<YYYYMMDDHH24MISS>.csv.gz

* <YYYYMMDDHH24MISS> = date and time of file generation.
* Sample Actual File Name is PRODUCTS\_20200203013512.csv.gz
* Sample Control File Name is PRODUCTS\_20200203013512.ctl.gz

## Paths

* SFTP PATH
  + Master: /home/fossuser/interfaces/FOSS/master\_files/product/
* MOBII Application
  + Master: /snoc/interfaces/S-NOC/interfaces/master\_files/product/
  + Backup: /snoc/interfaces/S-NOC/interfaces/backup/master\_files/product/
  + Rejected: /snoc/interfaces/S-NOC/interfaces/rejected\_files/product/
  + Rejected Backup: /snoc/interfaces/S-NOC/interfaces/backup/rejected\_files/product/

## Data feed fields

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Field | Mandatory | Data Types | Description |
| 1 | PRODUCT\_CODE | Yes | Number | Product ID in TNM |
| 2 | SAP\_MATERIAL\_CODE | Yes | Number | Material code associated with Product |
| 3 | PRODUCT\_NAME | Yes | String | Name/Description of the product code |
| 4 | PRODUCT\_CATEGORY | Yes | String | STARTERPACK/VOUCHER/BLANKSIM |
| 5 | PRODUCT\_SUBCATEGORY | Yes | String | IM3(for SP)/IND(VOUCHER) |
| 6 | PRODUCT\_START\_DATE | Yes | Date – YYYYMMDD | Start date of product |
| 7 | PRODUCT\_END\_DATE | Yes | Date – YYYYMMDD | Expiry date of product |
| 8 | SALE\_PRICE | Yes | Double | Selling price material code received from SAP as part of Material code creation. |
| 9 | MRP\_PRICE | Yes | Double | Same as sale price |
| 10 | IS\_SERIALIZED | Yes | Number | It’s always with value ‘1’ |
| 11 | SERIAL\_TYPE | Yes | Number | ‘1’ for VOUCHER/’2’ for STARTERPACK |
| 12 | SERIAL\_LENGTH | Yes | Number | ‘13’ for VOCHER/’20’ for STARTERPACK. At this moment voucher length is 13. |
| 13 | MIN\_ORDER\_QTY | No | String | Minimum Order Quantity value is empty.  If TNM provides any value here that will be used for validation. If it is empty by default it will be considered to 1. |
| 14 | MIN\_PACKAGE\_QTY | No | String | Minimum Package Quantity value is empty.  If TNM provides any value here that will be used for validation. If it is empty by default it will be considered to 1. |

**Example:**

|  |
| --- |
| PRODUCT\_CODE|SAP\_MATERIAL\_CODE|PRODUCT\_NAME|PRODUCT\_CATEGORY|PRODUCT\_SUBCATEGORY|PRODUCT\_START\_DATE|PRODUCT\_END\_DATE|SALE\_PRICE|MRP\_PRICE|IS\_SERIALIZED|SERIAL\_TYPE|SERIAL\_LENGTH|MIN\_ORDER\_QTY|MIN\_PACKAGE\_QTY  974|3520000114|VCR DATA FI 2GB KALISUMAPA|Voucher|IND|20191021|20200131|0|0|1|1|14|0|0 |

### Processing result

File status can be tracked from MOBII web. If any records are rejected, then rejected files can be downloaded from MOBII web which contains all columns of original file and additionally ERROR\_CODE|ERROR\_MESSAGE.

**Example:**

|  |
| --- |
| PRODUCT\_CODE|SAP\_MATERIAL\_CODE|PRODUCT\_NAME|PRODUCT\_CATEGORY|PRODUCT\_SUBCATEGORY|PRODUCT\_START\_DATE|PRODUCT\_END\_DATE|SALE\_PRICE|MRP\_PRICE|IS\_SERIALIZED|SERIAL\_TYPE|SERIAL\_LENGTH|MIN\_ORDER\_QTY|MIN\_PACKAGE\_QTY|ERROR\_CODE|ERROR\_MESSAGE  974||VCR DATA FI 2GB KALISUMAPA|Voucher|IND|20191021|20200131|0|0|1|1|14|0|0|1001|Mandatory Field SAP\_MATERIAL\_CODE is missing |

# STARTERPACK STOCK DUMP (INTTNM002)

|  |  |
| --- | --- |
| INTERFACE TYPE: | Off-line |
| INTERFACE METHOD: | Batch file |
| INITIATOR SYSTEM: | TNM |
| RESPONDER SYSTEM: | MOBII |

## Context goal

1. File contains starter pack transfer transactions along with its serial numbers, which is generated for internal dealers (Example, gallery channel). Transfer between Indosat warehouse to Dealer and Dealer to Dealer both will be shared in file.
2. In stock dump file SO\_ID is considered as transaction reference number, and transactions are grouped based on this.
3. All records from one SO\_ID should contain same source and destination dealer codes.
4. File will be pushed to MOBII path daily before 2AM with latest updated records. MOBII scheduler will pick pushed files for process.
5. Once file is copied to MOBII interface server then file will be deleted from SFTP path.

## Pre-conditions

1. Dealer code, Product master data is created in MOBII and active.

## Post-conditions

1. Given transaction will be registered in MOBII and assigned to dealer to accept from his inbox.
2. Once dealer is approved from his task stock will be added into his inventory.
3. If surrounding system shares the already shared serial numbers once again then MOBII will validate its SO\_ID with previously received SO\_ID, if it is greater than old SO\_ID then stock will be removed from old dealer and allocated to new dealer.

## File naming

SP\_STOCK\_DUMP\_<YYYYMMDDHH24MISS>.csv.gz

* <YYYYMMDDHH24MISS> = date and time of file generation.
* Sample Actual File Name is SP\_STOCK\_DUMP\_20200203013512.csv.gz
* Sample Control File Name is SP\_STOCK\_DUMP\_20200203013512.ctl.gz

## Paths

* SFTP PATH
  + Master: /home/fossuser/interfaces/FOSS/master\_files/sp\_dump/
* MOBII Application
  + Master: /snoc/interfaces/S-NOC/interfaces/master\_files/ sp\_dump/
  + Backup: /snoc/interfaces/S-NOC/interfaces/backup/master\_files/ sp\_dump/
  + Rejected: /snoc/interfaces/S-NOC/interfaces/rejected\_files/ sp\_dump/
  + Rejected Backup: /snoc/interfaces/S-NOC/interfaces/backup/rejected\_files/ sp\_dump/

## Data feed fields

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Field | Mandatory | Data Types | Description |
| 1 | ICCID | Yes | Number | Serial Number |
| 2 | MSISDN | Yes | Number | MSISDN assigned to serial number.  In case of Blank SIM/DSA SIM cards the field will be empty. |
| 3 | IMSI | Yes | Number | IMSI assigned to Serial number. |
| 4 | DO\_ID | Yes | String | DO\_ID in the format – BRAND/YYYYMMDD/Stock Transfer id. |
| 5 | DESTINATION\_DEALER\_ID | Yes | String | Gallery Id or Warehouse id |
| 6 | BRANCH\_CODE | No | Number | Empty |
| 7 | BRAND | Yes | String | IM3 |
| 8 | PRODUCT\_EXPIRED\_DATE | Yes | Date – YYYYMMDD | Expiry date of Product code. |
| 9 | STOCK\_TRANSFER\_DATE | Yes | Date – YYYYMMDD | Stock transfer date in TNM |
| 10 | PROGRAM\_CODE | Yes | Number | Product code ID |
| 11 | PROGRAM\_NAME | Yes | String | Product code description |
| 12 | SOURCE\_DEALER\_ID | Yes | String | Warehouse id or Gallery Id |

**Example:**

|  |
| --- |
| ICCID|MSISDN|IMSI|DO\_ID|DESTINATION\_DEALER\_ID|BRANCH\_CODE|BRAND|PRODUCT\_EXPIRED\_DATE|STOCK\_TRANSFER\_DATE|PROGRAM\_CODE|PROGRAM\_NAME|SOURCE\_DEALER\_ID 89620130003265723946|6285883178220|510018335274160|TEST/20191113/371978|TEST||IM3|20201130|20191113|920|SP IM3 90D LTE (N)|SAT10  89620130003266723945|6285883352548|510018335374160|TEST/20191113/371978|TEST||IM3|20201130|20191113|920|SP IM3 90D LTE (N)|SAT10 |

### Processing result

File status can be tracked from MOBII web. If any records are rejected, then rejected files can be downloaded from MOBII web which contains all columns of original file and additionally ERROR\_CODE|ERROR\_MESSAGE.

**Example:**

|  |
| --- |
| ICCID|MSISDN|IMSI|DO\_ID|DESTINATION\_DEALER\_ID|BRANCH\_CODE|BRAND|PRODUCT\_EXPIRED\_DATE|STOCK\_TRANSFER\_DATE|PROGRAM\_CODE|PROGRAM\_NAME|SOURCE DEALER ID|ERROR\_CODE|ERROR\_MESSAGE  80003265723946|6285883178220|510018335274160|TEST/20191113/371978|TEST||IM3|20201130|20191113|920|SP IM3 90D LTE (N)|SAT10  89620130003266723945|6285883352548|510018335374160|TEST/20191113/371978|TEST||IM3|20201130|20191113|920|SP IM3 90D LTE (N)|SAT10|1001|Field ICCID contains invalid data. |

# STARTERPACK ALLOCATION DUMP (INTTNM003)

|  |  |
| --- | --- |
| INTERFACE TYPE: | Off-line |
| INTERFACE METHOD: | Batch file |
| INITIATOR SYSTEM: | TNM |
| RESPONDER SYSTEM: | MOBII |

## Context goal

1. File contains starter pack allocation transactions along with its serial numbers, which is generated for external dealers (Transfer between Indosat warehouse to MPC).
2. In alloc dump file SO\_ID is taken from DO\_ID and combination of SO\_ID and ALLOC\_ID is considered as unique transaction reference number, and file records are grouped based on this.
3. All records from one unique transaction should contain same dealer code.
4. File will be pushed to MOBII path daily before 2AM with latest updated records. MOBII scheduler will pick pushed files for process.
5. Once file is copied to MOBII interface server then file will be deleted from SFTP path.

## Pre-conditions

1. Dealer code, Product master data is created in MOBII and active.

## Post-conditions

1. Given transaction will be registered in MOBII and assigned to dealer to accept from his inbox.
2. Once dealer is approved from his task stock will be added into his inventory.
3. If surrounding system shares the already shared serial numbers once again then MOBII will validate its SO\_ID with previously received SO\_ID, if it is greater than old SO\_ID then stock will be removed from old dealer and allocated to new dealer.

## File naming

SP\_ALLOC\_DUMP\_<YYYYMMDDHH24MISS>.csv.gz

* <YYYYMMDDHH24MISS> = date and time of file generation.
* Sample Actual File Name is SP\_ALLOC\_DUMP\_20200203013512.csv.gz
* Sample Control File Name is SP\_ALLOC\_DUMP\_20200203013512.ctl.gz

## Paths

* SFTP PATH
  + Master: /home/fossuser/interfaces/FOSS/master\_files/sp\_dump/
* MOBII Application
  + Master: /snoc/interfaces/S-NOC/interfaces/master\_files/ sp\_dump/
  + Backup: /snoc/interfaces/S-NOC/interfaces/backup/master\_files/ sp\_dump/
  + Rejected: /snoc/interfaces/S-NOC/interfaces/rejected\_files/ sp\_dump/
  + Rejected Backup: /snoc/interfaces/S-NOC/interfaces/backup/rejected\_files/ sp\_dump/

## Data feed fields

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Field | Mandatory | Data Types | Description |
| 1 | ICCID | Yes | Number | Serial Number |
| 2 | MSISDN | Yes | Number | MSISDN assigned to serial number.  In case of Blank SIM/DSA SIM cards the field will be empty. |
| 3 | IMSI | Yes | Number | IMSI assigned to Serial number. |
| 4 | DO\_ID | Yes | String | DO\_ID in the format – BRAND/YYYYMMDD/Stock Transfer id |
| 5 | DEALER\_ID | Yes | String | External Dealer id. |
| 6 | BRANCH\_CODE | No | Number | Empty |
| 7 | BRAND | Yes | String | IM3 |
| 8 | PRODUCT\_EXPIRED\_DATE | Yes | Date – YYYYMMDD | Expiry date of Product code. |
| 9 | SO\_CREATION\_DATE | Yes | Date – YYYYMMDD | Stock transfer date in TNM |
| 10 | ALLOC\_ID | Yes | Number | Sales Order ID |
| 11 | PROGRAM\_CODE | Yes | Number | Product code ID |
| 12 | PROGRAM\_NAME | Yes | String | Product code description |
| 13 | TYPE | No | Number | Empty |
| 14 | ALLOC\_DATE | Yes | Date – YYYYMMDD | Same as Sales of creation date |
| 15 | PAYMENT\_DATE | No | Date – YYYYMMDD | Empty |

**Example:**

|  |
| --- |
| ICCID|MSISDN|IMSI|DO\_ID|DEALER\_ID|BRANCH\_CODE|BRAND|PRODUCT\_EXPIRED\_DATE|SO\_CREATION\_DATE|ALLOC\_ID|PROGRAM\_CODE|PROGRAM\_NAME|TYPE|ALLOC\_DATE|PAYMENT\_DATE  89620160002764633575|6285658062012|510015867005194|BT40/20191112/2100111684|BT40||IM3|20201031|20191113|232324|927|SF IM3 2000 LTE N||13-NOV-19|| |

### Processing result

File status can be tracked from MOBII web. If any records are rejected, then rejected files can be downloaded from MOBII web which contains all columns of original file and additionally ERROR\_CODE|ERROR\_MESSAGE.

**Example:**

|  |
| --- |
| ICCID|MSISDN|IMSI|DO\_ID|DEALER\_ID|BRANCH\_CODE|BRAND|PRODUCT\_EXPIRED\_DATE|SO\_CREATION\_DATE|ALLOC\_ID|PROGRAM\_CODE|PROGRAM\_NAME|TYPE|ALLOC\_DATE|PAYMENT\_DATE|ERROR\_CODE|ERROR\_MESSAGE  89620160002764633575|6285658062012|510015867005194|BT40/20191112/2100111684|BT40||IM3|20201031|20191113|232324|927|SF IM3 2000 LTE N||||1001|Mandatory field ALLOC\_DATE is missing |

# VOUCHER STOCK DUMP (INTTNM004)

|  |  |
| --- | --- |
| INTERFACE TYPE: | Off-line |
| INTERFACE METHOD: | Batch file |
| INITIATOR SYSTEM: | TNM |
| RESPONDER SYSTEM: | MOBII |

## Context goal

1. File contains voucher transfer transactions along with its serial numbers, which is generated for internal dealers (Example, gallery channel). Transfer between Indosat warehouse to Dealer and Dealer to Dealer both will be shared in file.
2. In stock dump file SO\_ID is considered as transaction reference number, and transactions are grouped based on this.
3. All records from one SO\_ID should contain same source and destination dealer codes.
4. File will be pushed to MOBII path daily before 2AM with latest updated records. MOBII scheduler will pick pushed files for process.
5. Once file is copied to MOBII interface server then file will be deleted from SFTP path.

## Pre-conditions

1. Dealer code, Product master data is created in MOBII and active.

## Post-conditions

1. Given transaction will be registered in MOBII and assigned to dealer to accept from his inbox.
2. Once dealer is approved from his task stock will be added into his inventory.
3. If surrounding system shares the already shared serial numbers once again then MOBII will validate its SO\_ID with previously received SO\_ID, if it is greater than old SO\_ID then stock will be removed from old dealer and allocated to new dealer.

## File naming

VOUCHER\_STOCK\_DUMP\_<YYYYMMDDHH24MISS>.csv.gz

* <YYYYMMDDHH24MISS> = date and time of file generation.
* Sample Actual File Name is VOUCHER\_STOCK\_DUMP\_20200203013512.csv.gz
* Sample Control File Name is VOUCHER\_STOCK\_DUMP\_20200203013512.ctl.gz

## Paths

* SFTP PATH
  + Master: /home/fossuser/interfaces/FOSS/master\_files/vc\_dump/
* MOBII Application
  + Master: /snoc/interfaces/S-NOC/interfaces/master\_files/vc\_dump/
  + Backup: /snoc/interfaces/S-NOC/interfaces/backup/master\_files/vc\_dump/
  + Rejected: /snoc/interfaces/S-NOC/interfaces/rejected\_files/vc\_dump/
  + Rejected Backup: /snoc/interfaces/S-NOC/interfaces/backup/rejected\_files/vc\_dump/

## Data feed fields

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Field | Mandatory | Data Types | Description |
| 1 | SERIAL\_NUMBER | Yes | Number | Serial Number |
| 2 | DO\_ID | Yes | String | DO\_ID in the format – BRAND/YYYYMMDD/Stock Transfer id. |
| 3 | DEALER\_ID | Yes | String | Gallery Id or warehouse id |
| 4 | ARTWORK\_CODE | No | String | Empty |
| 5 | BRANCH\_CODE | No | Number | Empty |
| 6 | BRAND | Yes | String | This field always set to ‘IND’ |
| 7 | PRODUCT\_EXPIRED\_DATE | Yes | Date – YYYYMMDD | Expiry date of Product code. |
| 8 | NOMINAL\_VALUE | Yes | Double | This is associated with Product code. |
| 9 | STOCK\_TRANSFER\_DATE | Yes | Date – YYYYMMDD | Stock transfer date in TNM |
| 10 | PROGRAM\_CODE | Yes | Number | Product code ID |
| 11 | PROGRAM\_NAME | Yes | String | Product code description |
| 12 | SOURCE\_DEALER\_ID | Yes | String | Warehouse id or Gallery ID |

**Example:**

|  |
| --- |
| SERIAL\_NUMBER|DO\_ID|DEALER\_ID|ARTWORK\_CODE|BRANCH\_CODE|BRAND|PRODUCT\_EXPIRED\_DATE|NOMINAL\_VALUE|STOCK\_TRANSFER\_DATE|PROGRAM\_CODE|PROGRAM\_NAME|SOURCE\_DEALER\_ID  1000683641250|GIFT/20191113/372008|||008|IND|20201031|0|20191113|970|VCR DATA FI 2GB SUMATERA|SAT10  1000683641251|GIFT/20191113/372008|||008|IND|20201031|0|20191113|970|VCR DATA FI 2GB SUMATERA|SAT10 |

### Processing result

File status can be tracked from MOBII web. If any records are rejected, then rejected files can be downloaded from MOBII web which contains all columns of original file and additionally ERROR\_CODE|ERROR\_MESSAGE.

**Example:**

|  |
| --- |
| SERIAL\_NUMBER|DO\_ID|DEALER\_ID|ARTWORK\_CODE|BRANCH\_CODE|BRAND|PRODUCT\_EXPIRED\_DATE|NOMINAL\_VALUE|STOCK\_TRANSFER\_DATE|PROGRAM\_CODE|PROGRAM\_NAME|SOURCE DEALER ID|ERROR\_CODE|ERROR\_MESSAGE  1000683641250|GIFT/20191113/372008|||008|IND|20201031|0|20191113|970|VCR DATA FI 2GB SUMATERA|SAT10|1001|Mandatory field SOURCE DEALER ID is missing |

# VOUCHER ALLOCATION DUMP (INTTNM005)

|  |  |
| --- | --- |
| INTERFACE TYPE: | Off-line |
| INTERFACE METHOD: | Batch file |
| INITIATOR SYSTEM: | TNM |
| RESPONDER SYSTEM: | MOBII |

## Context goal

1. File contains voucher allocation transactions along with its serial numbers, which is generated for external dealers (Transfer between Indosat warehouse to MPC).
2. In alloc dump file SO\_ID is taken from DO\_ID and combination of SO\_ID and ALLOC\_ID is considered as unique transaction reference number, and file records are grouped based on this.
3. All records from one unique transaction should contain same dealer code.
4. File will be pushed to MOBII path daily before 2AM with latest updated records. MOBII scheduler will pick pushed files for process.
5. Once file is copied to MOBII interface server then file will be deleted from SFTP path.

## Pre-conditions

1. Dealer code, Product master data is created in MOBII and active.

## Post-conditions

1. Given transaction will be registered in MOBII and assigned to dealer to accept from his inbox.
2. Once dealer is approved from his task stock will be added into his inventory.
3. If surrounding system shares the already shared serial numbers once again then MOBII will validate its SO\_ID with previously received SO\_ID, if it is greater than old SO\_ID then stock will be removed from old dealer and allocated to new dealer.

## File naming

VOUCHER\_ALLOC\_DUMP\_<YYYYMMDDHH24MISS>.csv.gz

* <YYYYMMDDHH24MISS> = date and time of file generation.
* Sample Actual File Name is VOUCHER\_ALLOC\_DUMP\_20200203013512.csv.gz
* Sample Control File Name is VOUCHER\_ALLOC\_DUMP\_20200203013512.ctl.gz

## Paths

* SFTP PATH
  + Master: /home/fossuser/interfaces/FOSS/master\_files/vc\_dump/
* MOBII Application
  + Master: /snoc/interfaces/S-NOC/interfaces/master\_files/vc\_dump/
  + Backup: /snoc/interfaces/S-NOC/interfaces/backup/master\_files/vc\_dump/
  + Rejected: /snoc/interfaces/S-NOC/interfaces/rejected\_files/vc\_dump/
  + Rejected Backup: /snoc/interfaces/S-NOC/interfaces/backup/rejected\_files/vc\_dump/

## Data feed fields

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Field | Mandatory | Data Types | Description |
| 1 | SERIAL\_NUMBER | Yes | Number | Serial Number |
| 2 | DO\_ID | Yes | String | DO\_ID in the format – BRAND/YYYYMMDD/Stock Transfer id |
| 3 | DEALER\_ID | Yes | String | External Dealer id. |
| 4 | ARTWORK | No | Number | Empty |
| 5 | BRANCH\_CODE | No | String | Empty |
| 6 | BRAND | Yes | String | IM3 |
| 7 | PRODUCT\_EXPIRED\_DATE | Yes | Date – YYYYMMDD | Expiry date of Product code. |
| 8 | NOMINAL\_VALUE | Yes | Double | This is associated with Product code. |
| 9 | SO\_CREATION\_DATE | Yes | Date – YYYYMMDD | Stock transfer date in TNM |
| 10 | ALLOC\_NO | Yes | Number | Sales order Id |
| 11 | PROGRAM\_CODE | Yes | Number | Product code ID |
| 12 | PROGRAM\_NAME | Yes | String | Product code description |
| 13 | RESOURCE\_TYPE | No | Number | Empty |
| 14 | ALLOC\_DATE | Yes | Date – YYYYMMDD | Same as Sales of creation date |
| 15 | PAYMENT\_DATE | No | Date – YYYYMMDD | Empty |

**Example:**

|  |
| --- |
| SERIAL\_NUMBER|DO\_ID|DEALER\_ID|ARTWORK\_CODE|BRANCH\_CODE|BRAND|PRODUCT\_EXPIRED\_DATE|NOMINAL\_VALUE|SO\_CREATION\_DATE|ALLOC\_NO|PROGRAM\_CODE|PROGRAM\_NAME|RESOURCE\_TYPE|ALLOC\_DATE|PAYMENT\_DATE  1000683697500|EM37/20191107/1200077042|EM37|||IND|20201031|0|20191107|174403|971|VCR DATA FI 2GB JAB & SUK||07-NOV-19|  1000683697501|EM37/20191107/1200077042|EM37|||IND|20201031|0|20191107|174403|971|VCR DATA FI 2GB JAB & SUK||07-NOV-19| |

### Processing result

File status can be tracked from MOBII web. If any records are rejected, then rejected files can be downloaded from MOBII web which contains all columns of original file and additionally ERROR\_CODE|ERROR\_MESSAGE.

**Example:**

|  |
| --- |
| SERIAL\_NUMBER|DO\_ID|DEALER\_ID|ARTWORK\_CODE|BRANCH\_CODE|BRAND|PRODUCT\_EXPIRED\_DATE|NOMINAL\_VALUE|SO\_CREATION\_DATE|ALLOC\_NO|PROGRAM\_CODE|PROGRAM\_NAME|RESOURCE\_TYPE|ALLOC\_DATE|PAYMENT\_DATE|ERRORC\_CODE|ERROR\_MESSAGE  1000683697500|EM37/20191107/1200077042|EM37|||IND|20201031|0|20191107|174403|971|VCR DATA FI 2GB JAB & SUK||07-NOV-19|1001|Mandatory field DEALER\_ID is missing |

# EXTEND EXPIRY DATE (INTTNM006)

|  |  |
| --- | --- |
| INTERFACE TYPE: | Off-line |
| INTERFACE METHOD: | Batch file |
| INITIATOR SYSTEM: | TNM |
| RESPONDER SYSTEM: | MOBII |

## Context goal

1. TNM system will generate a file for when any update in serial number expiry date or changes in previously shared product code.
2. File will be pushed to MOBII path daily before 2AM with latest updated records. MOBII scheduler will pick pushed files for process.
3. Once file is copied to MOBII interface server then file will be deleted from SFTP path.

## Pre-conditions

1. Serial number and product code must exist in MOBII.

## Post-conditions

1. MOBII will update the given product code and expiry date to given serial number.

## File naming

EXPIRY\_DETAILS\_<YYYYMMDDHH24MISS>.csv.gz

* <YYYYMMDDHH24MISS> = date and time of file generation.
* Sample Actual File Name is EXPIRY\_DETAILS\_20200203013512.csv.gz
* Sample Control File Name is EXPIRY\_DETAILS\_20200203013512.ctl.gz

## Paths

* SFTP PATH
  + Master: /home/fossuser/interfaces/FOSS/master\_files/expiry\_details/
* MOBII Application
  + Master: /snoc/interfaces/S-NOC/interfaces/master\_files/expiry\_details/
  + Backup: /snoc/interfaces/S-NOC/interfaces/backup/master\_files/expiry\_details/
  + Rejected: /snoc/interfaces/S-NOC/interfaces/rejected\_files/expiry\_details/
  + Rejected Backup: /snoc/interfaces/S-NOC/interfaces/backup/ rejected\_files/expiry\_details/

## Data feed fields

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Field | Mandatory | Data Types | Description |
| 1 | SERIAL\_NUMBER\_SIM | Yes | Number | Voucher /sim Serial Number |
| 2 | NEW\_PRODUCT\_CODE | Yes | ID | Product code. |
| 3 | NEW\_EXPIRY\_DATE | Yes | Date – YYYYMMDD | Updated expiry date. |

**Example:**

|  |
| --- |
| SERIAL\_NUMBER\_SIM|NEW\_PRODUCT\_CODE|EXPIRY\_DATE  89620130002025749423|919|2019-12-31  89620130002025749431|919|2019-12-31 |

### Processing Result

File status can be tracked from MOBII web. If any records are rejected, then rejected files can be downloaded from MOBII web which contains all columns of original file and additionally ERROR\_CODE|ERROR\_MESSAGE.

**Example:**

|  |
| --- |
| SERIAL\_NUMBER\_SIM|NEW\_PRODUCT\_CODE|EXPIRY\_DATE|ERROR\_CODE|ERROR\_MESSAGE  |919|2019-12-31|1001|Mandatory field Serial Number is missing |

# QUERY STOCK (INTTNM007)

|  |  |
| --- | --- |
| INTERFACE TYPE: | On-line API |
| INTERFACE METHOD: | HTTP GET |
| INITIATOR SYSTEM: | MOBII |
| RESPONDER SYSTEM: | TNM |

## Actors, surrounding systems involved

* TNM application as actor

## Context goal

1. This API returns the available stock quantities for one or more Product Codes which are specified as a comma separated list in the query parameter. It will return success if at least 1 product is found. The response message will only contain the stock quantities for the found Product Code(s).

## Pre-conditions

1. API is ready and running in TNM server, server is accessible from MOBII server.

## Post-conditions

1. Expected products available quantity is returned in response body else reason will be provided for the same.

## Endpoint URL

http://{tnm-server}/mobii/stock

## request & response

### Request

http://{tnm-server}/mobi/stock?code=971,972,973

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Data Type | Parameter Type | Required | Description |
| Code | integer | query | Yes | Product Code. Multiple codes are to be separated using comma |

### RESPONSE

The API processing is considered Success if at least one Product Code is found valid (even if quantity is zero). In any case, the processing results of all products, success or otherwise (indicated in a boolean flag), will be included in the response payload. For valid products the quantity will be returned, otherwise the failure reason will be returned.

The payload object contains the following output fields:

|  |  |  |
| --- | --- | --- |
| Name | Data Type | Description |
| stockAvail | array of objects | Array of objects where each object contains the following fields: |
| prodCode | Integer | Product Code. |
| isValid | Boolean | A valid product code? true or false. |
| quantity | Integer | Optional available product quantity, exists for valid product only. |
| reason | String | Optional reason text string for failure, exists for invalid product only. |

### Rejection types & errors

|  |  |
| --- | --- |
| Result Code Range | Remark |
| 0 | Success |
| -1 | For uncaught error which is not in any error definitions below. |
| 4000 – 4999 | User input error, e.g. missing or invalid input parameter values. The exact reason will be provided in the response message. |
| 5000 – 5499 | Processing failed due to missing or incomplete application setup. e.g. missing configuration data. A generic error message will be provided in the response message. |
| 5500 – 5999 | Processing failed due to system internal or run-time errors, e.g. null-pointer exception. A generic message will be provided in the response message. |
| 6000 – 6499 | Processing could not be initiated, and corrective action is required from the client program, e.g. User session timeout. The exact reason will be provided in the response message. |

**Request example:**

|  |
| --- |
| http://{tnm-server}/mobi/stock?code=971,972,973 |

**Response example:**

|  |
| --- |
| {  "resultCode”: 0,  "message": "Success",  "payload": {  "stockAvail": [  {  "prodCode": 971,  "isValid": true,  "quantity": 9999  },  {  "prodCode": 988,  “isValid”: false,  "reason": “Does not exist in TNM”  },  {  "prodCode": 989,  “isValid”: false,  "reason": “Does not exist in TNM”  }  ]  }  } |

**Failure example:** Incase none of the product not found in TNM

|  |
| --- |
| {  "resultCode”: 4106,  "message": "None of the input Product Codes is valid",  "payload": {  "stockAvail": [  {  "prodCode": 971,  "isValid": false,  "reason": “Does not exist in TNM”  },  {  "prodCode": 988,  “isValid”: false,  "reason": “Does not exist in TNM”  },  {  "prodCode": 989,  “isValid”: false,  "reason": “Does not exist in TNM”  }  ]  }  } |

# RESERVE STOCK (INTTNM008)

|  |  |
| --- | --- |
| INTERFACE TYPE: | On-line API |
| INTERFACE METHOD: | JSON POST |
| INITIATOR SYSTEM: | MOBII |
| RESPONDER SYSTEM: | TNM |

## Actors, surrounding systems involved

* TNM application as actor

## Context goal

1. This API reserves stock for a single dealer for one or more product codes. For the API to succeed all reservation requests must be fulfilled completely. Partial reservation is not accepted.

## Pre-conditions

1. API is ready and running in TNM server, server is accessible from MOBII server.

## Post-conditions

1. Specified stock will be registered for given dealer code.

## Endpoint URL

http://{tnm-server}/mobi/stock/resv

## request & response

### Request

The following input fields are to be supplied in the request body RESPONSE

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data Type | Mandatory | Description |
| dealerId | String | Yes | Dealer ID. |
| stockResv | array of object | Yes | Array of reservation objects, where each object contains the below 2 fields: |
| prodCode | Integer | Yes | Product Code. |
| quantity | Integer | Yes | Quantity to reserve. |

### RESPONSE

The API processing is considered Success if all products' requested quantities are successfully reserved. In this case, the response message shall contain the "Success" indicators only without payload.

|  |  |  |
| --- | --- | --- |
| Name | Data Type | Description |
| resultCode | Integer | 0 – success |
| Message | String | Custom response message |

### Rejection types & errors

|  |  |
| --- | --- |
| Result Code Range | Remark |
| 0 | Success |
| 4001 | Missing required parameter <param-name>. |
| 4003 | <param-name> <param-value> not found in the system. |
| 4107 | Reservation failed for product <prodCode>: <reason> |

**Request example:**

|  |
| --- |
| {  "dealerId": "BR020",  "stockResv": [  {  "prodCode": 971,  "quantity": 9999  },  {  "prodCode": 972,  "quantity": 1235  },  {  "prodCode": 975,  "quantity": 50  }  ]  } |

**Response example:**

|  |
| --- |
| {  "resultCode”: 0,  "message": "Success"  } |

**Failure Example:**

|  |
| --- |
| {  "resultCode”: 4001,  "message": "Missing required parameter dealerId"  } |

# APPROVAL SIGN-OFF

I have received soft copy of current version document via e-mail, reviewed and approved confirm with reply to e-mail and as undersigned, I change request and agree that it’s appropriate:

|  |  |  |
| --- | --- | --- |
| 1. **PARTHIPAN RRAJAGOPAL** Business Analytic (ENHANCESYS) |  |  |
|  | Signature | Date |
| 1. **ROMET ROOSALU** Project Manager (TIMWETECH) |  |  |
|  | Signature | Date |
| 1. **EDO ASRI** Project Manager (INDOSAT) |  |  |
|  | Signature | Date |
| 1. **SANTI ROSANA** IT-DevOps (INDOSAT) |  |  |
|  | Signature | Date |