



भारतीय दूरसंचार विनियामक प्राधिकरण  
TELECOM REGULATORY AUTHORITY OF INDIA  
भारत सरकार /Government of India



F.No.12-37/2019-B&CS

Dated: 22<sup>nd</sup> October 2019

To,

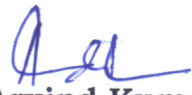
All DPOs (DTH/MSO/HITS/IPTV)

Telecom Regulatory Authority of India (TRAI) has released Telecommunication (Broadcasting and Cable) Services Standards of Quality of Service and Consumer Protection (Addressable Systems) (Second Amendment) Regulations, 2019 (6 of 2019) on 9<sup>th</sup> October 2019.

2. Through this regulation (second amendment), TRAI has mandated the DPOs to allow the consumers to access channels/ bouquets available on its platform and have ease in selection of channels and bouquets (addition/deletion) of their choice, view their subscription and modify the same through the TRAI's APP/Portal. DPOs are also mandated to share/exchange the information through API with the TRAI whenever the Authority asks for the same for ensuring integration with the TRAI's APP.

3. In the explanatory memorandum to the above said regulation it has been also mentioned that the Authority is in the process of finalising the API Specifications which will be communicated separately to the DPOs. After due consultation process and discussion with technical team of the DPOs, the Authority has finalised the API Specifications which are annexed as **Annexure**. These APIs are to be integrated with the APP/Portal developed by TRAI.

4. Ms. Archana Ahlawat, Jt. Advisor (IT) (jait.trai@gmail.com) will be the nodal officer from TRAI side for integrating API with the APP/Portal developed by TRAI. DPOs are hereby directed to share the URLs of production APIs as envisaged in API specification document (**Annexure**) by 21<sup>st</sup> November 2019.

  
(Arvind Kumar)  
Advisor (B&CS)

Encl: as above

## **Annexure**



### **Channel Selection System API Specifications**

**Telecom Regulatory Authority of India (TRAI)  
Mahanagar Door Sanchar Bhawan,  
Jawahar Lal Nehru Marg,  
New Delhi – 110002**

## Table of Contents

|   | <b>Page No.</b> |
|---|-----------------|
| 1. Glossary of Terms.....                           | 3               |
| 2. Introduction.....                                | 4               |
| 3. Objectives.....                                  | 4               |
| 4. Document Objectives.....                         | 4               |
| 5. TRAI IT Application .....                        | 5               |
| 6. Process Flow of the TRAI IT Application.....     | 5               |
| 7. API Description.....                             | 6               |
| 8. Validation of Subscriber .....                   | 7               |
| 9. Get Subscriber API.....                          | 8               |
| 10. Subscription Details.....                       | 10              |
| 11. Get Subscription API.....                       | 10              |
| 12. Platform offerings (Channels and Bouquets)..... | 15              |
| 13. Menu API.....                                   | 15              |
| 14. Get Channel API.....                            | 17              |
| 15. Get Bouquet API.....                            | 18              |
| 16. Set Subscription.....                           | 19              |
| 17. Set Subscription API .....                      | 20              |
| 18. Subscription Status API .....                   | 22              |
| 19. Response Code.....                              | 23              |

---

## 1. Glossary of Terms

---

- **DPO:** Distribution Platform Operator stands for MSO/DTH.
- **MSO:** Stands for Multiple System Operator. MSO is the industry term for "cable company". Each cable TV installation serving a community is known as a "cable system", and the operator of the system is called the "system operator". Most system operators run cable systems in more than one community. DTH is Direct to home
- **LCO:** LCO stands for Local Cable Operator.
- **JSON:** JSON stands for JavaScript Object Notation. JSON is a lightweight format for storing and transporting data.
- **URL:** URL is the abbreviation of Uniform Resource Locator and is defined as the global address of documents and other resources on the World Wide Web.
- **OTP:** OTP stands for one-time password. It is a password that is valid for only one login session or transaction, on a **computer** system or other digital device.
- **HTTP:** HTTP stand for HyperText Transfer Protocol. The communications protocol used to connect to Web servers on the Internet or on a local network (intranet). Its primary function is to establish a connection with the server and send HTML pages back to the subscriber's browser.
- **SUBSCRIBER:** A subscriber is a person who pays to receive channels regularly from operator. A subscriber will have a registered mobile no./customer ID/subscriber ID/VC Card.
- **HEADEND:** A headend is a facility that accepts TV signals as input from satellites, processes them into cable-quality signals, and then distributes them to homes and cable networks. It can be considered as the master distribution center where incoming television signals are received, selected, amplified and re-modulated, and sent for transmission to cable networks.
- **API:** Application Program Interface
- **JWT:** JSON Web Token
- **REST:** Representational State Transfer
- **IT Application:** TRAI Mobile App and Web portal

---

## 2. Introduction

---

- 2.1** This document defines specifications for Web APIs to allow easy interchange of information between a Distribution Platform and TRAI App & Portal. Such information includes the details of channels and bouquets that the platform offers to its subscribers, the current active subscription of a subscriber, and the updated subscription that the subscriber is desirous of setting.
- 2.2** The APIs are designed to enable such interchange without sharing information that either personally identifies a subscriber or which is not necessary to accomplish the task at hand.

---

## 3. Objectives

---

- 3.1** With TRAI's New Regulation for the Television and Broadcasting Sector having come into force, Subscribers have the freedom to select Television (TV) Channels they want to watch. The main objective of the API design is to develop App & Portal to assist subscribers in making their choice and accordingly configuring the subscription.
- 3.2** TRAI envisions a scalable model in which Subscriber (through Apps/Portals) can securely communicate with the platforms managed by DPOs. The IT application will facilitate subscribers to choose channels/bouquets of their interest among the offerings by their respective DPO. The IT application may suggest an optimum configuration of bouquets based on channels desired by the subscriber so as to reduce total monthly bill or offer other features such as suggested packages based on subscriber profile. The APIs by themselves, however, neither encourage nor facilitate subscriber profiling.

---

## 4. Document Objectives

---

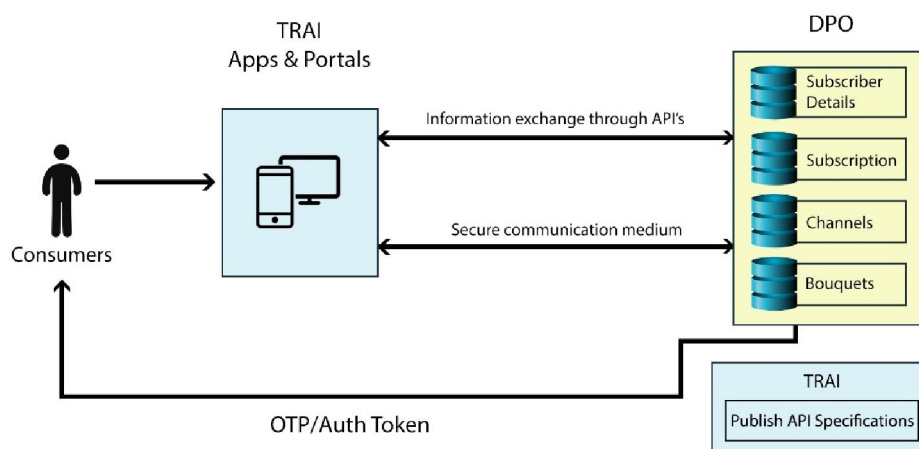
- 4.1** This document describes the base architecture of the IT application and elaborates the specifications for the data exchange APIs. These APIs shall be hosted by DPOs. All DPOs shall share the URL of hosted APIs with TRAI. These standard APIs shall ensure the interconnectivity among various platform which are based on different technology stacks.

---

## 5. TRAI IT Application

---

**5.1** The TRAI IT application shall support **APS** (Acquire subscriber current subscriptions from DPO, Provide options of channels/bouquets for selection, Set subscriber subscriptions of their choices on the DPO platform). All the three participants i.e. Subscriber, TRAI IT application & DPOs shall interact through Data APIs. The high-level architecture of the system is detailed below:



**5.2** TRAI IT application network communication shall be encrypted and secured, using SSL, to maintain the privacy and eligibility of the data travelling through the medium. Accordingly, it is proposed to implement JWT token. This will ensure only authenticated API requests are responded.

---

## 6. Process Flow of the TRAI IT Application

---

**6.1** The process flow is conceptualized as follow:

- I. Subscribers shall be able to access Apps/Portals developed by TRAI.
- II. Each Subscriber shall be validated through OTP against Subscriber ID or Registered mobile no. or VC number or Auth token.
- III. The validation of subscriber shall be done by DPO, when triggered by App/Portal.
- IV. Upon successful validation, subscriber will get session token to fetch further details i.e. subscriber's current subscription.

- V. Subscribers shall be able to modify their current subscription either by removing existing channel/bouquet and/or adding new channel/bouquet.
- VI. The App/Portal shall submit the changes in the subscriber's subscription to their respective DPO.
- VII. Subscriber can fetch the current status of subscription request.

**6.2** For implementation, TRAI App & Portal may cache DPOs offering at their end. The cache may be refreshed every 24 hrs or define on its own to avoid old data showing up.

**6.3** Upon selection of Channels/ Bouquets, TRAI App & Portal may optimize the channel selection to minimize cost or any other relevant optimization offerings.

---

## 7. API Description

---

**7.1** A set of REST APIs have been envisaged to facilitate the data exchange. Following are minimum set of REST APIs that shall enable information exchange. These APIs shall be hosted by DPOs. All DPOs shall share the URL of hosted APIs with TRAI. APIs shall provide information from DPOs (MSOs/DTHs) regarding:

- I. Validation of Subscriber
- II. Subscription details
- III. Channels on platform
- IV. Bouquets on platform
- V. Setting (request) Subscription changes on DPO platform
- VI. Fetching Subscription change request status

**7.2** The Channels and Bouquet API can be combined in one API – Menu API. Menu API shall provide information about all the offerings on the platform. The return object size may be bigger as compared to the Channel and Bouquet API called individually.

**7.3** This document contains the description, functionality, JSON (Request/Response) parameters, URL, method individually for each API.

---

## 8. Validation of Subscriber

---

- 8.1** This API will validate subscriber at DPO end. The subscriber validation can happen through subscriber's Subscriber ID or Registered mobile no. or VC number or Auth token.
- 8.2** Mobile number is the registered mobile number, which the subscriber has registered with its DPO. However, a subscriber can register multiple connections with same Mobile No. The Subscriber ID is allotted by DPO to its registered subscriber. For every unique connection there shall be unique Subscriber ID. Corresponding to subscriber ID there is unique Subscription No.
- 8.3** Subscriber shall provide its Subscriber ID to TRAI App/Portal. TRAI App/Portal shall send the same to DPO. DPO shall send the OTP SMS to the registered mobile no. of the subscriber. The SMS shall contain subscriber ID and OTP. Response SMS may also contain following text "Sample: TRAI app is requesting authorization to view your subscription. If you have requested, please share the authorization code. Do not share the code if you have not initiated this request."
- 8.4** The subscriber will submit the Subscriber ID or Registered mobile no. or VC number and OTP to the TRAI App/Portal. The validation will be done by DPO through TRAI App/Portal.
- 8.5** Auth token is the concept wherein every DPO hosts an interface (any of Apps, portal, IVR, SMS etc.) to provide Auth Token. Their subscribers can use any of the provided interface and in response DPO generates a Auth Token for each subscription mapped with a registered mobile number. For Multiple connections mapped with same registered mobile no. DPO shall generate and send multiple Auth Tokens corresponding to each subscriber ID. The validation will be done by DPO through TRAI App/Portal.
- 8.6** Upon successful validation, API will return session token, Subscription ID and some details required by the subscriber to identify the subscription which they wish to modify.



8.7 This can be done by calling the Get Subscriber API.

---

## 9. Get Subscriber API

---

### 9.1 API Detail: Get OTP

|                        |  |                 |           |           |  |
|------------------------|--|-----------------|-----------|-----------|--|
| <b>URL</b>             | /subscriber/doAuth/  |                 |           |           |  |
| <b>Method</b>          | GET  |                 |           |           |  |
| <b>Body Parameters</b> | S.no   | Name            | Mandatory | Data Type | Example  |
|                        | 1  | type            | Yes       | Integer   | 1 /2/3<br>1 for<br>Subscriber<br>ID, 2 for<br>Registered<br>mobile no.<br>and 3 for VC<br>number |
|                        | 2  | cons_identifier | Yes       | String    | "1234"   |
| <b>Response</b>        | <pre>{<br/>  "status": 200,<br/>  "message": "OTP has been sent"<br/>}</pre> |                 |           |           |  |

### 9.2 API Detail: Fetch subscription summary

|                        |                     |                 |           |           |  |
|------------------------|---------------------|-----------------|-----------|-----------|--|
| <b>URL</b>             | /subscriber/doAuth/ |                 |           |           |  |
| <b>Method</b>          | GET                 |                 |           |           |  |
| <b>Body Parameters</b> | S.no                | Name            | Mandatory | Data Type | Example  |
|                        | 1                   | type            | Yes       | Integer   | 1 /2/3<br>1 for<br>Subscriber<br>ID, 2 for<br>Registered<br>mobile no.<br>and 3 for VC<br>number |
|                        | 2                   | cons_identifier | Yes       | String    | "1234"   |

|                 |   |     |     |        |        |
|-----------------|---|-----|-----|--------|--------|
|                 | 3   | otp | Yes | String | "4578" |
| <b>Response</b> | <pre> {   "status": 200,   "accessToken": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXbzI9eyJzdWVudW9uLj0bJoSBg",   "tokenType": "Bearer",   // Corresponding to one mobile no there can one or more   subscriber ID   "subscriber": [     {       "subscriberID": "AB9875543"       "subscriptionId": "12345",       "amount": 525,       "type": "monthly",       "status": "inactive",       "activationDate": "2019-07-19T08:52:04.344+0000"     },     {       "subscriberID": "CD6677867"       "subscriptionId": "54321",       "amount": 450,       "type": "monthly",       "status": "inactive",       "activationDate": "2019-05-07T07:32:04.312+0000"     }   ] } </pre> |     |     |        |        |
|                 |   |     |     |        |        |

### 9.3 API Detail: Validate & Fetch subscription by Auth Token

|                        |                              |            |           |           |             |
|------------------------|------------------------------|------------|-----------|-----------|-------------|
| <b>URL</b>             | /subscriber/doAuth/authtoken |            |           |           |             |
| <b>Method</b>          | GET                          |            |           |           |             |
| <b>Body Parameters</b> | S.no                         | Name       | Mandatory | Data Type | Example     |
|                        | 1                            | auth_token | Yes       | String    | "fgdfdfdfd" |
| <b>Response</b>        | <pre> { </pre>               |            |           |           |             |

|  |   |
|--|---|
|  | <pre>“status”: 200, “accessToken”: “eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXZWQ9J0bJoSBg”, “tokenType”: “Bearer”, “subscriber”: [   {     “subscriberID”: “AB9875543”     “subscriptionId”: “12345”,     “amount”: 525,     “type”: “monthly”,     “status”: “inactive”,     “activationDate”: “2019-07-19T08:52:04.344+0000”   } ] }</pre> |
|--|---|

---

## 10. Subscription Details

---

**10.1** This API will fetch details of subscribed channels & bouquets currently opted by the subscriber. This can be done by calling Get Subscription API after successful validation of subscriber through Subscriber ID or Auth Token.

**10.2** There are two possible options either the Get subscription API returns subscription details, Complete Bouquet details, complete channel details, or To reduce the size of the response object, the API can be modified to provide only constituent bouquet IDs, Channel IDs , total no. of ala carte channel, total no. of bouquet, total price, subscription type and available balance. If option no. 2 is followed than individual bouquet/channel details can be fetched using, Get Channel and Get Bouquet API.

---

## 11. Get Subscription API

---

**11.1** This API will fetch details of subscribed channels & bouquets currently opted by the subscriber. This will include details of channels/bouquets and their prices, LockIn details and total no. of ala carte channel, total no. of bouquet, total price, subscription type and

available balance. If option no. 2 is followed than individual bouquet/channel details can be fetched using Get Channel and Get Bouquet API.

### 11.2 API Detail: Fetch subscription by Subscription ID

|                       |   |                 |           |           |   |
|-----------------------|---|-----------------|-----------|-----------|---|
| <b>URL</b>            | /subscriber/getSubscription   |                 |           |           |   |
| <b>Method</b>         | GET   |                 |           |           |   |
| <b>Body Parameter</b> | s.no  | Name            | Mandatory | Data Type | Example                                 |
|                       | 1   | subscription_id | Yes       | String    | "123456"                                |
|                       | 2   | Request_type    | Yes       | Integer   | 1/2<br>1: for summary<br>2: for details |
| <b>Authorization</b>  | 3   | Bearer_token    | Yes       | -         | "Bearer_fjhklasj"                       |
| <b>Response</b>       | <pre> {   "status": 200,   "bouquet": [     {       "bouquet_id": 2001,       "bouquet_name": "All South HD Value",       "bouquet_price": 39,       "total_channel": 2,       "lockInExpire": "2019-07-19T08:52:04.344+0000",       "broadcaster": "STAR India Private Limited",       "bouquetchannel": [         {           "channel_id": 1001,           "channel_name": "Zee TV",           "category": "GEC",           "language": "Hindi",           "price": 19,           "sdhd": "SD",           "type": 0, // 0- Broadcaster, 1- VAS           "broadcaster": "Zee Entertainment Enterprises Limited"         },         {           "channel_id": 1002,           "channel_name": "Zee Cinema",           "category": "GEC", </pre> |                 |           |           |   |

|  |   |
|--|---|
|  | <pre> “language”: “Hindi”, “lockInExpire”: “null”, “price”: 19, “sdhd”: “SD”, “type”: 0, // 0- Broadcaster, 1- VAS “broadcaster”: “Zee Entertainment Enterprises Limited” } ] }, { “bouquet_id”: 2002, “bouquet_name”: “All South HD Value 2”, “bouquet_price”: 36, “total_channel”: 2, “lockInExpire”: “2019-07-19T08:52:04.344+0000”, “broadcaster”: “STAR India Private Limited”, “bouquetchannel”: [ { “channel_id”: 1003, “channel_name”: “Star Plus”, “category”: “GEC”, “language”: “Hindi”, “price”: 19, “sdhd”: “SD”, “type”: 0, // 0- Broadcaster, 1- VAS “broadcaster”: “STAR India Private Limited” }, { “channel_id”: 1004, “channel_name”: “Asianet Plus”, “category”: “GEC”, “language”: “Malayalam”, “lockInExpire”: “null”, “price”: 19, “sdhd”: “SD”, “type”: 0, // 0- Broadcaster, 1- VAS “broadcaster”: “STAR India Private Limited” } ] } </pre> |
|--|---|

|  |  |
|--|--|
|  | <pre> ], "channels": [   {     "channel_id": 1001,     "channel_name": "Zee TV",     "category": "GEC",     "language": "Hindi",     "lockInExpire": "2019-07-19T08:52:04.344+0000",     "price": 19,     "sdhd": "SD",     "type": 0, // 0- Broadcaster, 1- VAS     "broadcaster": "Zee Entertainment Enterprises Limited"   }, {     "channel_id": 1002,     "channel_name": "Zee Cinema",     "category": "GEC",     "language": "Hindi",     "lockInExpire": "null",     "price": 19,     "sdhd": "SD",     "type": 0, // 0- Broadcaster, 1- VAS     "broadcaster": "Zee Entertainment Enterprises Limited"   } ], "total_channels": 6, "total_bouquet": 2, "total_alacarte": 2, "amount": 525, "availbalance": 952, "activationDate": "2019-07-19T08:52:04.344+0000" } </pre> |
|--|--|

**11.3** This API will fetch IDs of subscribed channels & bouquets currently opted by the subscriber, total no. of ala carte channel, total no. of bouquet, total price, subscription type and available balance. If option no. 2 is followed, then individual bouquet/channel details can be fetched using Get Channel and Get Bouquet API.

### 11.4 API Detail: Fetch subscription by Subscription ID

|                       |   |                 |           |           |   |
|-----------------------|---|-----------------|-----------|-----------|---|
| <b>URL</b>            | /subscriber/getSubscription   |                 |           |           |   |
| <b>Method</b>         | GET   |                 |           |           |   |
| <b>Form Parameter</b> | s.no  | Name            | Mandatory | Data Type | Example                                 |
|                       | 1   | subscription_id | Yes       | String    | "123456"                                |
|                       | 2   | Request_type    | Yes       | Integer   | 1/2<br>1: for summary<br>2: for details |
| <b>Authorization</b>  | 3   | Bearer_token    | Yes       | -         | "Bearer_fjhklasj"                       |
| <b>Response</b>       | <pre> {   "status": 200,   "bouquet": [     {       "bouquet_id": 2001,       "lockInExpire": "2019-09-30T08:52:04.344+0000"     },     {       "bouquet_id": 2002,       "lockInExpire": "null"     }   ],   "channels": [     {       "channel_id": 1001,       "lockInExpire": "2019-10-19T08:52:04.344+0000"     },     {       "channel_id": 1002,       "lockInExpire": "null"     }   ],   "total_channels": 6,   "total_bouquet": 2,   "total_alacarte": 2,   "amount": 525,   "availbalance": 952,   "activationDate": "2019-07-19T08:52:04.344+0000" } </pre> |                 |           |           |   |

---

## 12. Platform offerings (Channels and Bouquets)

---

**12.1** TRAI App/Portal shall need to fetch the information regarding the Bouquets and Channels offered by the Platform.

**12.2** This information can be fetched in two ways. Either get all the offerings of platform by calling one API i.e. Menu API or fetch channels and bouquet offered through Get Channel and Get Bouquet API.

**12.3** Menu API shall provide information about all the offerings on the platform. The return object size may be bigger as compared to the Channel and Bouquet API called individually.

**12.4** The response to these APIs is not subscriber specific. Thus, to call these APIs subscription id need not be passed as parameter. Keeping security in view user ids & password/key may be used to authenticate. Additionally, DPO may whitelist TRAI server IP.

---

## 13. Menu API

---

**13.1** This API shall be used to fetch all the offerings available on the DPO's platform. The response shall include complete list of offered channels and bouquets and their corresponding details.

### 13.2 API Detail: Fetch Offerings on the Platform

|                      |   |
|----------------------|---|
| <b>URL</b>           | /provider/platformoffering  |
| <b>Method</b>        | GET   |
| <b>Authorization</b> | User id & password/Key may be used to authenticate.   |
| <b>Response</b>      | <pre>{   "status": 200,   "channels": [     {       "channel_id": 1001,       "channel_name": "Zee TV",</pre> |



|  |  |
|--|--|
|  | <pre> “category”: “GEC”, “language”: “Hindi”, “lockInPeriod”: 30, // No. of days “price”: 19, “imageurl”: “Channellogo/logo/ZeeTV.jpg”, “sdhd”: “SD”, “type”: 0, // 0- Broadcaster, 1- VAS “broadcaster”: “Zee Entertainment Enterprises Limited” }, { “channel_id”: 1002, “channel_name”: “Zee Cinema”, “category”: “GEC”, “language”: “Hindi”, “lockInExpire”: “null”, “price”: 19, “imageurl”: “Channellogo/logo/ZeeCinema.jpg”, “sdhd”: “SD”, “type”: 1, // 0- Broadcaster, 1- VAS “broadcaster”: “null” } ], “bouquet”: [ { “bouquet_id”: 2001, “bouquet_name”: “All South HD Value”, “bouquet_price”: 39, “total_channel”: 2, “lockInPeriod”: 5, // No. of days “broadcaster”: “STAR India Private Limited”, “bouquetchannel”: [ { “channel_id”: 1001, “channel_name”: “Zee TV”, “category”: “GEC”, “language”: “Hindi”, “price”: 19, “imageurl”: “Channellogo/logo/ZeeTV.jpg”, “sdhd”: “SD”, </pre> |
|--|--|

|  |  |
|--|--|
|  | <pre> "type": 0, // 0- Broadcaster, 1- VAS "broadcaster": "Zee Entertainment Enterprises Limited" },{ "channel_id": 1002, "channel_name": "Zee Cinema", "category": "GEC", "language": "Hindi", "lockInExpire": "null", "price": 19, "imageurl": "Channellogo/logo/ZeeCinema.jpg", "sdhd": "SD", "type": 0, // 0- Broadcaster, 1- VAS "broadcaster": "Zee Entertainment Enterprises Limited" } ] } ]} </pre> |
|--|--|

---

## 14. Get Channel API

---

**14.1** This API shall be used to fetch list of channels available on the platform of DPO or to fetch the details of channel through its channel ID. Channel ID parameter is optional. When channel ID as parameter is passed then details of channel will be fetched. In case API is called without any parameter then the list of channels available on the platform of DPO will be fetched.

### 14.2 API Detail: Fetch Channel details

|                      |   |            |           |           |         |
|----------------------|---|------------|-----------|-----------|---------|
| <b>URL</b>           | /provider/getChannels                               |            |           |           |         |
| <b>Method</b>        | GET   |            |           |           |         |
| <b>Parameter</b>     | s.no  | Name       | Mandatory | Data Type | Example |
|                      | 1   | Channel_id | No        | String    | 1001    |
| <b>Authorization</b> | User id & password/Key may be used to authenticate. |            |           |           |         |
| <b>Response</b>      | <pre> { "status": 200, "channels": [ </pre>         |            |           |           |         |

|  |   |
|--|---|
|  | <pre> {   "channel_id": 1001,   "channel_name": "Zee TV",   "category": "GEC",   "language": "Hindi",   "lockInPeriod": 2,   "price": 19,   "imageurl": "Channellogo/logo/ZeeTV.jpg",   "sdhd": "SD",   "type": 0, // 0- Broadcaster, 1- VAS   "broadcaster": "Zee Entertainment Enterprises Limited" }, {   "channel_id": 1002,   "channel_name": "Zee Cinema",   "category": "GEC",   "language": "Hindi",   "lockInExpire": "null",   "price": 19,   "imageurl": "Channellogo/logo/ZeeCinema.jpg",   "sdhd": "SD",   "type": 1, // 0- Broadcaster, 1- VAS   "broadcaster": "null" } ] } </pre> |
|--|---|

---

## 15. Get Bouquet API

---

**15.1** This API shall be either used to fetch list of bouquets available on the platform of DPO or to fetch the details of bouquet through its bouquet ID. This API may need bouquet ID as parameter, which is optional. When bouquet ID as parameter is passed then details of bouquet will be fetched. In case API is called without any parameter then the list of bouquets available on the platform of DPO will be fetched.

## 15.2 API Detail: Fetch Bouquet details

|                      |  |            |           |           |         |
|----------------------|--|------------|-----------|-----------|---------|
| <b>URL</b>           | /provider/getBouquets  |            |           |           |         |
| <b>Method</b>        | GET  |            |           |           |         |
| <b>Parameter</b>     | s.no   | Name       | Mandatory | Data Type | Example |
|                      | 1  | Bouquet_id | No        | String    | 2001    |
| <b>Authorization</b> | User id & password/Key may be used to authenticate.  |            |           |           |         |
| <b>Response</b>      | <pre> {   "status": 200,   "bouquet": [     {       "bouquet_id": 2001,       "bouquet_name": "All South HD Value",       "bouquet_price": 39,       "total_channel": 3,       "lockInPeriod": 6,       "broadcaster": "STAR India Private Limited",       "bouquetchannel": [         {           "channel_id": 1001,           "channel_name": "Zee TV",           "category": "GEC",           "language": "Hindi",           "price": 19,           "imageurl": "Channellogo/logo/ZeeTV.jpg",           "sdhd": "SD",           "type": 0 // 0-Broadcaster, 1- VAS           "broadcaster": "Zee Entertainment Enterprises Limited"         }       ]     }   ] } </pre> |            |           |           |         |

---

## 16. Set Subscription

---

**16.1** Subscriber shall be able to modify his/her current subscription. Subscriber may delete some subscribed channels and bouquets or add few. TRAI App/Portal shall submit the changes to the DPO platform. The new subscription can be set either as a full & complete new request or only changes made to the current subscription.

---

## 17. Set Subscription API

---

**17.1** This API shall submit the request for modification in subscription with added/deleted channels & bouquets of the subscribers to their respective DPOs. Upon successfully receiving the request, DPO platform will return acknowledge number.

### 17.2 API Detail: Request (changes) update subscription

|                                 |  |                       |           |           |                                      |
|---------------------------------|--|-----------------------|-----------|-----------|--------------------------------------|
| <b>URL</b>                      | /subscriber/setSubscription  |                       |           |           |                                      |
| <b>Method</b>                   | PUT  |                       |           |           |                                      |
| <b>Body Parameter</b>           | s.no   | Parameter type / Name | Mandatory | Data Type | Example                              |
|                                 | 1  | subscription_id       | Yes       | String    | "12345"                              |
|                                 | 2  | subscription          | Yes       | Object    |                                      |
|                                 | 3  | request_type          | Yes       | Integer   | 1/2<br>1: for changes<br>2: for full |
| <b>Authorization</b>            | 4  | Bearer_token          | Yes       | -         | "Bearer_fj hklasj"                   |
| <b>Request Parameter Object</b> | <pre> {   "subscription_id": "12345",   "bouquet":   {     "added": [       {         "bouquet_id": 2001       }     ],     "deleted": [       {         "bouquet_id": 2001       }     ]   } },   "channels":   {     "added": [       { </pre> |                       |           |           |                                      |

|                 |  |
|-----------------|--|
|                 | <pre> "channel_id": 1001 } ], "deleted": [ { "channel_id": 1001 } ] }, "amount": 355, "type": "monthly", } </pre>  |
| <b>Response</b> | <pre> { "status": 200 "message": "Subscription request submitted", "acknowledgmentNo": "ZHT1344541255423" } </pre> |

### 17.3 API Detail: Request (Full) update subscription

|                                 |  |                       |           |           |                                      |
|---------------------------------|--|-----------------------|-----------|-----------|--------------------------------------|
| <b>URL</b>                      | /subscriber/setSubscription  |                       |           |           |                                      |
| <b>Method</b>                   | PUT  |                       |           |           |                                      |
| <b>Body Parameter</b>           | s.no   | Parameter type / Name | Mandatory | Data Type | Example                              |
|                                 | 1  | subscription_id       | Yes       | String    | "12345"                              |
|                                 | 2  | subscription          | Yes       | Object    |                                      |
|                                 | 3  | request_type          | Yes       | Integer   | 1/2<br>1: for changes<br>2: for full |
| <b>Authorization</b>            | 4  | Bearer_token          | Yes       | -         | "Bearer_fj hklasj"                   |
| <b>Request Parameter Object</b> | <pre> { "subscription_id": 12345, "bouquet": [ { "bouquet_id": 2001 } ] } </pre> |                       |           |           |                                      |

|                 |   |
|-----------------|---|
|                 | <pre> }, {   "bouquet_id": 2002 } ], "channels": [   {     "channel_id": 1001   },   {     "channel_id": 1002   } ], "amount": 355, "type": "monthly", } </pre> |
| <b>Response</b> | <pre> {   "status": 200   "message": "Subscription request submitted",   "acknowledgmentNo": "ZHT1344541255423" } </pre>  |

---

## 18. Subscription Status API

---

**18.1** This API shall fetch the status of subscription request. TRAI App/Portal shall send acknowledgment number of previous request and in response shall receive Subscription ID, if subscription is activated. Otherwise API shall respond the status as Inactive or Rejected.

### 18.2 API Detail: Request subscription status

|               |                                   |      |           |           |         |
|---------------|-----------------------------------|------|-----------|-----------|---------|
| <b>URL</b>    | /subscriber/getSubscriptionStatus |      |           |           |         |
| <b>Method</b> | GET                               |      |           |           |         |
|               | s.no                              | Name | Mandatory | Data Type | Example |

|                       |  |                  |     |        |                            |
|-----------------------|--|------------------|-----|--------|----------------------------|
| <b>Form Parameter</b> | 1  | acknowledgmentNo | Yes | String | "ZHT1344<br>54125542<br>3" |
| <b>Authorization</b>  | 2  | Bearer_token     | Yes | -      | "Bearer_fj<br>hklasj"      |
| <b>Response</b>       | <pre>{   "status": 200,   "subscriptionStatus": "Active" or "Inactive" or "Rejected",   "subscription_id": 12345,   "ActRejDate": "2019-07-19T08:52:04.344+0000" }</pre> |                  |     |        |                            |

---

## 19. Response Code

---

| S. No. | API Response Codes | Response Code Meaning               |
|--------|--------------------|-------------------------------------|
| 1.     | 200                | Success                             |
| 2.     | 400                | Bad Request/Invalid URL             |
| 3.     | 401                | Invalid Subscriber/Mobile no/VC no. |
| 4.     | 402                | Invalid Subscription                |
| 5.     | 404                | Parameter Mismatch                  |
| 6.     | 416                | Invalid Token                       |
| 7.     | 500                | Internal Server Error               |
| 8.     | 501                | Token Expired                       |
| 9.     | 502                | Invalid Channel                     |
| 10.    | 503                | Invalid Bouquet                     |
| 11.    | 505                | Channel/Bouquet in Lock in period   |

DPOs shall develop these APIs in scalable model and during implementation or testing phase of these APIs, if any parameter/authentication mechanism requires changes DPO shall be informed accordingly.

\*\*\*\*\*