TRIPMAZA
AIR API DOCUMENTATION

Table of Contents:

Postman Collection

Link to Download Postman Collection :
https://www.getpostman.com/collections/a8c29e67e54f96c29201

Step : 1 : Download the content from Postman Link and save with extension (.json)
Step : 2 : Export the json file to Postman Application
Step : 3: Create Environment File with "apikey" (shared user api key) and host (st-agent.tripmaza.com).

Sample Format for Certification logs - Sample Logs

1. Introduction

Tripmaza API exports a REST based web service over HTTPS, using JSON to encode service request and response data. We use an SSL secured channel to transmit data from client to the service.It is intended specially for developers and other team members to help them know how the
Air API Service works. This guide provides all the required information to integrate this
TRIPMAZA.COM Air API.
All the steps right from search, to review,seat sell, additional ssr, fare rules and final booking are sequentially followed by using a shared api key header ("apikey").

apikey – Your sandbox or production API key

2. Sample Code

1. Java

```java
protected String getApiReponse(String request, String apiUrl) {
    Proxy proxy = null;
    URL endPoint = null;
    String responseBody = null;
    try {
        endPoint = new URL(apiUrl);
        URLConnection urlConnection = endPoint.openConnection();
        setHeaderParams(urlConnection);
        setPostData(urlConnection, request);
        urlConnection.connect();
        responseBody = getReponseContent(urlConnection);
    } catch (IOException ioe) {
        ioe.printStackTrace();
    } finally {

    }
    return responseBody;
}

private void setHeaderParams(URLConnection urlConnection) {
    urlConnection.setRequestProperty("content-type", "application/json");
    urlConnection.setRequestProperty("apikey", "xxxx-yyyy-zzzz");
}

private void setPostData(URLConnection urlConnection, String request) throws IOException {
    urlConnection.setDoOutput(true);
    OutputStream outStream = urlConnection.getOutputStream();
    OutputStreamWriter osw = new OutputStreamWriter(outStream);
    osw.write(request.toString(),  off: 0, request.toString().length());
    osw.flush();
    osw.close();
}
```

2. Python

```python
import requests
import json

apikey="*****-*****-*****"
api_url="https://st-agent.tripmaza.com"
headers={"apikey":apikey,"Content-Type":"application/json"}


def doSearch(base_url,apikey,endpoint,search_request):
    print("Searching flight list.....")
    url = base_url + endpoint
    response = requests.post(url,data=json.dumps(search_request),headers=headers)
    if response.ok:
        print(response.json())
        #parse search response
        return response


if __name__ == "__main__":
    search_request = {
    "searchQuery": {
        "cabinClass": "ECONOMY",
        "paxInfo": {
            "ADULT": "1",
            "CHILD": "0",
            "INFANT": "0"
        },
        "routeInfos": [
            {
                "fromCityOrAirport": {
                    "code": "CCU"
                },
                "toCityOrAirport": {
                    "code": "DEL"
                },
                "travelDate": "2021-05-19"
            }
        ],
        "searchModifiers": {}
        }
    }
    doSearch(api_url,apikey,'fms/v1/air-search-all',search_request)
```

3. PHP

```php
function callAPI($method, $url, $data){
    $curl = curl_init();
    switch ($method){
        case "POST":
            curl_setopt($curl, CURLOPT_POST, 1);
            if ($data)
                curl_setopt($curl, CURLOPT_POSTFIELDS, $data);
            break;
        default:
            if ($data)
                $url = sprintf("%s?%s", $url, http_build_query($data));
    }
    // OPTIONS:
    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_HTTPHEADER, array(
        'apikey: 11111111111111111111',
        'Content-Type: application/json',
    ));
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, 1);
    curl_setopt($curl, CURLOPT_HTTPAUTH, CURLAUTH_BASIC);
    // EXECUTE:
    $result = curl_exec($curl);
    if(!$result){die("Connection Failure");}
    curl_close($curl);
    return $result;
}
```

3. Service Definition

Below are the available service URL & definition:-

UAT Environment:

Protocol : https

Domain: st-agent.tripmaza.com

Service Name	URL
Search Service	https://st-agent.tripmaza.com/fms/v1/air-search-all
FareRule Service	https://st-agent.tripmaza.com/fms/v1/farerule
Review Service	https://st-agent.tripmaza.com/fms/v1/review
Book Service (Instant Ticket)	https://st-agent.tripmaza.com/oms/v1/air/book
Book Service (Hold)	https://st-agent.tripmaza.com/oms/v1/air/book
Confirm Fare Before Ticket	https://st-agent.tripmaza.com/oms/v1/air/fare-validate
Book Service (Confirm Hold Book)	https://st-agent.tripmaza.com/oms/v1/air/confirm-book
Booking Details	https://st-agent.tripmaza.com/oms/v1/booking-details
Release PNR(Hold)	https://st-agent.tripmaza.com/oms/v1/air/unhold
Seat Service	https://st-agent.tripmaza.com/fms/v1/seat
Get Amendment Charges	https://st-agent.tripmaza.com/oms/v1/air/amendment/amendment-charges
Submit Amendment	https://st-agent.tripmaza.com/oms/v1/air/amendment/submit-amendment
Amendment Details	https://st-agent.tripmaza.com/oms/v1/air/amendment/amendment-details

**Note : End point never ends with "/" otherwise System will throw an error.**

Production Environment:

Protocol : https

Domain: book.tripmaza.com

| Service Name	URL
Search Service	https://book.tripmaza.com/fms/v1/air-search-all
FareRule Service	https://book.tripmaza.com/fms/v1/farerule
Review Service	https://book.tripmaza.com/fms/v1/review
Book Service (Instant Ticket)	https://book.tripmaza.com/oms/v1/air/book
Book Service (Hold)	https://book.tripmaza.com/oms/v1/air/book
Confirm Fare Before Ticket	https://book.tripmaza.com/oms/v1/air/fare-validate
Book Service (Confirm Hold Book)	https://book.tripmaza.com/oms/v1/air/confirm-book
Booking Details	https://book.tripmaza.com/oms/v1/booking-details
Release PNR(Hold)	https://book.tripmaza.com/oms/v1/air/unhold
Seat Service	https://book.tripmaza.com/fms/v1/seat
Get Amendment Charges	https://book.tripmaza.com/oms/v1/air/amendment/amendment-charges
Submit Amendment	https://book.tripmaza.com/oms/v1/air/amendment/submit-amendment
Amendment Details	https://book.tripmaza.com/oms/v1/air/amendment/amendment-details

Note : End point never ends with "/" otherwise System will throw an error.

4. Web Service flow

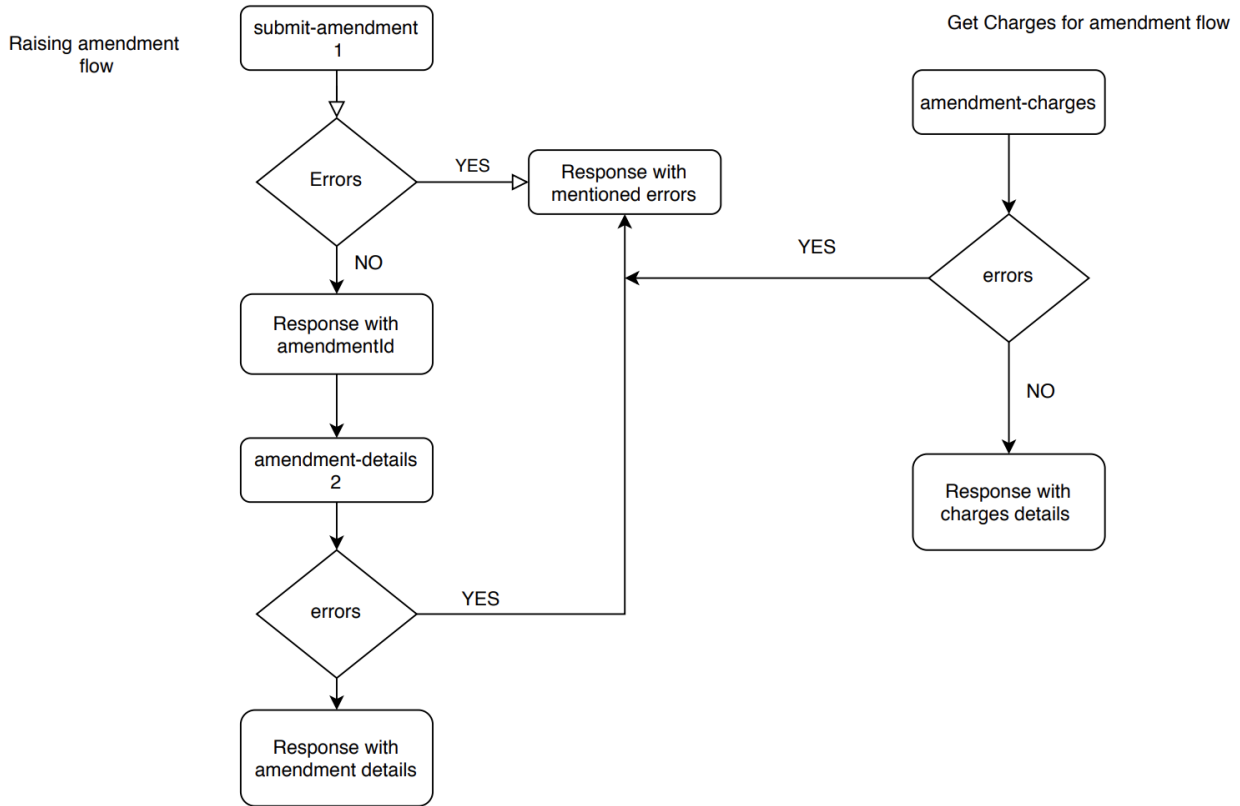### 1. Search (Without SSR)
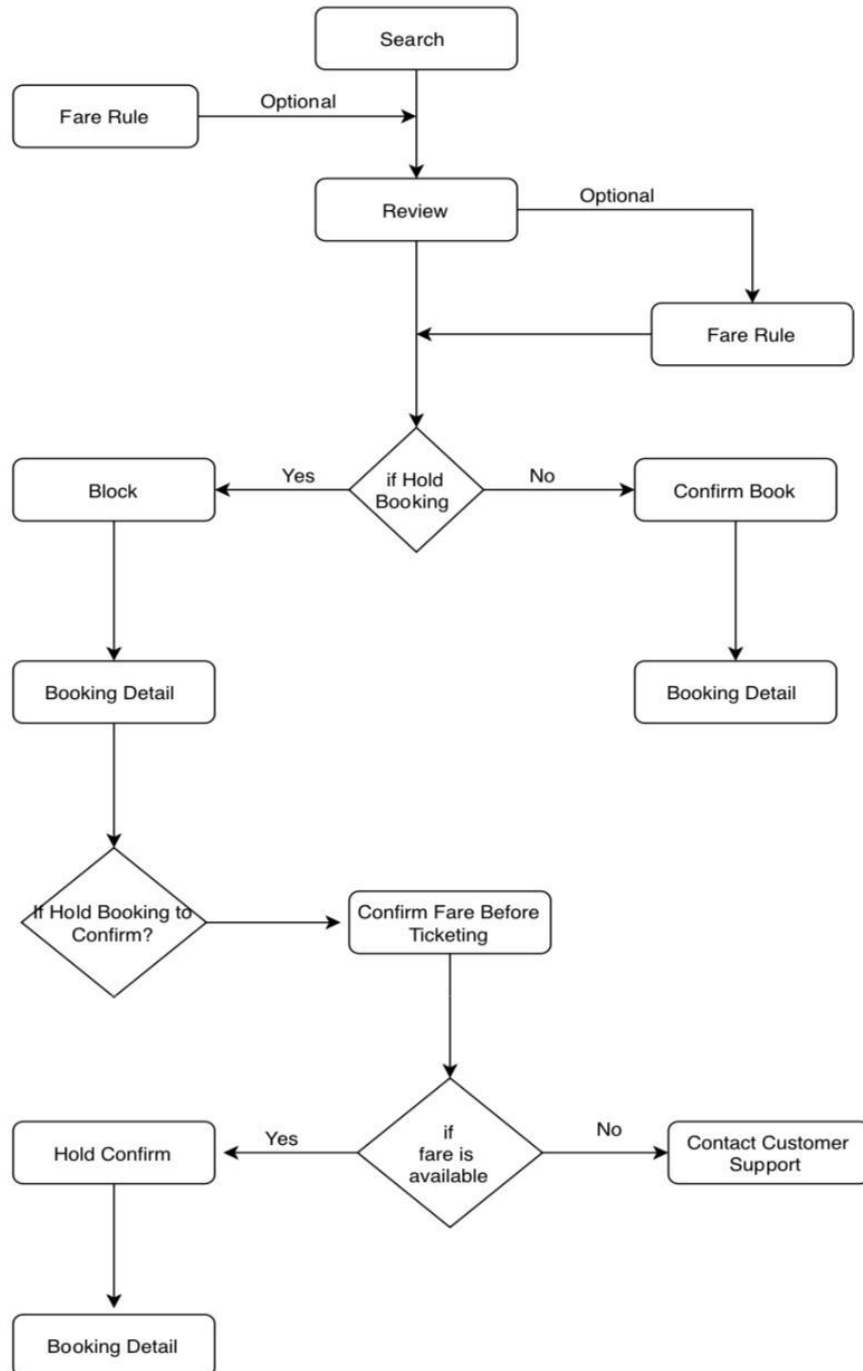
2. Search (With SSR)
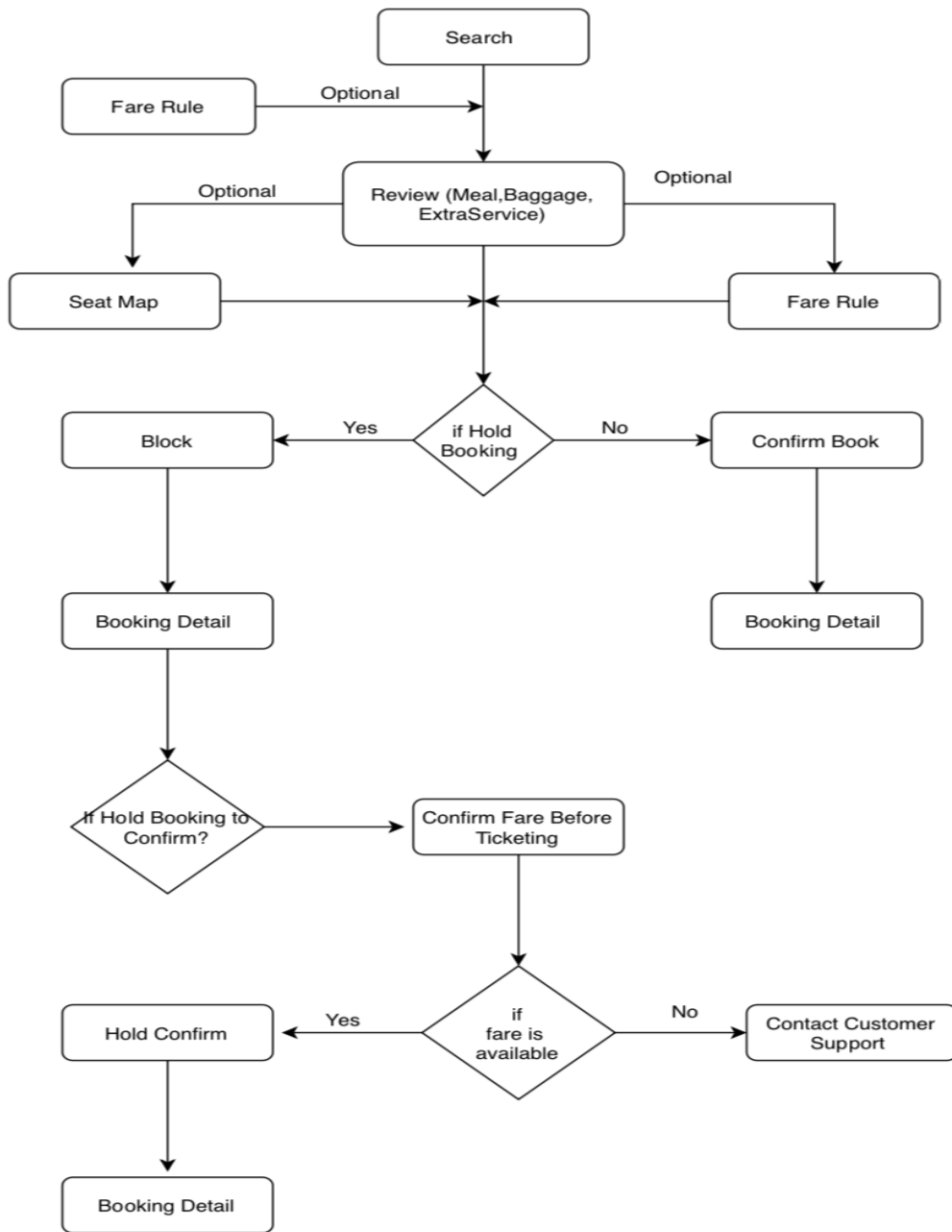
* Booking Detail Should be called elapsed 5 seconds

* Hold Booking will be decided from review response

3.CANCELLATION AMENDMENT API

Raising amendment flow

```
submit-amendment
1
        |
        v
      /Errors\ ---YES---> Response with
      \      /            mentioned errors
        |NO                   ^
        v                     |
Response with                |
amendmentId                  |
        |                     |
        v                     |
amendment-details            |
2                            |
        |                     |
        v                     |
      /errors\ ---YES---------+
      \      /
        |NO
        v
Response with
amendment details
```

Get Charges for amendment flow

```
amendment-charges
        |
        v
      /errors\ ---YES---> (to Response with
      \      /            mentioned errors)
        |NO
        v
Response with
charges details
```

5.FLIGHT SERVICE

1. SEARCH
2. REVIEW
3. FARE RULE
4. SEAT MAP
5. BOOKING API
   a. INSTANT BOOK
   b. HOLD
   c. CONFIRM FARE BEFORE TICKET
   d. CONFIRM-BOOK (TICKETING AFTER HOLD)
6. RETRIEVE BOOKING
7. RELEASE PNR
8. AMENDMENT API
   a. Get Charges
   b. Cancellation Amendment

Complete Booking Log : (2Adult -1Child -1Infant)
Sample logs Link :
Click Here to Take Complete Flow - Sample Logs

5.1 SEARCH

The following section describes the availability method exposed via the API to search.

This method provides availability information with the cheapest fare option available.The request includes a single query for all cases (Oneway,Return,Multi-city). When the request is requested,the system starts validating and the search engine starts searching for the search query and will **return the best price options available SearchResults from all suppliers**.

```
API : air-search-all
Request : Search Request
Response : Search Result (All Supplier Search
Results)
```
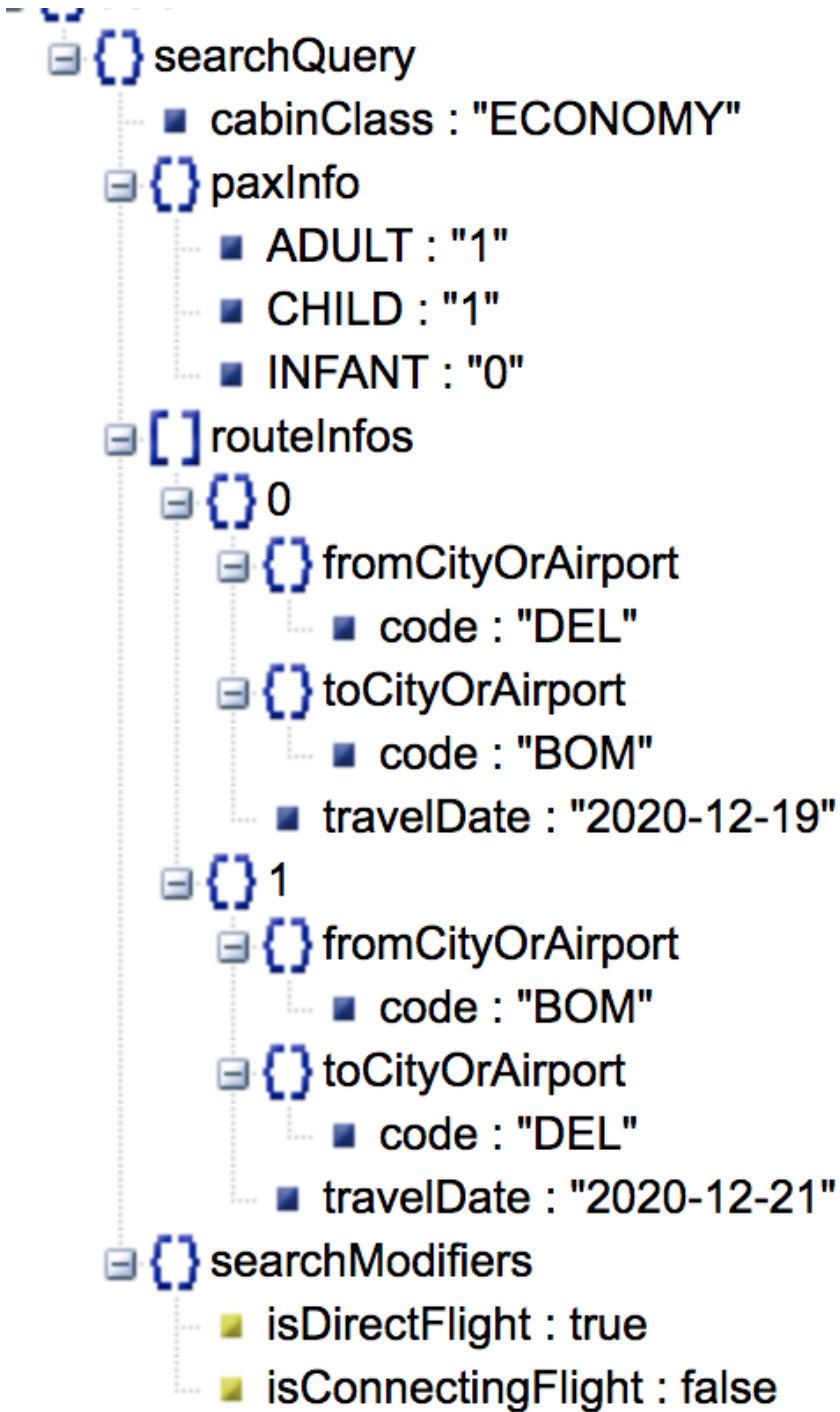
URL: https://st-agent.tripmaza.com/fms/v1/air-search-all
Request Method: POST

Sample logs Link : All Examples for Search (Oneway,Return,Multi-city)

*Please change dates in sample request

Sample Request View :

```
   searchQuery
       cabinClass : "ECONOMY"
       paxInfo
           ADULT : "1"
           CHILD : "1"
           INFANT : "0"
       routeInfos
           0
               fromCityOrAirport
                   code : "DEL"
               toCityOrAirport
                   code : "BOM"
               travelDate : "2020-12-19"
           1
               fromCityOrAirport
                   code : "BOM"
               toCityOrAirport
                   code : "DEL"
               travelDate : "2020-12-21"
       searchModifiers
           isDirectFlight : true
           isConnectingFlight : false
```

* Travel Date Format : YYYY-MM-DD

CabinClass Allowed values: (Default Value : ECONOMY)

1. ECONOMY
2. PREMIUM_ECONOMY
3. BUSINESS
4. FIRST

Route Info

1. Incase of **Domestic One Way or International One Way** - 1 route info with fromCityOrAirport and toCityOrAirport along with travelDate should be there.
2. Incase of **Domestic Return or International Return** - 2 route info with fromCityOrAirport and toCityOrAirport along with travelDate should be there.
3. InCase of **Domestic Multi-City or International Multi-City** - 2-6 route infos with fromCityOrAirport and toCityOrAirport along with travelDate should be there.
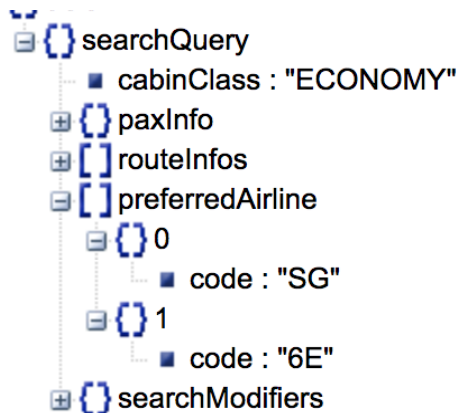
* Travel Date Format : YYYY-MM-DD

Search Modifiers :

1. If **Only Direct Flight** Required then isDirectFlight : true and isConnectingFlight : false
2. If **Only Connecting Flight** Required then isConnectingFlight : true and isDirectFlight : false
3. If **Both Direct & Connecting Flight** Required - then fields are not required to send or both should be true in request.

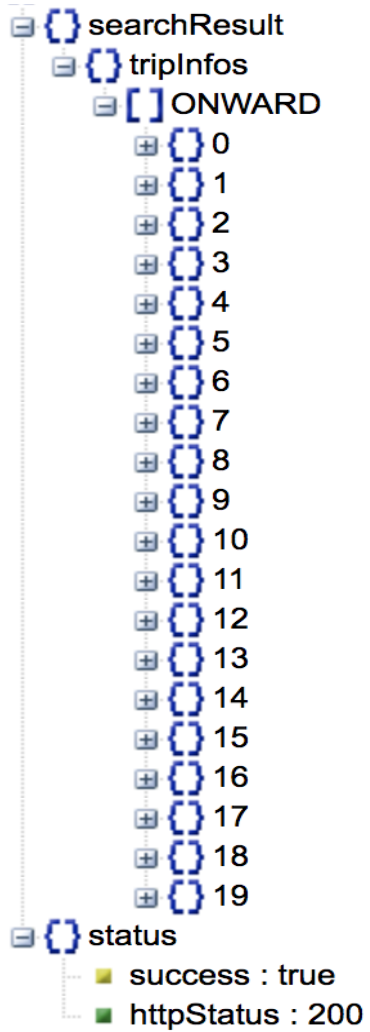**Preferred Carrier (Sample Logs Link):** Sample Request Link

1.preferredAirline (field) - which supports multiple airline code as preferred airline to search.(List<String>)

```
searchQuery
    cabinClass : "ECONOMY"
    paxInfo
    routeInfos
    preferredAirline
        0
            code : "SG"
        1
            code : "6E"
    searchModifiers
```

Request Payload Structure :

Key	Parent	Type	Description	Mandatory/ Optional
searchQuery	root	Object		Mandatory
cabinClass	searchQuery	String	CabinClass to Search	Optional
paxInfo	searchQuery	Object	Pax count	Mandatory
ADULT	paxInfo	String		Mandatory
CHILD	paxInfo	String		Optional
INFANT	paxInfo	String		Optional
preferredAirline	searchQuery	ArrayList		Optional
code	preferredAirline	String		Optional
searchModifiers	searchQuery	Object		Optional
isDirectFlight	searchModifiers	Boolean		Optional
isConnectingFlight	searchModifiers	Boolean		Optional
routeInfos	searchQuery	ArrayList		Mandatory
fromCityOrAirport	routeInfos	Object		Mandatory
code	fromCityOrAirport	String		Mandatory
toCityOrAirport	routeInfos	Object		Mandatory
code	toCityOrAirport	String		Mandatory
travelDate	routeInfos	String	YYYY-MM-DD	Mandatory

Response Sample View:

```
searchResult
   tripInfos
      [ ] ONWARD
         0
         1
         2
         3
         4
         5
         6
         7
         8
         9
         10
         11
         12
         13
         14
         15
         16
         17
         18
         19
   status
      success : true
      httpStatus : 200
```

2. In Case Of No Search Result From Supplier , then Response will be returned (only httpStatus 200).

```
status
   success : true
   httpStatus : 200
```
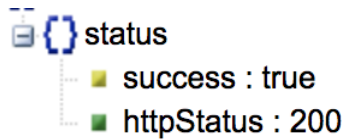
SearchResult Structure :

1. In the case of **Domestic One way or International Oneway Search** : Search Result will have an "ONWARD" key which will have tripInfos (List<Trips>).
2. In case of **Domestic Return** : Search Result will have "ONWARD" and "RETURN" key which will have corresponding Oneway and Return Search Results.
3. In case of **International Return or International Multi-City :** Search Result will have a "COMBO" key which will have corresponding Combined Requested Route Infos.
4. In the case of **Domestic Multi-City :** Equivalent Number of Route Infos , Equivalent index ids generated. (Each index id belongs to each route info).

   Example : (3 Route infos of multi-city.)
   DEL - BOM - (21-02-2022) & BOM - BLR (26-02-2022) , BLR - GOI (02-03-2022) - 3 index ids will be created in order of routeinfo.
   Each Id belongs to corresponding route info (**in search it will be treated each route info as Multiple one ways**)**.**

Response:

| Key	Parent	Type	Description	Mandatory/ Optional
searchResult	root	Object		
tripInfos	searchResult	Object		
ONWARD	tripInfos	ArrayList	Onward Info	
RETURN	tripInfos	ArrayList	Return Info	
COMBO	tripInfos	ArrayList	Combined Info (International Return/Multi city)	

ONWARD/RETURN/COMBO - tripInfos

Key	Parent	Type	Description	Mandatory/ Optional
sl	ONWARD/RETURN/ COMBO	ArrayList	Segment Information	Mandatory
totalPriceList	ONWARD/RETURN/	ArrayList		

	COMBO	(priceInfo)		

TripInfo Brief : - Each TripInfo which has sI (segments information and totalPriceList of trip).

sI (Segment Information) - List<Segments> : which has control of the fields.

➔ **fD** (Flight Designator) - which has Marketing Airline Information , Airline Equipment type, flight number).
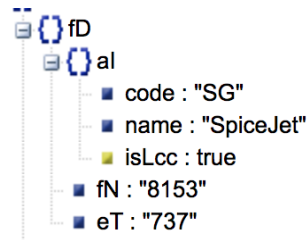
- ● al - airline info which has code (IATA Airline Code, followed with Airline name).
- ● fN - Marketing Airline Flight Number
- ● eT - Marketing Airline Equipment Type
➔ **stops** - Number of Stops in this segment.
➔ **duration -** Segment Journey duration (In Minutes)
➔ **land -** is this flight arriving the next day.
➔ **isRs -** isReturnSegment will be true and this field is present only when the segment is return flight.
➔ **sN -** segment number
➔ **dt -** Departure Time of this segment (YYYY-MM-SS HH:MM Format)
➔ **at -** Arrival time of this segment (YYYY-MM-SS HH:MM Format)
➔ **aa -** Arrival Airport Information (which has IATA Airport code,name,country,**Terminal**)
➔ **da -** Departure Airport Information (which has IATA Airport code,name,country,**Terminal**)
➔ **oB -** Operating Airline which has code (IATA Airline Code, followed with Airline name).
➔ **so -** Stop Over airports (which has IATA Airport code,name,country)
➔ **cT -** Connecting Time (LayOver time) - In Minutes
  .

totalPriceList - Under TripInfo - This is Summarised PriceInformation about the trip

➔ **fareIdentifier** - what type of fare is this. PUBLISHED, SPECIAL_RETURN,...
➔ **Id - Unique Trip price Id (which is used for reviewing/revalidating the price which is going to be sold and booked) in upcoming requests and valid for 30 Minutes from search.**
➔ **sri -** special return identifier
➔ **msri -** matched special return identifier (Special Return Explained Below)
➔ **fD -** Fare Detail in **Per PaxType Wise -** (Example : 2Adult, 1Child Search then have to multiply by 2 for Adult Fare and multiply by 1 for Child Fare)

- ◆ **ADULT/CHILD/INFANT -** will be key (Each PayType - Per Pax Fare)
- ◆ **fC -** fare Components of Fare to PaxType -
    - TAF - Taxes and Fees
    - NF - Net Fare
    - NCM - Net Commission
    - BF - Base Fare
    - TF - Total Fare
- ◆ **afc** - additional fareComponents in Detail - Break Up of fC Component
    - TAF - Break Up of Taxes and Fees from fC
        - OT - Other Charges
        - MF - Management Fee
        - MFT - Management Fee Tax
        - AGST - Airline GST Component
        - YQ - Fuel Surcharge
        - YR - Carrier Misc Fee
    - NCM - Break Up of Net Commission
        - OT - Gross Commission
        - TDS - TDS
- ◆ **sR** - Seats Remaining
- ◆ **bl** - Baggage information (Hand Baggage & Checking baggage Related)
    - iB - Checking Baggage (No format specific, example :20Kg , 20 Kg, 1 Piece , 1 Piece, 20 Kilograms)
    - cB - Cabin Baggage
- ◆ **rT** - Refundable Type (Integer type as below follows)
    - 0 - Non Refundable
    - 1 - Refundable
    - 2 - Partial Refundable
- ◆ **cc** - Cabin Class Fare
    - ECONOMY
    - PREMIUM_ECONOMY
    - BUSINESS
    - FIRST
- ◆ **cB -** Class Of Booking
- ◆ **fB** - Fare Basis
- ◆ **mi -** Meal Indicator (Boolean Type)
    - **True -** Free Meal
    - **False -** Paid Meal

Special Return : (Applicable in case of Domestic Return)

SPECIAL RETURN where the onward and return journeys fare filed as Special Return and which needs to be sold as single fare and booking also will be created Single PNR of Onward and Return Journey (where it has different cancellation policies and restrictions in journey wise cancellation and penalties are subject to airlines and Tripmaza).

If the Fare Identifier of one Journey is "SPECIAL_RETURN", then another journey's Fare Identifier should also be of "SPECIAL_RETURN" type.

This will be identified by Object from **totalPriceList** which has **sri** (special return identifier) and **msri** (matched special return identifier).

Case 1: msri and sri is not present but the fare type is SPECIAL_RETURN , which means any of SPECIAL RETURN Fare from that Particular airline can be combined and sold.

Example : DEL - BOM (Onward) and BOM - DEL (Return) in Indigo/Spicejet/GoAir flights if totalPriceList any price is SPECIAL_RETURN fare in onward and any fare of SPECIAL_RETURN in Return flights available can be sold and vice versa for Return to Onward.

*(sri and msri must be empty in this case) , when trying to sell any SPECIAL_RETURN Fare both journeys must be of SPECIAL_RETURN fare**.
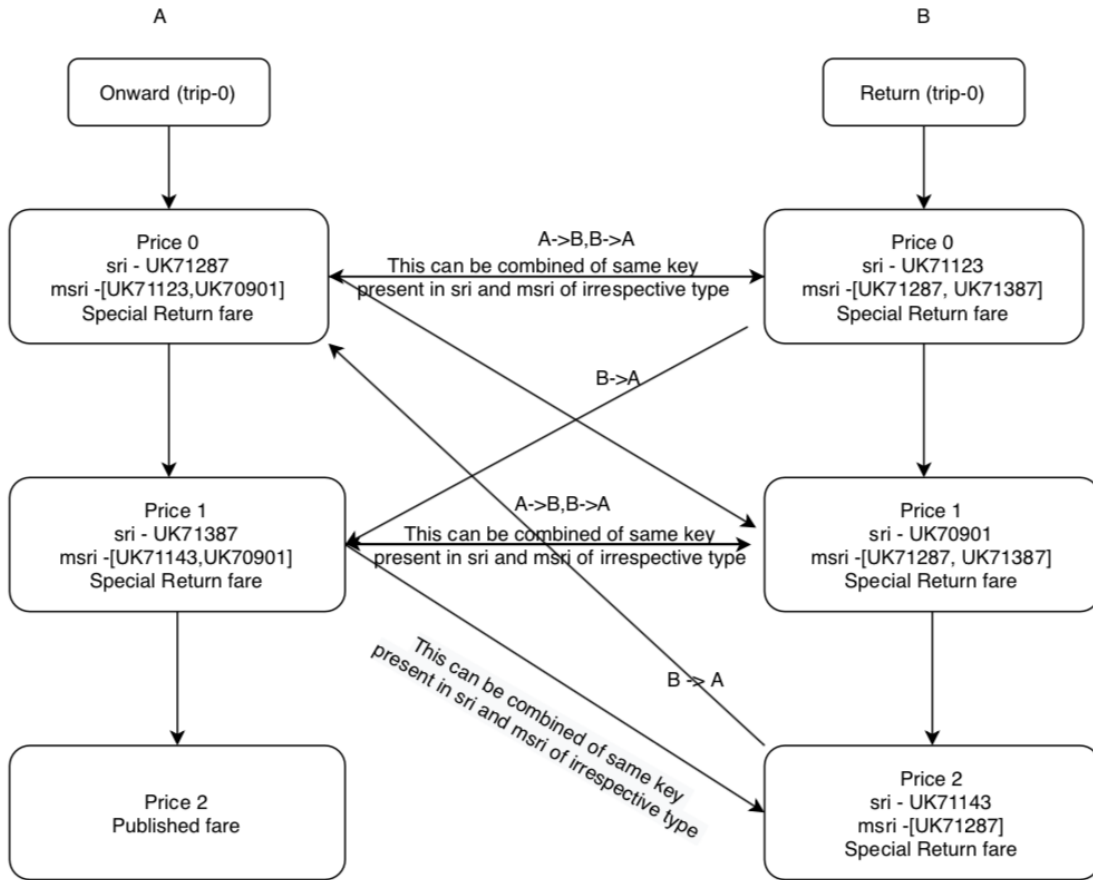
Case 2: And In case there is sri and msri is not empty in **totalPriceList** than any priceInfo Cannot be combined opened as Case 1,sri is Special Return Identifier of Unique Current Price Information and msri will have list of Unique Identifier of sri (which is sri key of SPECIAL_RETURN fare in Return journey)".

```
"sri": "6461UK975",
"msri": [
    "6071UK910"
],
"messages": []
```

Example : DEL - BOM (Onward) and BOM - DEL (Return) in Air Vistara

A

Onward (trip-0)

Price 0
sri - UK71287
msri -[UK71123,UK70901]
Special Return fare

Price 1
sri - UK71387
msri -[UK71143,UK70901]
Special Return fare

Price 2
Published fare

B

Return (trip-0)

Price 0
sri - UK71123
msri -[UK71287, UK71387]
Special Return fare

Price 1
sri - UK70901
msri -[UK71287, UK71387]
Special Return fare

Price 2
sri - UK71143
msri -[UK71287]
Special Return fare

A->B,B->A
This can be combined of same key present in sri and msri of irrespective type

B->A

A->B,B->A
This can be combined of same key present in sri and msri of irrespective type

B -> A

This can be combined of same key present in sri and msri of irrespective type

[A - B] Onward price id (sri) in Onward Flights (A) and key from msri of A must be matched key of sri in return flights (B)

[B-A] Which is vice versa from Return to Onward of sri and msri

Examples for Special Return (Sample logs Link) : Click here to take examples

5.2 Review API (Revalidate/Price RBD)

This method provides selected flight information with revalidated fare available from supplier.The request supports all cases (Oneway,Return,Multi-City). When the request is requested,the system starts validating and the review engine starts validating the selected flight **from the supplier and returns the confirmed fare for particular session time with Tripmaza Booking Reference Created** and returned **Tripmaza booking reference (reviewed price id) valid for next mentioned seconds in AirReviewResponse (conditions::st).**
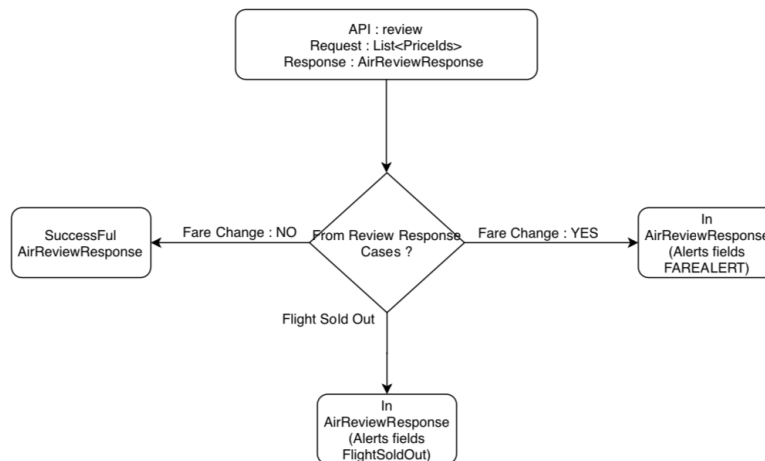
URL: https://st-agent.tripmaza.com/fms/v1/review
Request Method : POST
Sample logs Link : [Click Here to Take Samples](#)

Sample Request View :

* Sample Domestic Return case.

Request Payload:

Key	Parent	Type	Description	Mandatory/ Optional
priceIds	root	ArrayList	List of PriceId for which itinerary has to be reviewed(price validated).	Mandatory

1. It should be in order of route info requested.
2. In Case of **Domestic One way,International One way , International Return and International Multi-City** - only 1 price id will be requested (to validate the price available).
3. In Case of **Domestic Return** - 2 Price id has to be requested (1 from ONWARD and another from RETURN Search Result).
4. In Case of Domestic Multi-City - n (till 6) Price id can be requested (Each will be treated as multiple one ways).

Sample Response View :

```
⊞ {} searchQuery
  ▪ bookingId : "TMS104000003147"
⊟ {} totalPriceInfo
  ⊟ {} totalFareDetail
    ⊞ {} fC
    ⊞ {} afC
⊟ {} status
  ▪ success : true
  ▪ httpStatus : 200
⊟ {} conditions
  [] ffas
  ▪ isa : true
  ⊟ {} dob
    ▪ adobr : false
    ▪ cdobr : false
    ▪ idobr : true
  ▪ isBA : true
  ▪ st : 840
  ▪ sct : "2020-02-07T15:32:57.514"
  ⊟ [] ast
```

*Sample Domestic Return Case

Case :1 : If priceIds keys are expired (if keys sent in review request are wrong or if expired from system).
- will give "Keys Passed in the request is already expired. Please pass valid keys" with httpStatus :400 .

```
□ {} status
    ■ success : false
    ■ httpStatus : 404
□ [ ] errors
    □ {} 0
        ■ errCode : "808"
        ■ message : "Keys Passed in the request is already expired. Please pass valid keys"
```

Case : 2 : **If Booking Class is available with Fare Change from Supplier** then AirReviewResponse will have field "alerts" (type : **FAREALERT**) which indicates old fare and newfare difference. - Example Link

```
□ [ ] tripInfos
□ [ ] alerts
    □ {} 0
        ■ oldFare : 2878
        ■ newFare : 3298
        ■ type : "FAREALERT"
□ {} searchQuery
    ■ bookingId :  "TMS104000003147"
□ {} totalPriceInfo
    ■ priceValidation : true
□ {} status
□ {} conditions
```

Case 3 : If Selected Flight (Price Id is not available from supplier due to booking class not available,other reasons) then AirReviewResponse will have errors (List<errors> which give description and httpStatus : 400).

```
□ {} status
    ■ success : false
    ■ httpStatus : 400
□ [ ] errors
    □ {} 0
        ■ errCode : "1000"
        ■ message : "Request flight is not longer available.Please try different flight"
        ■ details : " You can use booking Id TMS104000003147  for reference"
```

How to use conditions from AirReviewResponse and field Description.

```
conditions
    [ ] ffas
        isa : true
    { } dob
            adobr : false
            cdobr : false
            idobr : true
        isBA : true
        st : 840
        sct : "2020-02-07T15:44:42.816"
    { } gst
            gstappl : true
            igm : false
```
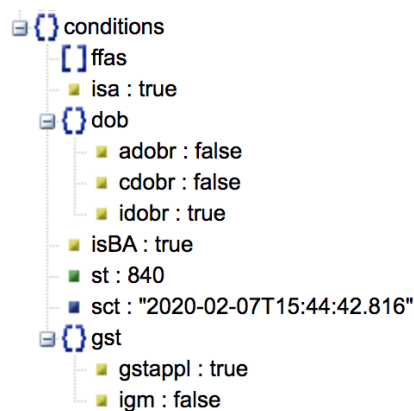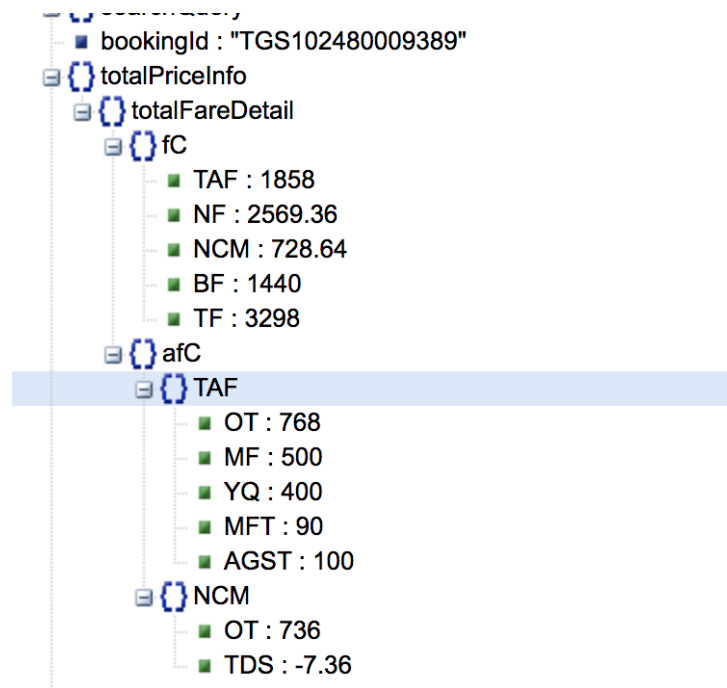
- ➔ **ffas** - Frequent Flier Airlines
- ➔ **isa** - is Seat Applicable on selected flight (if this true ,another api to fetch seat information)
- ➔ **isBA** - is Hold (Block) option applicable for selected flight
- ➔ **st** - Session Time In Seconds (Reviewed Price Id is valid for mentioned seconds to make booking)
- ➔ **sct** - Session Created Time.
- ➔ **pcs -** Passport Booking Conditions
  - ◆ **pped** - Passport Expiry Date is required at the time of booking
  - ◆ **pid** - Passport Issue Date is required at the time of booking
  - ◆ **pm** - Passport is Mandatory for booking
- ➔ **dob** - Date Of Birth Required or Not in PaxType wise
  - ◆ **adobr** - Adult Date of Birth Required
  - ◆ **cdobr** - Child Date of Birth Required
  - ◆ **idobr** - Infant Date of Birth Required
- ➔ **gst** - GST is Applicable and mandatory information
  - ◆ **igm** - GST Mandatory (true) - Valid GST Number and Registered Name and GST Address, GST mobile number has to passed in Booking request.
  - ◆ **gstappl** - GST Applicable (true) - GST Can be passed in this booking but not mandatory.

totalPriceInfo - Selected Journey's Total Fare

```
      bookingId : "TGS102480009389"
   {} totalPriceInfo
      {} totalFareDetail
         {} fC
              TAF : 1858
              NF : 2569.36
              NCM : 728.64
              BF : 1440
              TF : 3298
         {} afC
            {} TAF
                 OT : 768
                 MF : 500
                 YQ : 400
                 MFT : 90
                 AGST : 100
            {} NCM
                 OT : 736
                 TDS : -7.36
```

totalFareDetail - Total Journey Summarized Price of Total Travellers. - (Example : 2Adult, 1Child Search then have to multiply by 2 for Adult Fare and multiply by 1 for Child Fare)

➔ fc - Fare Components (which has airline , commission related fare information)
  ◆ TAF - Taxes and Fees
  ◆ NF - Net Fare
  ◆ NCM - Net Commission (Net Commission is Gross Commission - TDS)
  ◆ BF - Base Fare
  ◆ TF - Total Fare (This fare will be charged from agent , net commission will be returned after that)
➔ afc - Additional Fare Components (which has break of Above fc fare components)
  ◆ TAF - Break Up of Taxes and Fees from fC
    ● OT - Other Charges
    ● MF - Management Fee
    ● MFT - Management Fee Tax (It's nothing but GST of Management Fee)
    ● YQ - Fuel Surcharge
    ● AGST - Airline GST Component
  ◆ NCM - Break Up of Net Commission
    ● OT - Gross Commission
    ● TDS - TDS

TripInfo Brief : - Each TripInfo which has sI (updated segments information and totalPriceList of trip).

1. sI (Segment Information) - List<Segments> : which has control of the fields.

➔ Id - Segment ID - which is used for SSR if any to be selected.
➔ **fD** (Flight Designator) - which has Marketing Airline Information , Airline Equipment type, flight number).
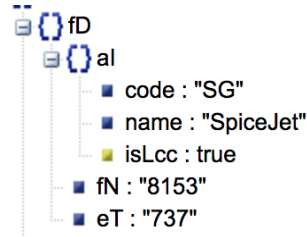
- ● al - airline info which has code (IATA Airline Code, followed with Airline name).
- ● fN - Marketing Airline Flight Number
- ● eT - Marketing Airline Equipment Type

➔ **stops** - Number of Stops in this segment.
➔ **duration -** Segment Journey duration
➔ **land -** is this flight arriving the next day.
➔ **isRs -** isReturnSegment will be true and this field is present only when the segment is return flight.
➔ **sN -** segment number
➔ **dt -** Departure Time of this segment (YYYY-MM-SS HH:MM Format)
➔ **at -** Arrival time of this segment (YYYY-MM-SS HH:MM Format)
➔ **aa -** Arrival Airport Information (which has IATA Airport code,name,country,**Terminal**)
➔ **da -** Departure Airport Information (which has IATA Airport code,name,country,**Terminal**)
➔ **oB -** Operating Airline which has code (IATA Airline Code, followed with Airline name).
➔ **so -** Stop Over airports (which has IATA Airport code,name,country)
➔ **cT -** Connecting Time (LayOver time)
➔ **ssrInfo - Baggage,Meal,Extra Services ..** (explained below)

2. totalPriceList : Journey Summarised fare

➔ **fd -** fareDetails of Requested Passenger Types - (Which is explained in page - 21)
➔ **fareIdentifier -** what kind of fare **PUBLISHED,CORPORATE,SPECIAL_RETURN.**
➔ **messages - if any supplier or airline remarks**
➔ **id - unique id of price**
➔ **Pc - Plating Carrier of Journey**

Fare Detail : - PaxType Wise Fare Break Up , Booking class and fare related information.

- ➔ **ADULT/CHILD/INFANT -** will be key
- ➔ **fC -** fare Components of Fare to PaxType - (Which is explained in page -)
- ➔ **sR** - Seats Remaining
- ➔ **bl** - Baggage information (Hand Baggage & Checking baggage Related)
    - ◆ iB - Checking Baggage (No format specific example :20Kg , 20 Kg, 1 Piece , 1 Piece)
    - ◆ cB - Cabin Baggage
- ➔ **rT** - Refundable Type (Integer type as below follows)
    - ◆ 0 - NON - Refundable
    - ◆ 1 - Refundable
    - ◆ 2 - Partial Refundable
- ➔ **cc** - Cabin Class Fare
    - ◆ ECONOMY
    - ◆ PREMIUM_ECONOMY
    - ◆ BUSINESS
    - ◆ FIRST
- ➔ **cB -** Class Of Booking
- ➔ **fB** - Fare Basis
- ➔ **mi -** Meal Indicator (Boolean Type)
    - ◆ **True -** Free Meal
    - ◆ **False -** Paid Meal

SSR Informations :

Applicable SSR's are under segment information : BAGGAGE, MEAL , EXTRASERVICES.
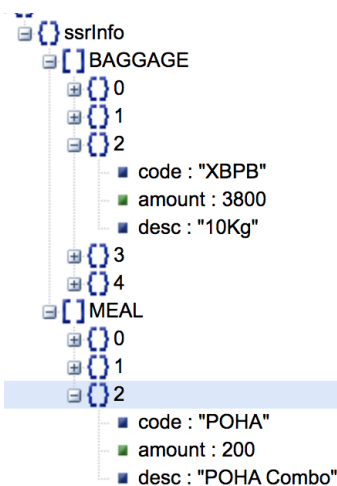
Each SSR Information has detailed information that needs to be shown and passed to the next request.

- ➔ code : which is a unique code of that SSR Received from supplier and that's needed to be sent in booking request if customer selected any.
- ➔ amount : Amount of Corresponding SSR.
- ➔ desc - Description of Corresponding SSR that needs to be shown to Customer for clear information.

- In Case connecting Segments , Baggage (SSRType , amount field will be not present) - which means - whatever selected on the first segment of journey applicable to that corresponding journey. (Baggage always works as journey wise, when the amount is present in segment wise then segment wise Baggage applicable ,it can be varied).
- In Case of Meal (SSR type) - Each segment will have a unique Meal SSR which is applicable only on that segment.
- In Case of Meal (SSR type, if the amount field is empty , then any meal can be selected at free of cost).

  If SSR is being selected for any segment then corresponding segment (segment "id" as key has to be passed in booking request field key) - which is explained in below booking flows.

Air Review Response:

Key	Parent	Type	Description	Mandatory/ Optional
tripInfos	root	Array of tripInfo	Array of tripInfo object. Each Object contains details of each trip.	Mandatory
searchQuery	root	Object	Contains details of Search made.	Mandatory
bookingId	root	String	Booking Id generated after setting up session.	Mandatory
totalPriceInfo	root	Object	Contains the total fare details of the booking	Mandatory
conditions	root	Object	Details like passport , DOB, GST.	Mandatory
pcs	conditions	Object	Only in case of International search. Passport components	Optional
pped	pcs	Boolean	Passport Expiry Date	Optional

pid	pcs	Boolean	Passport Issue Date	Optional
pm	pcs	Boolean	Passport Mandatory	Optional
ffas	conditions	ArrayList	Frequent Flier	Optional
dob	conditions	Object	Whether date of birth of ADULT, CHILD and INFANT is required or not during booking	Optional
adobr	dob	Boolean	Is Adult date of Birth is required in Passengers details	Optional
cdobr	dob	Boolean	Is Child date of Birth is required in Passengers details	Optional
idobr	dob	Boolean	Is Infant date of Birth is required in Passengers details	Optional
isBA	conditions	Boolean	Blocking Allowed (Booking can be put on HOLD)	Optional
st	conditions	Integer	Review Session Valid Time in seconds	Optional
sct	conditions	String	Session Created Time	Optional
gst	conditions	Object	Whether GST is required or not during booking	Optional
gstappl	gst	Boolean	GST applicable for current booking.	Optional
igm	gst	Boolean	GST Mandatory	Optional
status	root	Object	Status of the API call	Mandatory

5.3 Fare Rule API

This method is used to get the fare rules, penalties and condition of a specific flight option. It contains the fare rule info and the relevant details associated with it, such as restrictions and Penalty charges,cancellation charges,no show charges and seat information.

This API can be used for Before ticketing (using priceid) to fetch the required penalty information.**(During Search & Review Only).**

URL: https://st-agent.tripmaza.com/fms/v1/farerule
Request Method: POST

Sample logs Link : [Click here to take examples](#)

1. **Mini Rule Structure**
2. **Cat 16 Category (Unstructured Paragraph Format)**

Request Sample View :

```
JSON
   id : "16-0764987189_BOMDELI5330_282038212531214"
   flowType : "SEARCH"
```

* Sample Search fare Rule
* flowType : SEARCH, REVIEW

*System can respond to any one type from the supplier (Mini Rule or Cat 16 Category).
1. If Cat 16 Category under miscInfo (rest of fields will be empty/not present)
2. If Mini Rule MiscInfo will not be present.

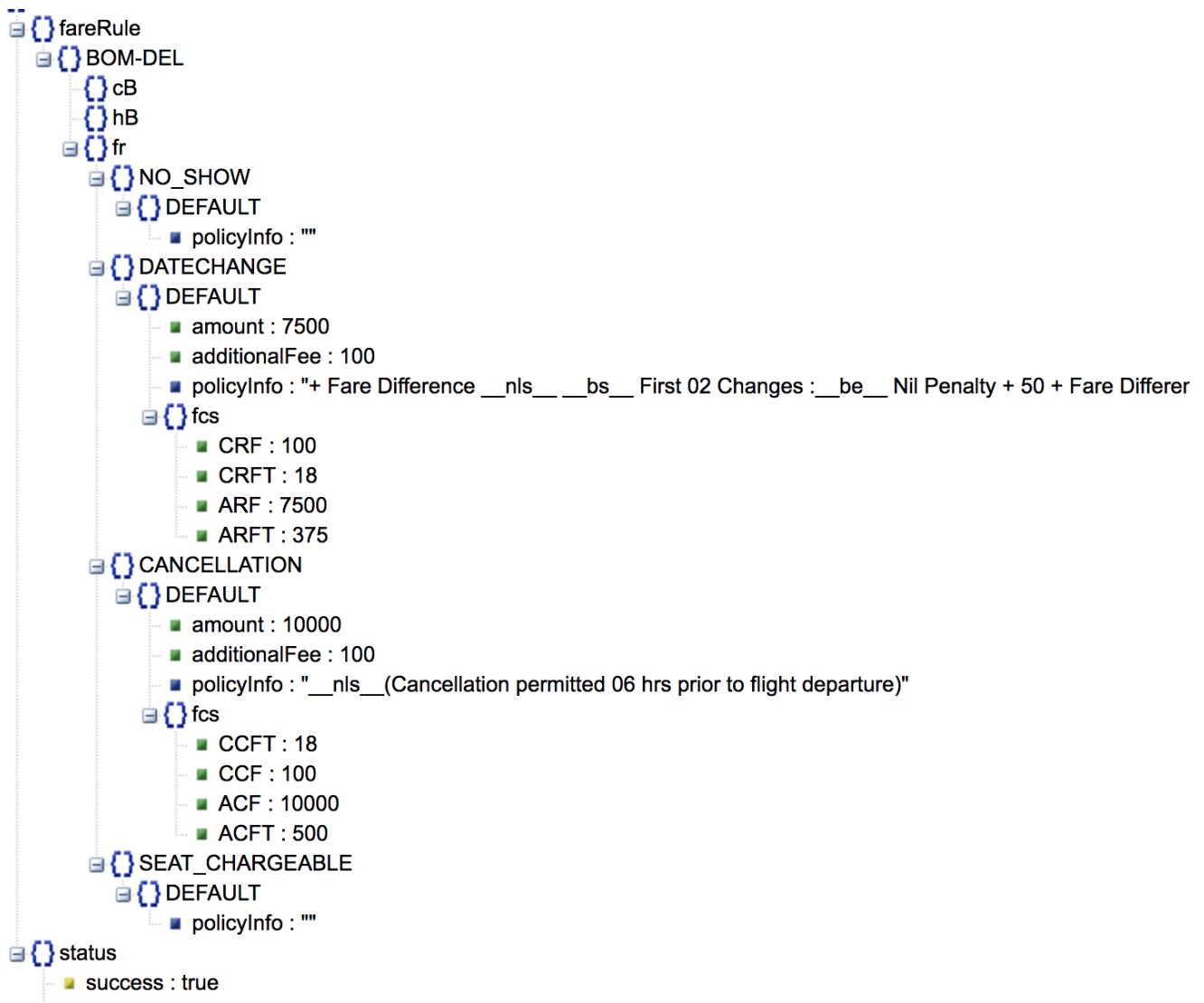**Case :1 :** If priceIds keys are expired (if keys sent in review request are wrong or if expired from system).
- will give "Keys Passed in the request has already expired. Please pass valid keys" with httpStatus :400
.

```
status
   success : false
   httpStatus : 404
errors
   0
      errCode : "808"
      message : "Keys Passed in the request is already expired. Please pass valid keys"
```

Sample Fare Rule Response (Mini Rule):

```
fareRule
  BOM-DEL
    cB
    hB
    fr
      NO_SHOW
        DEFAULT
          policyInfo : ""
      DATECHANGE
        DEFAULT
          amount : 7500
          additionalFee : 100
          policyInfo : "+ Fare Difference __nls__ __bs__ First 02 Changes :__be__ Nil Penalty + 50 + Fare Differer
          fcs
            CRF : 100
            CRFT : 18
            ARF : 7500
            ARFT : 375
      CANCELLATION
        DEFAULT
          amount : 10000
          additionalFee : 100
          policyInfo : "__nls__(Cancellation permitted 06 hrs prior to flight departure)"
          fcs
            CCFT : 18
            CCF : 100
            ACF : 10000
            ACFT : 500
      SEAT_CHARGEABLE
        DEFAULT
          policyInfo : ""
  status
    success : true
```
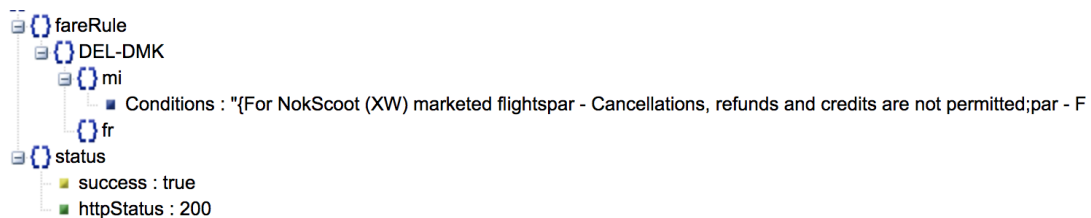
fr - Fare rule which has CANCELLATION, DATECHANGE, NO_SHOW, SEAT_CHARGEABLE
information as a key.

Each Policy Can have Different Types of penalty information such Before Departure,After Departure,Default.

➔ amount - Corresponding Fare Rule Type Airline Fee (Airline Cancellation,Reschedule Fee)
➔ additionalFee - Tripmaza Fee
➔ policyInfo - Some Additional Text which is shown for customers.
➔ **fcs** - fare components related to fare rule
  ◆ ARF - Airline Reschedule Fee
  ◆ ARFT - Airline Reschedule Fee Tax
  ◆ CRF - Tripmaza Reschedule Fee
  ◆ CRFT - Tripmaza Reschedule Fee Tax
  ◆ ACF - Airline Cancellation Fee
  ◆ ACFT - Airline Cancellation Fee Tax
  ◆ CCF - Tripmaza Cancellation Fee
  ◆ CCFT - Tripmaza Cancellation Fee Tax

Case : 2: In case of no mini rule for supplier then Cat 16 Category Type fare rule returned in below structure under the "mi" tag as plain string format.

```
⊟ { } fareRule
   ⊟ { } DEL-DMK
      ⊟ { } mi
          ■ Conditions : "{For NokScoot (XW) marketed flightspar - Cancellations, refunds and credits are not permitted;par - F
        { } fr
⊟ { } status
      ■ success : true
      ■ httpStatus : 200
```

Request Payload:

Key	Parent	Type	Description	Mandatory/ Optional
flowType	root	String	Flow Type. Whether SEARCH or REVIEW	Mandatory
id	root	String	Price Id whose fare rule need to be fetched	Mandatory

Response:

Key	Parent	Type	Description	Mandatory/ Optional
farerule	root	Object	Trip level Id	Mandatory
status	root	Object	Status of the API call	Mandatory
[departureCode-arrivalCode]	farerule	Object	Departure code and Arrival code. Eg: DEL-MAA	Mandatory
isMealIndicator	[departureCode-arrivalCode]	Boolean	Indicates selected flight is included free Meal	Optional
isHandBaggageIndicator	[departureCode-arrivalCode]	Boolean	Indicates selected flight is hand baggage fare	Optional
refundableType	[departureCode-arrivalCode]	String	Refundable type	Optional
checkedInBaggage	[departureCode-arrivalCode]	Object	Check in baggage information pax wise	Optional
handBaggage	[departureCode-arrivalCode]	Object	Hand Baggage information pax wise	Optional
fareRuleInfo	[departureCode-arrivalCode]	Object	Fare rule Information about Cancellation,DateChange..	Optional
NO_SHOW	fareRuleInfo	Object	No show information	Optional
DATECHANGE	fareRuleInfo	Object	Date Change Information	Optional
CANCELLATION	fareRuleInfo	Object	Cancellation information	Optional
SEAT_CHARGEABLE	fareRuleInfo	Object	Seat chargeable information	Optional
miscInfo	[departureCode-arrivalCode]	ArrayList	Plain Text Fare Rule info	Optional

5.4 Seat MAP API

This method is used to get the Seat Information about the reviewed (revalidated flight).After Review is Successful, Seat Map can be requested by calling this api by passing "**bookingid**" from AirReviewResponse and isa (from conditions in AirReviewResponse should be also true) to fetch enough seat information otherwise no seat information will be provided in SeatMapResponse.

*Response will contain seat configuration of all the aircrafts that are present in selected flights.

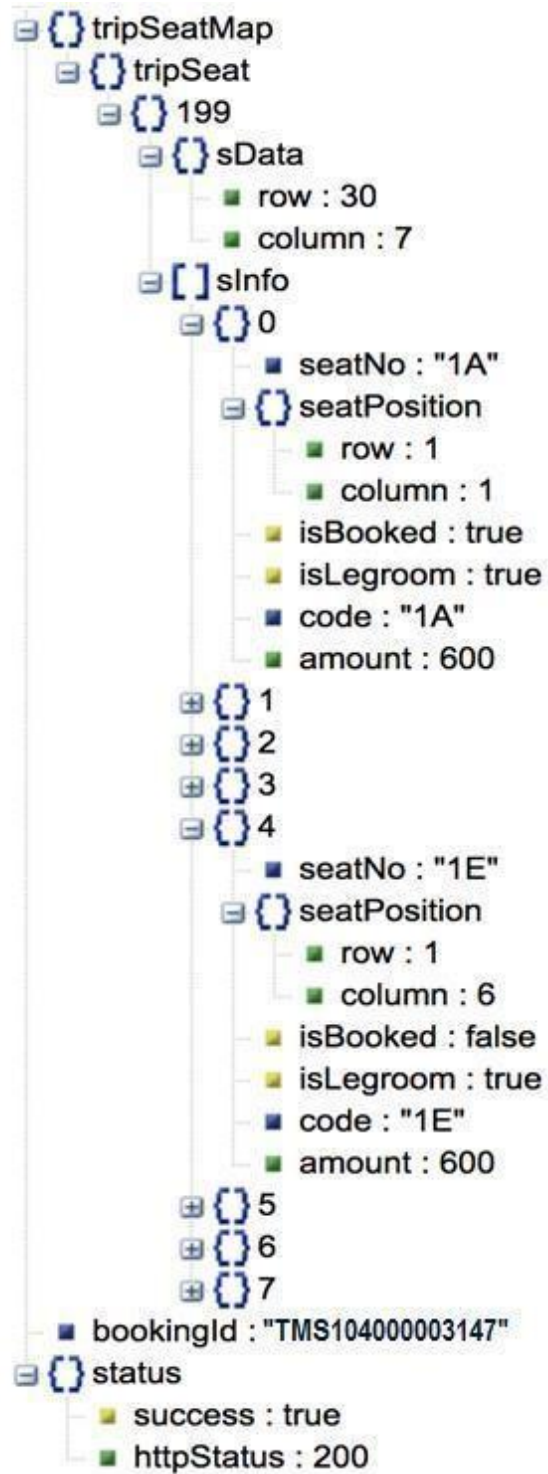**URL:** https://st-agent.tripmaza.com/fms/v1/seat
Request Method : POST
Sample logs Link : Click Here to Examples

Sample Request View :

{} searchQuery
■ bookingId : "TMS104000003147"

***BookingId from AirReviewResponse has to be passed to fetch seat map

Sample Response View :

```
⊟ {} tripSeatMap
   ⊟ {} tripSeat
      ⊟ {} 199
         ⊟ {} sData
               ▪ row : 30
               ▪ column : 7
         ⊟ [ ] sInfo
            ⊟ {} 0
                  ▪ seatNo : "1A"
               ⊟ {} seatPosition
                     ▪ row : 1
                     ▪ column : 1
                  ▪ isBooked : true
                  ▪ isLegroom : true
                  ▪ code : "1A"
                  ▪ amount : 600
            ⊞ {} 1
            ⊞ {} 2
            ⊞ {} 3
            ⊟ {} 4
                  ▪ seatNo : "1E"
               ⊟ {} seatPosition
                     ▪ row : 1
                     ▪ column : 6
                  ▪ isBooked : false
                  ▪ isLegroom : true
                  ▪ code : "1E"
                  ▪ amount : 600
            ⊞ {} 5
            ⊞ {} 6
            ⊞ {} 7
      ▪ bookingId : "TMS104000003147"
   ⊟ {} status
         ▪ success : true
         ▪ httpStatus : 200
```

TripSeatMap : This will contain a selected journey applicable seat map based on AirCraft type.

- ➔ tripSeat (key)
  - ◆ **key -** (segment :id from tripInfos in review response) - belongs to that particular segment
  - ◆ **sData** - Seat Deck Data (How many columns and rows to be created)
  - ◆ **sInfo** - List of Seats in Selected Segment
    - ● **seatNo** : Seat Information Number with code
    - ● **seatPosition** : will have row and column number to place seat
      - ○ row
      - ○ column
    - ● **isBooked** - is already booked for a seat or not.
    - ● **isLegRoom** : Leg Space (attribute will present only if available)
    - ● **code** : Supplier Booking Code which needs to be passed for booking if a customer selects this particular seat.
    - ● **amount** : Seat Chargeable amount.

Request Payload:

Key	Parent	Type	Description	Mandatory/ Optional
bookingId	root	String		Mandatory

Response:

Key	Parent	Type	Description	Mandatory/ Optional
bookingId	root	String		Mandatory
tripSeatMap	root	Object	It contains a Map of AirCraft and its Seat configuration for the whole trip.	Mandatory
tripSeat	tripSeatMap			
sData		Object		

row	sData	Integer		
column	sData	Integer		
sInfo		Array of Object	Each Object contains single seat Information.	
seatNo	sInfo	String	Seat No	
isBooked	sInfo	boolean	Seat status	
isLegroom,	sInfo	Boolean	Seat with Legroom	
isAisle	sInfo	boolean	Aisle seat	
code	sInfo	String	Seat Code	
amount	sInfo	Double	Seat Price	
seatPosition	sInfo	Object		
row	seatPosition	Integer	Deck Data Row	
column	seatPosition	Integer	Deck Data column	

5.5 BOOKING API

The following section describes the availability methods exposed via the API to Book.

1. INSTANT BOOK
2. HOLD
3. CONFIRM FARE BEFORE TICKET
4. CONFIRM-BOOK (TICKETING AFTER HOLD)

Book method is to **hold the Booking or Direct Ticketing**. After Submitting the passenger & contact details for the selected itinerary ,BookingResponse will be given with bookingId and status of submitted request.

1. While submitting BookingRequest it will be validated then taking forward to booking, in case of any validation failure, Error Message will be given in response.
2. After validation,Booking will be created in the Tripmaza System by deducting Payment ,and Supplier Side will be done. (BookingDetails Api has to be used to GET PNR and TicketNumber if any and Order Status must be Success).
3. In case of any fare jump , PNR and Ticket Number will be not generated, and can directly coordinate with Tripmaza Customer Support.

1: INSTANT BOOK (With Payment Info)

This method is used to book the reviewed/selected flight confirming instantly with Airline and Tripmaza by passing Payment Information,Travellers and Contact Details , If Any GST also can be passed.
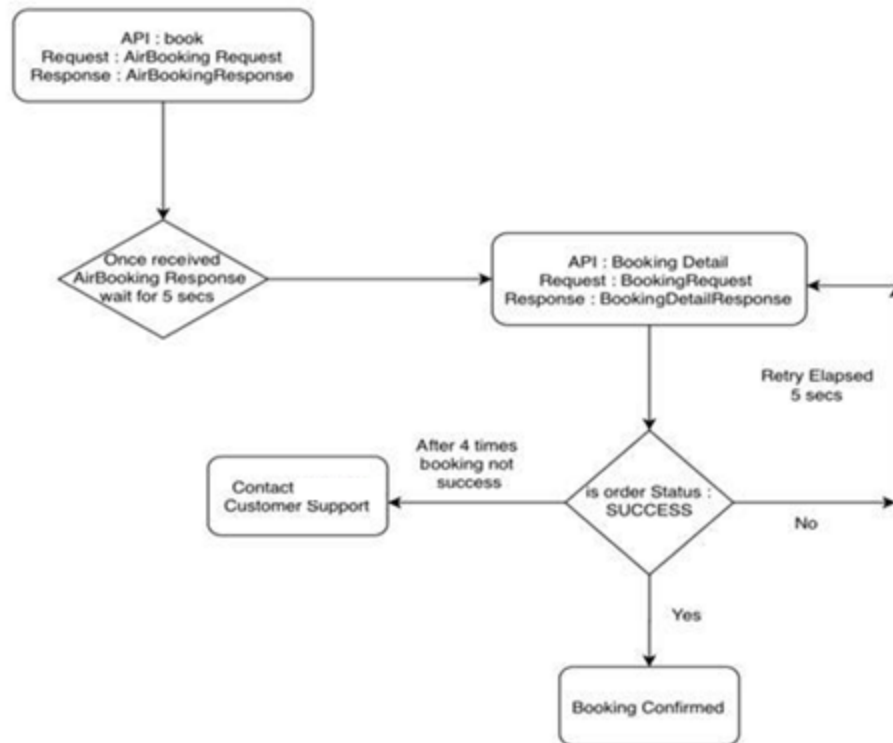
URL: https://st-agent.tripmaza.com/oms/v1/air/book
Request Method : POST

* amount sent in Book Request should be Gross Fare (TF from AirReviewResponse).
Sample logs Link : Click to here to take examples

Cases to verify before integration :

- **With GST & Passport**
- **With SSR**

Notes on Booking Request :

Case :1 : SSR - List of Meal, Baggage - In Request code which is selected by customer Vs on segment key (AirReviewResponse -> "tripInfos" under segment will be having unique key called id which is key)

Case :2 : In Case of GST (all fields of GST has to be passed) with Valid and as IATA Standard have to be used.

Case 3 : In case of SME fares , GST will be mandatory , in that case have to verify AirReviewResponse Conditions (igm and gstappl need to be true) & have to pass GST Info in booking.

Sample Request View (Without SSR) :

```
JSON
    bookingId : "TMS104000003147"
[ ] paymentInfos
    { } 0
        amount : 7403
[ ] travellerInfo
    { } 0
        ti : "Mr"
        fN : "Test"
        IN : "AdultA"
        pt : "ADULT"
    { } 1
        ti : "Mr"
        fN : "Test"
        IN : "ChildA"
        pt : "CHILD"
    { } 2
        ti : "Mr"
        fN : "Test"
        IN : "InfantA"
        pt : "INFANT"
        dob : "2019-08-09"
{ } gstInfo
    gstNumber : "04AABCA7122A1ZF"
    email : "xxx@yyyy.com"
    registeredName : "XYZ Pvt Ltd"
    mobile : "+919231100324"
    address : "gurugram"
{ } deliveryInfo
    [ ] emails
        0 : "xxx@yyyy.com"
    [ ] contacts
        0 : "+919231100324"
```

Description :

- **bookingId :** Booking Id from AirReviewResponse
- **paymentInfos** - Payment for the booking to Tripmaza (List<PaymentInofs>)
    - amount - total payable amount to Tripmaza for the booking (TF from AirReviewResponse)
- **deliveryInfo** - Delivery Details Related to Booking (which need to be passed to supplier)
    - emails - Booking Email Id (It should be valid email id)
    - contacts - Valid Contact Number (Followed with country code and valid mobile number) - Example (+919500112233)
- **travellerInfo -** List of Traveller which flight to be booked
    - **ti -** Title of the Traveller
        - **Adult:** Mr,Mrs,Ms
        - **Child:** Ms, Master
        - **Infant :** Ms, Master
    - **pt** - PaxType - ADULT,CHILD, INFANT (Values should be Caps)
    - **fN -** First Name of the Traveller
    - **IN** - Last Name of the Traveller (Key : lowercase "L"N)
    - **dob** - Traveller Date Of Birth (YYYY-MM-DD Format)
    - **pNum** - Passport Number
    - **eD** - Expiry Date of the passport (YYYY-MM-DD Format)
    - **pNat** - Passport Nationality (Passport Issued Country) - IATA Country Code(2-Letter)
    - **pid** - Passport Issue Date (YYYY-MM-DD Format)
    - **ssrBaggageInfos** (List of baggage on segment selected)
        - **key** : Segment Key from AirReviewResponse
        - **code** : Customer Selected SSR Code on Corresponding Segment
    - **ssrMealInfos**
        - **key** : Segment Key from AirReviewResponse
        - **code** : Customer Selected SSR Code on Corresponding Segment
    - **ssrSeatInfos**
        - **key** : Segment Key from AirReviewResponse
        - **code** : Customer Selected SSR Code on Corresponding Segment
    - **ssrExtraServiceInfos**
        - **key** : Segment Key from AirReviewResponse
        - **code** : Customer Selected SSR Code on Corresponding Segment
- **gstInfo -** GST Which need to be passed for billing and supplier - **all fields under gst are mandatory.**
    - gstNumber - Valid 15 digit GST Number (mandatory)
    - registeredName - GST Registered Name (maximum 35 characters as IATA standard) (mandatory)
    - Email - GST Email
    - Mobile - GST Mobile
    - Address - GST Registered Address (maximum 70 characters as IATA standard)

Request Payload:

Key	Parent	Type	Description	Mandatory/ Optional
bookingId	root	String	Booking id which is generated at Review.	Mandatory
paymentInfos	root	Array of Objects	Object containing the payment details of Booking. Do not send paymentInfos in case of Hold booking.	Mandatory
amount	paymentInfos	number	Total cost for the Booking	Mandatory
travellerInfo	root	Array of Objects	List of traveller details object	Mandatory
gstInfo	root	Object	Object containing GST info	Mandatory
gstNumber	gstInfo	String	GST number	Mandatory
registeredName	gstInfo	String	GST name	Mandatory
deliveryInfo	root	Object	Object containing the details of delivery emails and mobile numbers	Mandatory

Response:

Key	Parent	Type	Description	Mandatory/ Optional
bookingId	root	String	Booking id which is generated at Review.	Mandatory
status	root	Object	Status of the APi call	Mandatory

2: HOLD (WithOut PaymentInfo)

Book method is to **hold the Booking** . After Submitting passenger & contact details for the selected itinerary, BookingResponse will be given with bookingId and status of submitted request.

    1.While submitting BookingRequest will be validated then taking forward to booking, in case of any validation failure, Error Message will be given in response.

    2.After validation , Booking will be done and Supplier Booking is Success in BookingDetails API Call will get PNR and TicketNumber if any.

    3.In case of any fare jump, PNR and Ticket Number will be not generated. And can directly coordinate with Customer Support.

Which is the same as **INSTANT BOOK** (**request should not have paymentInfo**).

Flow For Hold & Hold Confirm Ticket :
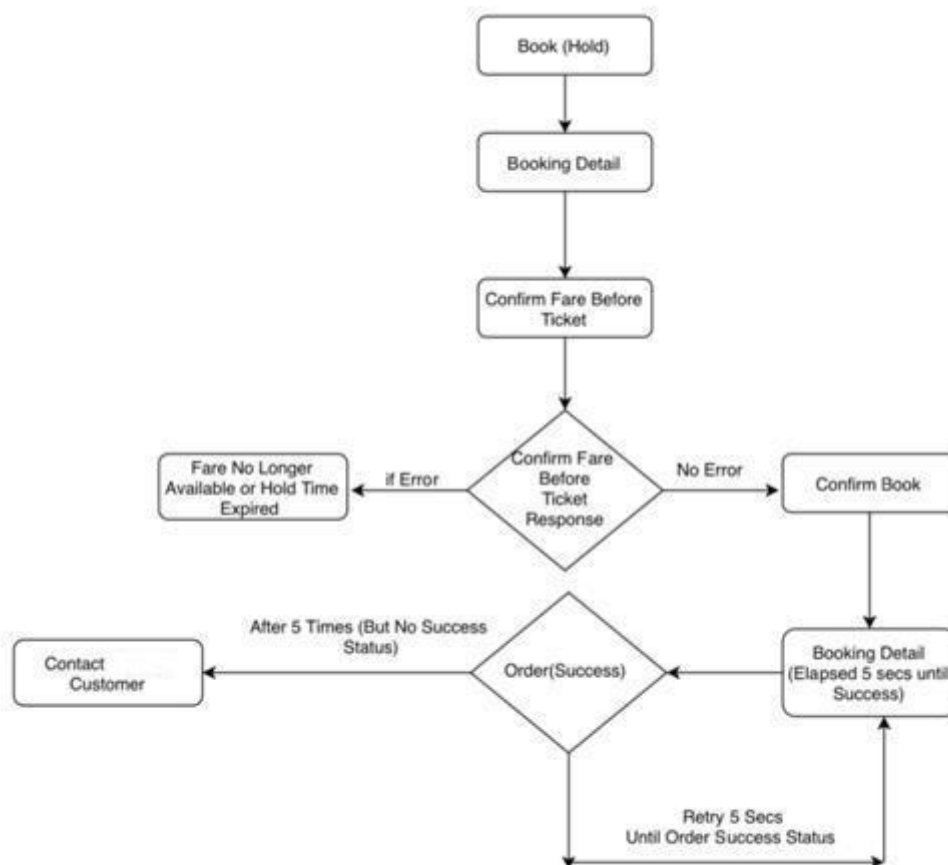
Sample logs Link : Click to here to take examples

Sample Request View :

- bookingId : "TJ1XXXXXX123"
- [ ] travellerInfo
  - { } 0
    - pt : "ADULT"
    - ti : "Mr"
    - fN : "XXXX"
    - lN : "XXXXX"
    - pNum : "PXXXXXX5"
    - pid : "2018-04-07"
    - eD : "2026-04-06"
    - pNat : "IN"
    - dob : "1998-03-09"
- { } gstInfo
  - gstNumber : "XXXXXXXXXXXXX"
  - registeredName : "XXXXXXXXXXX"
  - mobile : "9511223344"
  - email : "test@test.com"
  - address : "Valid 70 digit characters IATA Standard"
- { } deliveryInfo
  - [ ] emails
    - 0 : "xxxxxxx@xxx.com"
  - [ ] contacts
    - 0 : "XXXXXXXXXX"

3.CONFIRM-FARE BEFORE TICKETING (BEFORE CONFIRM TICKETING)

This method is used to Validate the fare for PNR is available or not , if fare is not available it will give "Fare No Longer available or if time limit of hold booking if expired".

URL: https://st-agent.tripmaza.com/oms/v1/air/fare-validate
Method: POST
Sample Logs Link : Click to here to take examples

Sample Request View :
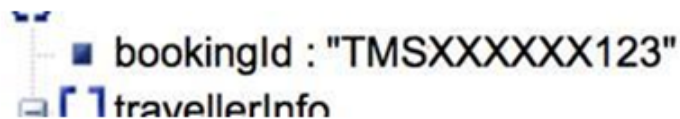
Sample Response View :

Case : 1 : If Successful , Fare & Booking class still valid to confirm on Supplier before ticketing.

Case : 2: If **fare or booking class is not available on the Supplier side** for Hold PNR.

Case : 3 : If the **Hold time limit is expired** for the Hold PNR.
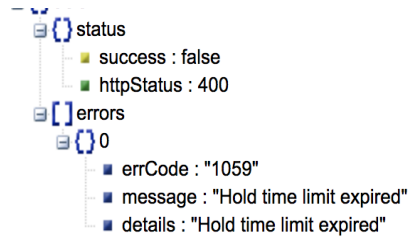
```
status
   success : false
   httpStatus : 400
errors
   0
      errCode : "1059"
      message : "Hold time limit expired"
      details : "Hold time limit expired"
```
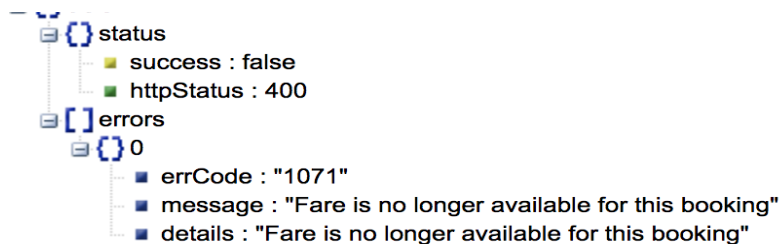
Request Payload:

Key	Parent	Type	Description	Mandatory/ Optional
bookingId	root	String	Id of the booking which needs to be released	Mandatory

Response:

Key	Parent	Type	Description	Mandatory/ Optional
status	root	Object	Status of the API call	Mandatory

4.CONFIRM-BOOK (HOLD BOOKING TO CONFIRM)

This method is used to confirm the hold(blocked) ticket by making payment to the booking and confirming the ticket (after the fare confirmation before ticketing api).

URL: https://st-agent.tripmaza.com/oms/v1/air/confirm-book
Method: POST
Sample Logs Link : [Click to here to take example](#)

Sample Request View :

Request Payload:

Key	Parent	Type	Description	Mandatory/ Optional
bookingId	root	String	Booking id which is generated at Review.	Mandatory
paymentInfos	root	Array of Objects	Object containing the payment details of Booking.	Mandatory
amount	paymentInfos	number	Total cost for the Booking	Mandatory

Response:

Key	Parent	Type	Description	Mandatory/ Optional
status	root	Object	Status of the API call	Mandatory
bookingId	root	String	Booking id which is already generated.	Mandatory

6. RETRIEVE BOOKING: (BookingDetail)

This method is used to retrieve booking details and booking status of an existing booking using BookingId. It is typically used after a booking has been made to capture a current snapshot of the reservation to get sync of PNR,ticket number.

URL: https://st-agent.tripmaza.com/oms/v1/booking-details
Method: POST
Sample Logs Link : Click Here to Take Examples

Sample Request View :

bookingId : "TMS104000003147"

Sample Response View :
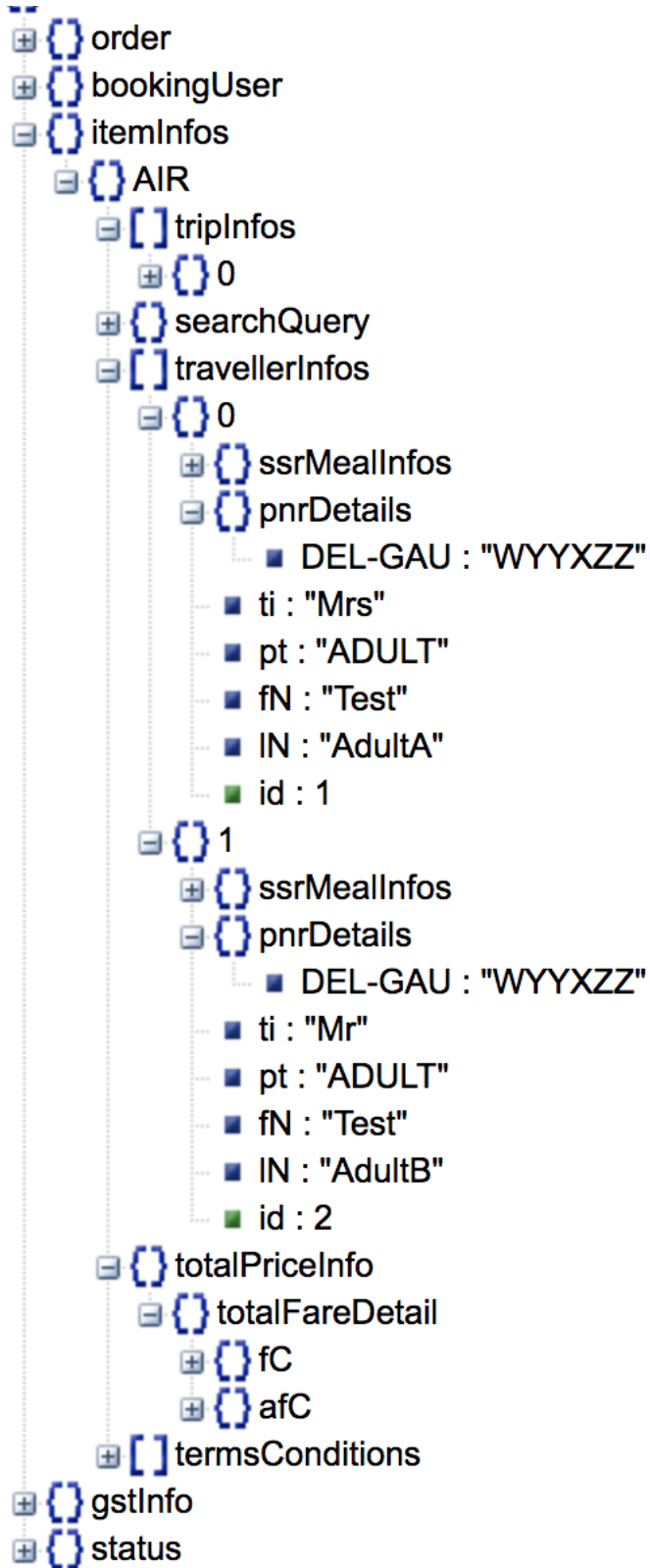
```
⊞ {} order
⊞ {} bookingUser
⊟ {} itemInfos
   ⊟ {} AIR
      ⊟ [ ] tripInfos
         ⊞ {} 0
      ⊞ {} searchQuery
      ⊟ [ ] travellerInfos
         ⊟ {} 0
            ⊞ {} ssrMealInfos
            ⊟ {} pnrDetails
               ▪ DEL-GAU : "WYYXZZ"
            ▪ ti : "Mrs"
            ▪ pt : "ADULT"
            ▪ fN : "Test"
            ▪ lN : "AdultA"
            ▪ id : 1
         ⊟ {} 1
            ⊞ {} ssrMealInfos
            ⊟ {} pnrDetails
               ▪ DEL-GAU : "WYYXZZ"
            ▪ ti : "Mr"
            ▪ pt : "ADULT"
            ▪ fN : "Test"
            ▪ lN : "AdultB"
            ▪ id : 2
      ⊟ {} totalPriceInfo
         ⊟ {} totalFareDetail
            ⊞ {} fC
            ⊞ {} afC
      ⊞ [ ] termsConditions
⊞ {} gstInfo
⊞ {} status
```

Request Payload:

Key	Parent	Type	Description	Mandatory/ Optional
bookingId	root	String	Id of the booking which needs to be retrieved	Mandatory

Response:

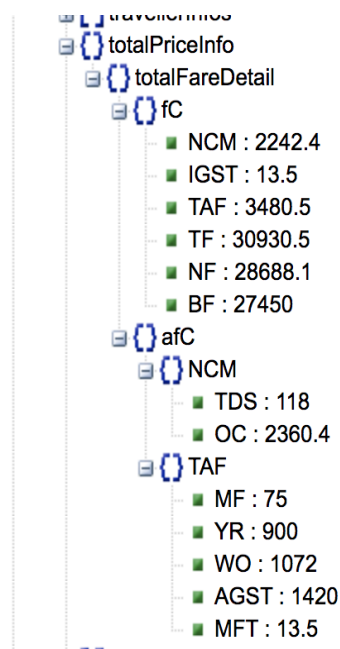| Key	Parent	Type	Description	Mandatory/ Optional
order	root	Object	This object contains the order details	Mandatory
bookingUser	root	Object	This contains user details who has done the booking	Mandatory
itemInfos	root	Object	This includes the item info like price information, passenger details and search query	Mandatory
AIR	itemInfos	Object	Includes all the details of the AIR booking	Mandatory
tripInfos	AIR	Array of tripInfo	Array of tripInfo object	Mandatory
travellerInfos	AIR	Object	Array of travellerInfo	Mandatory
gstInfo	root	Object	This contains the GST data that is given when booking	Mandatory
status	root	Object	Status of the API call.	Mandatory

Description of Booking Details Response :

➜ **Order -** Order which Created in Tripmaza System
   ◆ **bookingId** - Unique Reference Number for Booking in Tripmaza System
   ◆ **amount** - Order Amount which charged in Tripmaza System
   ◆ **deliveryInfo** - Booking EmailId and Contact Info which passed in Booking Request
   ◆ **status - Order Status of Booking Id (Its Success or ON_HOLD, PENDING)**
➜ **itemInfos**
   ◆ **tripInfos -** which is already explained same in Review & Search - which contains Journey Related Information
   ◆ **totalPriceInfo** - which has Fare Component & Additional Fare Components - Fare Related Information to the Booking.

```
   [ ] travellerInfos
 ⊟ { } totalPriceInfo
   ⊟ { } totalFareDetail
     ⊟ { } fC
          ▪ NCM : 2242.4
          ▪ IGST : 13.5
          ▪ TAF : 3480.5
          ▪ TF : 30930.5
          ▪ NF : 28688.1
          ▪ BF : 27450
     ⊟ { } afC
       ⊟ { } NCM
            ▪ TDS : 118
            ▪ OC : 2360.4
       ⊟ { } TAF
            ▪ MF : 75
            ▪ YR : 900
            ▪ WO : 1072
            ▪ AGST : 1420
            ▪ MFT : 13.5
```

➜ fc - Fare Components
   ◆ NF - Net Fare
   ◆ IGST - Integrated Goods and Services Tax.
   ◆ TAF - Taxes and Fees
   ◆ SSRP - Special Service Price (SSR - Baggage,Meal, Seat..)
   ◆ TF - Total Fare
   ◆ BF - Base Fare
   ◆ NCM - Net Commission
➜ afc - Additional Fare Components - Break Up of "fc" if applicable
   ◆ TAF - Taxes and Fees Break Up -Refer Taxes and codes Description
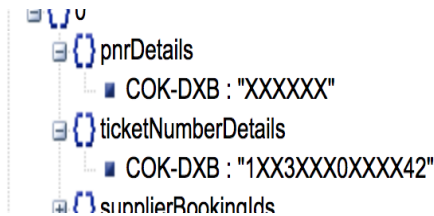   ◆ NCM - Net Commission

- ● TDS - TDS
- ● OT - Gross Commission
- ◆ SSRP - SSR Price
  - ● SP - Seat Price
  - ● MP - Meal Price
  - ● BP - Baggage Price

- ◆ **travellerInfos -** which has PNR , SSR Related information to Booked Passengers.
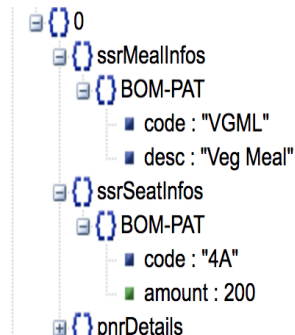 - ● pnrDetails - key - Segment - "DepartureCode-ArrivalCode":"Airline PNR"
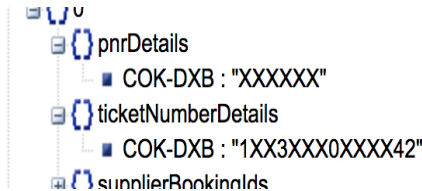
  - ● ticketNumberDetails - key - Segment -
    "DepartureCode-ArrivalCode":"ticketNumber"

  - ● ssrMealInfos,ssrBaggageInfos,ssrSeatInfos,ssrExtraServicInfos will followed by
    key : Segment - "DepartureCode-ArrivalCode":"corresponding SSR Information"

- ➔ **gstInfo -** GST Info which passed in Booking request for billing.

7. Release PNR:

This method is used to Release the Pnr that is currently on HOLD status, when Customer don't want to confirm the pnr.Once this API Called , PNR's will be released from Supplier (PNR will be cancelled) , then to confirm the Status of this API , again Booking Details API need to be called Order Status : UNCONFIRMED (which confirms PNR is released). If Order Status is not UNCONFIRMED, then can contact the Tripmaza Team.

URL: https://st-agent.tripmaza.com/oms/v1/air/unhold
Method: POST
Sample Logs Link : Click Here to take examples.

Sample Request View :

```
bookingId : "TMS104000003147"
[ ] pnrs
    0 : "XXXXXA"
    1 : "XXXXXB"
```

Sample Response View :

```
{} status
    success : true
    httpStatus : 200
```

Request Payload:

Key	Parent	Type	Description	Mandatory/ Optional
bookingId	root	String	Id of the booking which needs to be released	Mandatory
pnrs	root	Array of String	Pnrs that need to be released	Mandatory

8. Amendment API

This method is used to apply cancellation over a booking. It consists of three apis namely:
1. **Get Amendment Charges (Optional)** -To get the charges of the cancellation over a booking.It won't apply amendment over the booking.
2. **Submit Amendment -** To apply amendment on a booking this api must be called. It will provide an **amendmentId** which is used to check/poll the status of the amendment.
3. **Get Amendment Details -** To get the status and other details of an applied amendment.

Get Amendment Charges:

This is an optional api request to know the amendment charges for Cancellation. For any amendment prerequisite is that the booking must be in SUCCESS state. It will provide amendment charges and refundable amount pax wise for the booking or if the charges are not present then "**Please contact the support team to know the amendment charges**" in response.

URL: https://st-agent.tripmaza.com/oms/v1/air/amendment/amendment-charges
Method: POST
Sample Logs Link : Click Here to take examples.

Request Structure :

➔ **bookingId**
➔ **type**
➔ **trips -** ArrayList of trips for which amendment is to be raised (If Particular Journey to be cancelled)
   ◆ **src:** departure airport code
   ◆ **dest:** arrival airport code
   ◆ **departureDate -**Departure travel date (YYYY-MM-DD)
   ◆ **travellers :** list of travellers from a particular trip in a booking for which amendment is to be raised.
      ● **fn -** first name of traveller
      ● **ln -** last name of traveller

1. **Get amendment charges request for complete booking:** Submitting only the booking id will provide charges for all the pax types for the whole complete/booked trip.

```
JSON
    bookingId : "xxxxxxxxxx"
    type : "CANCELLATION"
```
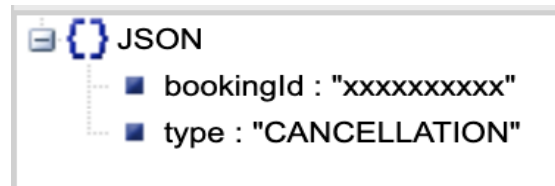
2. **Get amendment charges request for some selected travellers** : Submit travellers details along with their trip details. Charges will be returned only for the given pax types for the given trips.

```
JSON
    bookingId : "xxxxxxxxxx"
    type : "CANCELLATION"
    trips
        0
            src : "DEL"
            dest : "DXB"
            departureDate : "2020-09-23"
            travellers
                0
                    fn : "xxxxx"
                    ln : "xxxx"
```
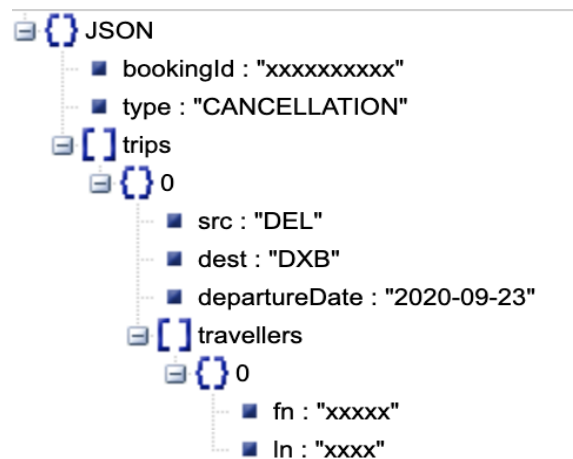
3. **Get amendment charges request for a particular trip:** If charges are required for all pax types in a particular trip then submit trip details without any traveller details.

```
JSON
    bookingId : "xxxxxxxxxx"
    type : "CANCELLATION"
    trips
        0
            src : "DEL"
            dest : "DXB"
            departureDate : "2020-09-23"
```

Response Structure

- ➔ **bookingId**
- ➔ **trips**
  - ◆ **src -** source airport code
  - ◆ **dest -** destination airport code
  - ◆ **departureDate -** Departure travel date(YYYY-MM-DD)
  - ◆ **flightNumbers -** flight numbers list (includes connecting flights flight numbers)
  - ◆ **airlines -** airlines code
  - ◆ **amendmentInfo -** (Map of PaxType,PaxAmendmentCharges)
    - ● **amendmentCharges -** Amendment Charges for Particular paxtype
    - ● **refundAmount -** Refundable Amount for Particular paxtype
    - ● **totalFare** - Total Fare for Particular paxtype

Sample Response

```
bookingId : "TMI01330019559"
trips
  0
    src : "DEL"
    dest : "BOM"
    departureDate : "2020-10-12T01:30"
    flightNumbers
      0 : "530"
    airlines
      0 : "G8"
    amendmentInfo
      ADULT
        amendmentCharges : 3000
        refundAmount : 8024
        totalFare : 11024
      CHILD
        amendmentCharges : 3000
        refundAmount : 8024
        totalFare : 11024
      INFANT
        amendmentCharges : 0
        refundAmount : 1500
        totalFare : 1500
status
  success : true
  httpStatus : 200
metaInfo
```

Submit Amendment : (Cancellation)

This is used to apply an amendment on a booking.After applying, an "amendmentId" will be returned in response which will be used to check/poll the status of the amendment.

URL: https://st-agent.tripmaza.com/oms/v1/air/amendment/submit-amendment
Method: POST
Sample Payloads :Click Here to take examples.
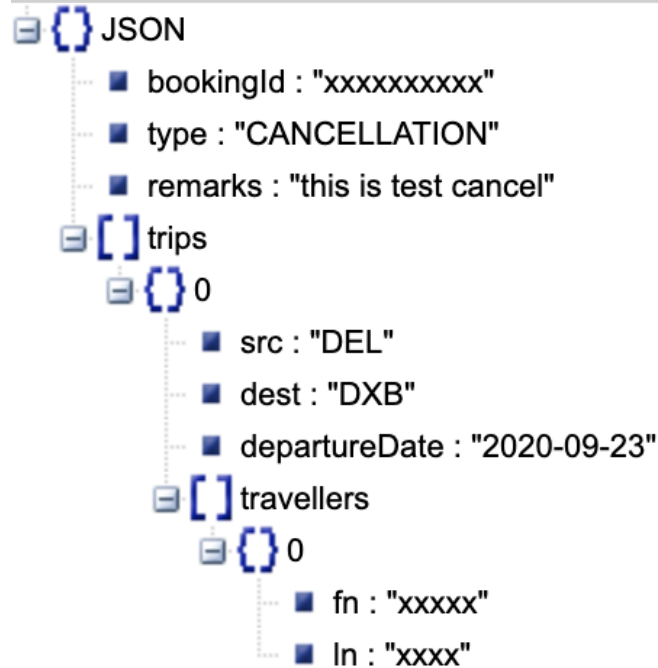
Request Structure :

> ➔ **bookingId**
> ➔ **type**
> ➔ **remarks-** Mandatory to send some remarks to submit an amendment.
> ➔ **trips :** ArrayList of trips for which amendment is to be raised
>> ◆ **src -** departure airport code
>> ◆ **dest -** arrival airport code
>> ◆ **departureDate -** Departure travel date(YYYY-MM-DD)
>> ◆ **travellers-** list of travellers from a particular trip in a booking for which amendment is to be raised.
>>> ● **fn** - first name of traveller
>>> ● **ln** - last name of traveller

Submit amendment request for complete booking: Submitting only the booking id with remark will apply amendment on the complete trip.

```
⊟ {} JSON
      ■ bookingId : "xxxxxxxxx"
      ■ type : "CANCELLATION"
      ■ remarks : "this is test cancel"
```

Submit amendment request for some selected travellers : Submit travellers details along with their trip details. Amendment will be applicable only on the given trip and given travellers.

```
JSON
    bookingId : "xxxxxxxxxx"
    type : "CANCELLATION"
    remarks : "this is test cancel"
    trips
        0
            src : "DEL"
            dest : "DXB"
            departureDate : "2020-09-23"
            travellers
                0
                    fn : "xxxxx"
                    ln : "xxxx"
```

Submit amendment request for a particular trip: If amendment needs to be submitted for all pax types in a particular trip then submit trip details without any traveller details.

```
JSON
    bookingId : "xxxxxxxxxx"
    type : "CANCELLATION"
    remarks : "this is test cancel"
    trips
        0
            src : "DEL"
            dest : "DXB"
            departureDate : "2020-09-23"
```

Response Structure :

- **bookingId**
- **amendmentId -** Should be used for polling current amendment status

Sample Response
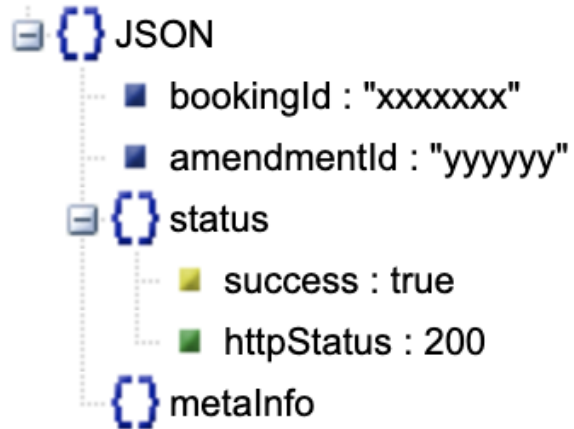
Get Amendment Details:

This request is used for polling the status of amendment, amendment and refund charges. It requires an **amendmentId** returned in the submit amendment response . If the amendment status is in REQUESTED state polling must be done 4-5 times with elapsed 10sec to get the status.(even more than 5 times still in REQUESTED State, then contact the support team to know more about the amendment status).

URL: https://st-agent.tripmaza.com/oms/v1/air/amendment/amendment-details
Method: POST

Sample Logs Link : Click Here to take examples.

Request Sample:

Request Structure

- **amendmentId**: AmendmentId (from submit amendment response)

Response Structure :

```
JSON
    bookingId : "xxxxxxx"
    amendmentId : "yyyyyyy"
    amendmentStatus : "SUCCESS"
    amendmentCharges : 0
    refundableamount : -3740
    trips
        0
            src : "DEL"
            dest : "MAA"
            date : "2020-09-24T04:50"
            flightNumbers
                0 : "6834"
                1 : "171"
            airlines
                0 : "6E"
            travellers
                0
                    fn : "TEST1"
                    ln : "TEST1"
                    amendmentCharges : 3067.7
                    refundableamount : 1870
                    totalFare : 4937.7
                1
                    fn : "TEST2"
                    ln : "TEST2"
                    amendmentCharges : 3067.7
                    refundableamount : 1870
                    totalFare : 4937.7
    status
        success : true
        httpStatus : 200
    metaInfo
```

Response Structure : It consists of the amendment details

- ➔ **bookingId** - Booking Id
- ➔ **amendmentId** - Amendment Id
- ➔ **amendmentStatus** -Status of amendment
- ➔ **amendmentCharges -** Total Amendment Charges of selected trips
- ➔ **refundableAmount** - Total Refund Amount of selected trips
- ➔ **totalFare -** Total fare of Booking
- ➔ **trips** : It is a list consisting of the tripwise amendment details
    - ◆ **src-** source airport code
    - ◆ **dest-**destination airport code
    - ◆ **departureDate-** travel date
    - ◆ **flightNumbers -**flight numbers list
    - ◆ **Airlines-**airlines list
    - ◆ **travellers**- list of traveller with amendment charges details
        - ● **fn -**traveller first name
        - ● **ln -**traveller last name
        - ● **amendmentCharges** - traveller amendment charges for particular trip
        - ● **refundAmount-** traveller refund amount for particular trip
        - ● **totalFare-** traveller total fare for particular trip

6.Common Object Structure:

tripInfo:

Key	Parent	Type	Description	Mandatory/ Optional
sl	tripInfo	Array of segmentInfo	Array of segments for the current trip.	Mandatory

segmentInfo:

Key	Parent	Type	Description	Mandatory/ Optional
id	segmentInfo	String	Id which generated during review for ssr	Optional
fD	segmentInfo	Object	Flight Designator (Marketing airline,Equipment and Flight number)	Mandatory
stops	segmentInfo	Integer	Count of the stops in current segment.	Mandatory
duration	segmentInfo	Integer	Travel duration of the current segment in minutes.	Mandatory
cT	segmentInfo	Number (Long)	Connecting Time Between two segments	Optional
da	segmentInfo	Object	Departure Airport	Mandatory
aa	segmentInfo	Object	Arrival Airport	Mandatory
dt	segmentInfo	String	Departure date and time. Format: YYYY-MM-DDTHH:MM	Mandatory
iand	segmentInfo	boolean	arrivalNextday	Optional
at	segmentInfo	String	Arrival date and time. Format: YYYY-MM-DDTHH:MM	Mandatory
isRs	segmentInfo	Boolean	Identify if Return	Optional

			Segment	
sN	segmentInfo	Integer	Segment Number	Mandatory
ssrInfo	segmentInfo	Object	Special service requests	Optional

fD: [Flight Designator - Marketing Airline,Equipment Type & Flight Number]

Key	Parent	Type	Description	Mandatory/ Optional
al	fD	Object	Airline Information	Mandatory
code	al	String	Airline code	Mandatory
name	al	String	Airline name	Mandatory
isLcc	al	Boolean	True if the airline is LCC else false	Mandatory
fN	fD	String	Flight Number	Mandatory
eT	fD	String	Equipment Type	Mandatory

ssrInfo:

Key	Parent	Type	Description	Mandatory/ Optional
MEAL	ssrInfo	Array of Object	Meal Information	Optional
BAGGAGE	ssrInfo	Array of Object	Baggage Information	Optional

SSR Info:

Key	Parent	Type	Description	Mandatory/ Optional
code	Object	String	Unique code of each ssr	Mandatory
desc	Object	String	Description which needs to be shown in UI	Mandatory

ADULT / CHILD/ INFANT:

Key	Parent	Type	Description	Mandatory/ Optional
fC	ADULT (or) CHILD (or) INFANT	Object	Fare Components	Mandatory
afC	ADULT (or) CHILD (or) INFANT	Object	Additional Fare Components	Optional
sR	ADULT (or) CHILD (or) INFANT	Integer	Seat Remaining	Optional
bI	ADULT (or) CHILD (or) INFANT	object	CheckIn & Hand Baggage	Optional
iB	bI	String	Check In Baggage	Optional
cB	bI	String	Cabin Baggage	Optional
rT	ADULT (or) CHILD (or) INFANT	Integer	Refundable Attribute 0: Non-Refundable, 1: Refundable, 2: Partial Refundable	Mandatory
cc	ADULT (or) CHILD (or) INFANT	String	Cabin class	Mandatory
cB	ADULT (or) CHILD (or) INFANT	String	Class Of Booking	Mandatory
fB	ADULT (or) CHILD (or) INFANT	String	Fare Basis	Mandatory
mI	ADULT (or) CHILD (or) INFANT	Boolean	Meal Inclusive	Optional

searchQuery:

Key	Parent	Type	Description	Mandatory/ Optional
routeInfos	searchQuery	Array of rountInfo	This contains the list of route details	Mandatory
preferredAirline	searchQuery	Array of Object	List of Airline codes that are preferred during search.	Optional
cabinClass	searchQuery	String	Cabin class for which search is done.	Mandatory
paxInfo	searchQuery	Object	Pax details	Mandatory

ADULT	paxInfo	number	Number of ADULT in search	Mandatory
CHILD	paxInfo	number	Number of CHILD in search	Mandatory
INFANT	paxInfo	number	Number of INFANT in search	Mandatory
searchModifiers	searchQuery	Object	Additional (if only direct or connecting flight)	Optional
isDirectFlight	searchModifiers	Boolean	True or False	Optional
isConnectingFlight	searchModifiers	Boolean	True or False	Optional

Preferred Airlines

Key	Parent	Type	Description	Mandatory/ Optional
code	preferredAirline	String	Airline code	Mandatory

Refundable Type

Key	Parent	Type	Description	Mandatory/ Optional
rT		Integer		Mandatory

travellerInfo:

Key	Parent	Type	Description	Mandatory/ Optional
pt	travellerInfo	String	PaxType of the passenger	Mandatory
ti	travellerInfo	String	Passenger Title	Mandatory
fN	travellerInfo	String	FirstName of the passenger	Mandatory
lN	travellerInfo	String	LastName of the passenger	Mandatory

dob	travellerInfo	String	Date of Birth if paxtype is INFANT. Accepted type: YYYY-MM-DD	Mandatory only if pt is INFANT
ff	travellerInfo	Object	Frequent flyer object for respective passenger	Optional
[airline_code]	ff	String	Frequent flyer number with airline code as key. Eg: "ff":{"AI":"23422"}	Optional
ssrMealInfos	travellerInfo	Array of Object	Meal Information	Optional
code	ssrMealInfos	String	Unique key from AirReviewRespons e if customer selected any	Optional
key	ssrMealInfos	String	Unique key from AirReviewRespons e on which segment customer selected	Optional
ssrBaggageInfos	travellerInfo	Array of Object	Baggage Information	Optional
code	ssrBaggageInf os	String	Unique key from AirReviewRespons e if customer selected any	Optional
key	ssrBaggageInf os	String	Unique key from AirReviewRespons e on which segment customer selected	Optional
statusMap	travellerInfo	Object	Pax wise status (pending/reissue...)	Optional

routeInfo:

Key	Parent	Type	Description	Mandatory/ Optional
fromCityOrAirport	routeInfo	Object	Details of Departure City and Airport	Mandatory
toCityOrAirport	routeInfo	Object	Details of Arrival City and Airport	Mandatory
travelDate	routeInfo	String	Date of Departure. Format: YYYY-MM-DD	Mandatory

Order Detail:

Key	Parent	Type	Description	Mandatory/ Optional
bookingId	root	String	Unique Booking Id	Mandatory
bookingUserId	root	String	API booking User Id	Mandatory
amount	root	Float	Total cost for the booking	Mandatory
markup	root	Float	Markup price of the booking	Mandatory
deliveryInfo	root	Object	Object containing the details of delivery emails and mobile numbers	Mandatory
status	root	String	Status of the booking	Mandatory
createdOn	root	String	Timestamp when the booking was created. Format: YYYY-MM-DDTHH:mm:ss.SSS	Mandatory
itemInfos	root	Object	Trip which booking has been created (contains all info about segment, pnr,ticket Number on	Mandatory

			segments and paxwise)	

Delivery Info:

Key	Parent	Type	Description	Mandatory/ Optional
emails	deliveryInfo	Array of String	List of emails to which booking details will be sent	Mandatory
contacts	deliveryInfo	Array of String	List of mobile numbers to which booking details will be sent	Mandatory

Response Status :

Key	Parent	Type	Description	Mandatory/ Optional
success	status	Boolean	True if success and False if failure	Mandatory
httpStatus	status	number	Status code of the API call	Mandatory

Gst Info :

Key	Parent	Type	Description	Mandatory
gstNumber	gstInfo	String	GST number which is entered while booking	Mandatory
registeredName	gstInfo	String	GST name	Mandatory
mobile	gstInfo	String	GST Mobile	Optional
email	gstInfo	String	GST Email	Optional
address	gstInfo	String	GST Address	Optional

AmendmentRequest :

Name	Type	Mandatory	Explanation	Mandatory/Optional
bookingId	String	yes	The booking id received while booking the ticket	Mandatory
type	ENUM	yes	Ex: CANCELLATION	Mandatory
amendmentId	String	Yes,only when polling the status of previously raised amendment	It is the id of amendment received after raising the amendment for a booking	Mandatory ,only while polling the amendment status
remarks	String	yes	Any remarks for the amendment request	Mandatory, Only in submit amendment request
trips	AmendmentTrip	no	If amendment is to be raised for specific trips in the booking ,then this object should be passed.	optional

AmendmentTrip:

Name	Type	Mandatory	Explanation	Mandatory/Optional
src	String	yes	Departure airport code for the trip.	Mandatory
dest	String	yes	Arrival airport code for the trip.	Mandatory
departureDate	LocalDate	yes	Departure date of the trip.	Mandatory
travellers	List of AmendmentTraveller	no	It is a list of all the travellers for which the amendment should be raised. If not passed in request then the request is considered for all the travellers in that particular trip.	optional

AmendmentTraveller

Name	Type	Mandatory	Explanation	Mandatory/Optional
fn	String	yes	First name of traveller	Mandatory
ln (small "L")	String	yes	Last name of traveller	Mandatory

AmendmentChargesResponse

Name	Type	Description
bookingId	String	Booking Id
errors	List of errors	Errors in the request if any
trips	List of trips	Tripwise amendment charges

AmendmentTripCharges

Name	Type	Description
src	String	Source of the single trip
dest	String	Destination of the single trip
departureDate	LocalDateTime	Departure date of trip
flightNumbers	Set of String	Flight number
airlines	Set of String	Airline codes
amendmentInfo	Map of AmendmentInfo Object	PaxWise charges Ex Adult,Child,Infant
travellers	List of Traveller Object	Traveller Wise charges details

PaxAmendment Charges

Name	Type	Description
message	String	message
amendmentCharges	Double	Each Pax charge
refundAmount	Double	Each Pax charge
totalFare	Double	Each Pax charge

Amendment Traveller Detail :

Name	Type	Description
fn	String	First Name of traveller
ln (small 'L')	String	Last Name of traveller
amendmentCharges	Double	Traveller charge
refundAmount	Double	Traveller charge
totalFare	Double	Traveller charge

Amendment Response

Name	Type	Description
bookingId	String	Booking Id
amendmentId	String	Amendment Id
amendmentStatus	String	Status of amendment
amendmentCharges	Double	Charges for whole booking
refundableAmount	Double	Charges for whole booking
totalFare	Double	Charges for whole booking
trips	List <AmendmentTripCharges>	Tripwise details

7. Tax Codes and Description

These are below Fare Component Codes which are followed by Tripmaza.

Key	Description
AT	Airline Taxes
BF	Base Fare
MF	Tripmaza Management Fees
TF	Total Fare
NF	Net Fare
TAF	Taxes And Fees
WO	Airport Tax
MFT	Tripmaza Management Fee Tax
UDF	User Development Fee
AGST	Airline GST
IGST	IGST
TAF	Taxes and Fee
MF	Management Fee
OT	Other Tax
OB	Carrier Ticketing Fee
YR	Carrier Misc Fee
RCF	Travel Fee
YQ	Fuel Surcharge
WO	Airport Tax
PSF	Passenger Service fee
WC	Airport Arrival Tax
YM	Development Fee

OC	Other Charges
BP	Baggage Price
MP	Meal Price
SP	Seat Price
CCF	Tripmaza Cancellation Fee
CCFT	Tripmaza Cancellation fee tax
ACF	Airline Cancellation Fee
ACFT	Airline Cancellation fee tax
ARF	Airline Reschedule Fee
ARFT	Airline Reschedule fee tax
AI	Additional Incentive
NCM	Net Commission
BCM	Commission
IATA	IATA Commission
PLB	PLB
SM	Segment Money
CGST	Central GST
SGST	State GST
IGST	Integrated GST
UGST	Union Territory GST
XT	Additional Taxes
PF	Payment Fee

8. Error Codes and Message

Error Code	Message
400	Bad Request
404	Resource Not Found
407	You are not allowed to do search with given parameters. Please change your search criteria
408	Access Denied
411	You don't have enough permission to perform this action
412	UnAuthorized Access!, The provided API key is invalid.
802	Your account is currently inactive
805	Gst number (GSTIN) should be 15 characters only and valid GST
806	Invalid Email or Mobile
810	There is something went wrong with the backend service. It could be due to invalid/bad data.
816	Duplicate request.
819	Invalid Address
1000	Request flight is not longer available.Please try different flight
1001	Infant can't be greater than no of adults.
1002	Child can't be greater than no of adults.
1003	Travel date should be in ascending order . Next travel date can't be less than the previous travel date.
1004	Travel date can't be after year.
1005	Origin and Destination can't be the same.
1006	No of passenger can't be greater than 9.
1007	First Name shouldn't be blank and it shouldn't contain any whitespace character.
1008	Last Name shouldn't be blank and it should contain only alphabet and

	whitespace characters.
1009	Price Id's cannot be empty
1010	Any two passengers can't have same name
1012	Adult's age should be between 12-100 years from the date of departure.
1013	Child's age should be between 2-12 years from the date of departure.
1014	Infant's age should be between 0-2 years from the date of departure.
1015	Total amount passed in payment doesn't match with total order Amount
1019	Minimum time between two consecutive trips does not satisfy the criteria.
1020	Same sector same day is not allowed
1051	Adult's dob required.
1052	Child's dob required.
1053	Infant's dob required.
1056	Seat Selection Not Applicable
1057	No booking found for booking Id
1059	Hold time limit expired
1064	Please, enter passport number.
1065	Please, enter valid passport issue date.
1066	Please, enter valid passport expiry date.
1067	Passport should not expire within 6 months from travel date.
1068	Travel date can't occur before Passport Issue Date.
1071	Fare is no longer available for this booking
2540	Please contact support team to know the amendment charges
2541	Amendment has already been raised for requested booking Id
2542	Invalid Trip/Passenger requested , Check Trip or Passenger details submitted
2543	Amendment Not allowed because travel date is passed

2544	Amendment already raised, call support or check amendment for updates
2545	Invalid Amendment type requested
2546	Invalid Amendment Id sent in request
2549	Amendment Type not passed in request
2550	Departure Airport cannot be empty or Blank
2551	Arrival Airport cannot be empty or blank
2552	Departure Date cannot be empty or Blank
2553	Amendment unable to be raised due to #custom , Please contact support team
2554	Either Booking id or amendment type is invalid
2555	Invalid amendment id
2556	Remarks is mandatory to send in request
2557	Either sector or departure date submitted is invalid

9. UAT Certification (Test Case)

API Partner needs to go through a certification process in order to get live access. Here we are mentioning the certification process which is Mandatory to be completed when the client is done with the API Integration.

1. Published Fares
2. Special Return
3. Passport Cases
4. With GST / Without GST
5. With SSR

S.no	Type	Sector *	Pax	Flight Type
Oneway				
1	Oneway	DEL - BOM	1A	DIRECT
2	Oneway	DEL - BOM	1A	CONNECTING
3	Oneway	DEL - DXB	2A - 2C	DIRECT
4	Oneway	BOM - SIN	2A - 2C	CONNECTING
5	OneWay	MAA - DMK	3A - 2C	DIRECT
6	OneWay	DXB - BKK	5A - 3C - 2I	DIRECT
7	OneWay	DXB - BKK	5A - 3C - 2I	CONNECTING
7	OneWay	BOM - MAA	5A - 4C - 3I	DIRECT
Return				
1	Dom Return	DEL - BOM	1A	DIRECT
2	Dom Return	DEL - BOM	1A	CONNECTING
3	Return	DEL - DXB	2A - 2C	DIRECT
4	Return	BOM - SIN	2A - 2C	CONNECTING
5	Return	MAA - DMK	3A - 2C	DIRECT
6	Return	DXB - BKK	5A - 3C - 2I	DIRECT

7	Return	DXB - BKK	5A - 3C - 2I	CONNECTING
7	Return	BOM - MAA	5A - 4C - 3I	DIRECT
8	Dom Return	DEL - BOM	3A - 2C	DIRECT
9	Dom Return	DEL - BOM	5A - 3C - 2I	CONNECTING
Multicity				
1	Domestic	Any	5A-3C	CONNECTING
2	Domestic	Any	5A-4C-1I	DIRECT
3	Intl	Any	4A-2C-2I	DIRECT
4	Intl	Any	4A-2C-2I	CONNECTING

Note:

1. Please provide Request & response for all methods that you are implementing.
2. Certification request by API Partner (API Partner makes a certification request by providing JSON Request/ Response along with confirmation numbers for the test cases)
3. Test Cases Verification - will verify the test cases using JSON Request/ Response and suggest, if there is any change required. The turnaround time for the test case verification is around 1-4 working days.
4. Sign off and Live access - On successful completion of all the above steps, Tripmaza.com will give sign off and provide Live access details to the client. It is strongly recommended that the API Partner also completes their UAT before making the services live at their end.

Going Live Steps :

1. Certification Process

- Clients must provide all the test cases mentioned in the document.
- Logs should be in json format.
- Clients should not send their logs.
- No modification in logs should be there.
- API Key also needs to be attached along with the logs.
- There should be different files for each request and response.
- Passenger Name should be proper. (TBA, Test cannot be passed for certification)

Bookings Should have :

1. Published Fare Booking - Domestic, International
2. Special Return Fare Booking
3. Booking with Passport
4. Booking with GST
5. Booking with SSR (Baggage,Meal)

2. Post Certification

- Point your Integration to the Production environment endpoints (Urls mentioned in document) and apikey.
- It is important for you to provide the current IP address ranges to whitelist at our end.

Frequently Asked Questions

The following are support queries that are frequently asked by developers to the API Support team. This is not an exhaustive list, and you can find more detailed help in our guidelines and specifications.If you need technical help and cannot find it here or elsewhere in the documentation, then please contact the Tripmaza support team.

Whenever you raise any queries on support email. Below things are necessary for a speedy response from the support team.

1. URL on which you are trying/sending a request.
2. API key used in the request
3. Request and Response json
4. API version.

General Errors

- **Invalid Access. Send valid authorization token.Either token is invalid or it's expired** : Please pass the api key in the request headers(mandatory for all APIs), Please refer the document for sample code.
- **Unauthorized Access Error** : API Key passed in headers is wrong, Please contact the tech support team.
- **Insufficient balance** : Please contact the Business Team to topup the account.
- **Inactive Account :** Please contact the Business Team , your current account or apikey is Disabled.

* ALL System/Flight Related error mentioned on Error code and Message

Q. How to pass headers in request ?
   A. Refer to Sample Code

Q. Where to find the Airport and Airline Related Information?
    A. For Airport Codes and Airline Info , Please refer to IATA Website

Q. Is there an API that checks source and destination or do we need to maintain masters at our level?
   A. No, we don't have the api to check source and destination. On search api request you have to pass both.

Q. **How many passengers can be passed in Search API at minimum and maximum?**

    A. Minimum: 1 adult

      Maximum: Sum of adult and child passengers should be equal or less than 9. Number of infants should be equal or less than the number of adults. If these conditions are not satisfied then you will get errors.

Q. **What are the possible values for CabinClass types?**

    A. "ECONOMY", "PREMIUM_ECONOMY" , "BUSINESS" , "FIRST"

Q. **For how long are the Search PriceIds valid?**

    A. PriceIds are Valid for 30 minutes

Q. **Is there a way to filter Price, Departure from time, Arrival to time and 2 stops and more than 3?**

    A. No, we don't have this flexibility in API.

Q. **How would I know If an Segment has connecting flights?**

    A. There is an element in response to Search (TripInfo :: sI) named as "stops" and "so". It has all the information for connecting stops.

        a. stops - number of stops

        b. so - stopover airport information

Q. **Can you please let me know how to get Layover information in Search API between each flight?**

    A. This can be checked in (sI :: cT) object. if there is any connecting flight. If there is no connecting flight this field will come in response.

Q. **Which id can be used in request to fetch Fare-Rule?**

    A. Particular Trip priceId from search Response (During Search/Review) , in case after ticket booked BookingId has to be passed in request with flowtype.

Q. **How can I get Description of Each Fare/Tax Component ?**

    A. Refer Tax and Code Section

Q. **What is the difference in review requests for Domestic One Way and Domestic Return and Domestic multi city?**

    A. **In Case of Domestic Return** - 2 price id has to be passed (1 flight selected from Onward and another from Return)- in a single review request.

      In Case of Domestic Multi-city - N (Maximum 6) of price id has to be passed - based on route info

      requested in search (in single review request.)

      In case of International Domestic/ Multi city - Only one price id has to be passed in review request.

Q. **In case of Domestic Return or Multi City , will we get more than 1 bookingId?**
  A. No, In all cases there will be only a single bookingId.

Q. **Is Passport Info mandatory for all International sectors?**
  A.   No, It depends on the supplier , We will return the parameters(**pm**) in the review response that need to be sent in the booking request. If its not available in response it means passport is not mandatory

Q. **Is Gst Applicable for All bookings?**
  A. No, it depends on the supplier, whether that supplier supports gst or not,This information will be returned in review response If the value of igm is true then gst information has to be passed mandatorily in the booking, It it's false and gstappl is true it means gst option is available against this price.

Q. **When to call Seat API ?**
  A. This is only allowed to call only when "**isa**" is true in the review response.

Q. **Do We need to validate fare before going to book?**
  A. No , There is no need to validate fare before calling book api , If there is any fare Jump , then it will be returned in review response. (**alerts**) - Example Link

Q. **Do we need to validate fare in case of Hold booking , before going to confirm?**
  A. Yes , After hold , you need to call fare-validate api to check if the fare is still available or not, if this api returns success, only then proceed for confirm booking. Else that fare is no longer available and you will have to create a new booking.
   Note : In case if there is any FARE JUMP (which can happen in GDS Booking) then this API will return false and you have to create a new booking.

Q. **What is the Age Limit of Child & Infant booking?**
  A. An Infant had to be less than 2 years , A child should be less than 12 years.

Q. **What are all the Passengers Titles that can be passed?**
  A. For Adults : (Mr , Mrs , Ms) , for Child (Ms , Master), for Infant (Ms , Master).

Q. **An itinerary has a connecting flight then, will PNR and E Ticket be different for connecting flights?**
  A. No. PNR and E Ticket will be the same as validating carrier(issuing carrier) for all connecting flights will be the same.

Q. **API for Date/Time change**
  A. Review/Revalidate API , **(sl :: dt and at) will get updated** , if supplier updates new departure and arrival time.

Q. Is all Gst Fields (GST Registered Number,Name, Mobile, Email & Address) Mandatory ?
  A. Yes, all fields are mandatory, when ever passing gst fields (all relevant fields need to be passed in request).

Q. **How to support Special Return Fare in case of Domestic Return ?**
  A. In Case of Special Return Fare (FareType Should be SPECIAL_RETURN) and current "**sri" tag from PriceInfo has to chosen from ONWARD FLIGHT and selected "sri" tag has to be matched RETURN FLIGHT Price "msri" tag should contain selected "sri" tag.** (vice versa for Return flights to onward flights). Refer More on : Page No : 20, 21

Q. **What is the time duration between hold booking and confirm booking?** -
  A. In Booking Detail Response under (**itemInfos :: timeLimit**) can be referred , and in case of no time limit specified in response it means our system doesn't have any information about the same. It would be an individual client what they want to display to their customers.

Q. **What Payment Medium type will be used in Booking ?**
  A. API Partner no need to send Payment Medium in Book or Confirm Book API Request, Tripmaza system will decide which payment should be used for API Partner.

Q. **What are the SSRType supported in the Tripmaza platform ?**
  SSR is supported only for LCC. Currently our system doesn't support SSR for GDS bookings.
  A. BAGGAGE
  B. MEAL
  C. SEAT

Q. How can I add SSR to Travellers while Booking ?
  A. In AirReviewResponse for every (**segment :: ssrInfo)** can be used, where code is primary key for which is selected for traveller and key will be segment key ,that need to be passed in Book Request.
  B. In AirReviewResponse , ssrInfo will be present only if the quantity available is greater than requested passenger count.
  C. In Case of connecting segments (Baggage SSRType) -
    a. if the amount field will be null to represent whatever selected on the first segment , the same will be applicable on all consecutive segments.
    b. If the amount field is not null , then the corresponding segment is applicable to selected different baggage options. (Example : Thru Flight Scenarios).
      Refer more on : Page No : 44

Q. **Can I pass firstname or lastname as blank ?**
  A. Yes, anyone at a time can be blank.

Q. **Can I pass the default/dummy DOB ?**
  A. No, In case of DOB/Passport mandatory , exact DOB of Passenger/Traveller needs to be passed.
  B. you can validate the dob object under condition in the review response for the same.

Q. **What are the FareTypes supported in Search Response ?**
  A. Please check this information with the Business Team.

Q. **What cases need to be covered in the Certification Process ?**
  A. Published Fare Booking - Domestic, International
  B. Special Return Fare Booking
  C. Booking with Passport
  D. Booking with GST
  E. Booking with SSR (Baggage,Meal).

Sample logs Link for Certification logs : - Link

Q. **Is Duplicate Booking allowed In Tripmaza ?**
  A. No, System will check Same Itinerary and Same Passenger Booked by the same agent , then System wont allow to create the booking for a new request.

Q. **Is Multiple Current Code supported In Tripmaza ?**
  A. No, Tripmaza Air API Supports Only INR

Q. **In the search API response, we are getting all types of cabin class results, Whereas in the request, we have passed Cabin Class as "ECONOMY" only.**

Case 1 : If Domestic (India)
  A. If we have ECONOMY in the request then we are supposed to get results for all the cabin classes.
  B. If we have PREMIUM ECONOMY in the request then we are supposed to get PREMIUM ECONOMY, BUSINESS and FIRST class.
  C. If we have BUSINESS on request then we are supposed to get BUSINESS and FIRST class.

Case 2 : If International
  A. Requested Cabin Class Results only be available.(Ex. Economy then only Economy Results).

Q. In case a flight appears in special return fare. Do we also get the same flight if available in normal fare. In other words do we get special return fare and normal onward and return fare in the same flight in the same search result?

    A. Yes, if flights are available in normal fare, then it will reflect in the same search results.

Q. **How can we identify if a fare is refundable or nonrefundable?**

    A. You can check the same from the "rT" tag under totalPriceList in search/review response.
        a. rT - Refundable Type
            i.   0 - Non Refundable
           ii.   1 - Refundable
          iii.   2 - Partial Refundable

Q. Can we select different baggage in 2 connecting segments(DEL-BLR--> DEL-BOM-BLR, DEL-BOM 30kg, BOM-BLR 60 Kg)?

    A. No, you have to select the same baggage details for all connecting segments.

Q. Can we pass Frequent flyers on booking requests?

    A. Yes, you can pass the same in book request under travellerInfo:

    "ff": {
    "AirlineCode": "frequent flyer number"
    }

Q. How can we identify validating and operating airlines?

    A. For Validating airlines you can refer to oB Tag.
        a. In case "oB" is not present in the response, we have to consider the "fD" tag. In this case both Marketing airline (Validating airline) and Operating airline are the same.
        b. In case the operating airline and marketing airline are different "oB" tag will be present in the response.