

Unica Interact V12.0 Rest API



Contents

- Chapter 1. Unica Interact Rest API Guide..... 1**
- Rest API for Interact Runtime..... 1
 - API - Sessions..... 1
 - API - Offers..... 3
 - API - Events..... 7
 - API - Profiles..... 8
 - API - Audience..... 9
 - API - Batch Execute..... 11
 - API - Deployment..... 16
 - API - Event Pattern State..... 20
 - API - Statistics..... 22
 - API - Interact Status..... 25
- Rest API for Interact Design time..... 25
 - API - Interactive Channel..... 26
 - API - Zone and Interaction Point wrapper..... 34
 - API - Category..... 41
 - API - Event..... 46
 - API - Event Pattern..... 64
 - API - Strategy..... 87
 - API - Smart Rule..... 96
 - API - Strategy Deployment..... 106
 - API - Smart Rule Group..... 108
 - API - Interactive Channel Depolyment..... 114

Chapter 1. Unica Interact Rest API Guide

contains information about Rest API for Unica Interact Runtime and Unica Interact Design Time.

Rest API for Unica Interact Runtime

The following section includes the Rest APIs for Unica Interact Runtime.

API - Sessions

Start Session

These APIs are used to start or end the Interact Runtime session.

Methods

```
POST/interact/v2/sessions/{sessionID}?  
ic={icName}&audienceLevel={audienceLevel}&audien  
celDField={audienceIDField}
```

This API call starts the Interact session.

Parameters

Parameters	Description
ic	Interactive Channel Name
sessionID	Interact Runtime Session ID
audienceLevel	Audience Level
audienceIDField	Audience ID

Body Parameter

This is used to provide the start session command details.

Content-Type: application/json

```
{
  "auIdParams": [
    {
      "n": "string",
      "t": "string",
      "v": ""
    }
  ],
  "debug": true,
  "parameters": [
    {
      "n": "string",
      "t": "string",
      "v": ""
    }
  ],
  "relyOnExistingSession": true
}
```

Example Result

Content-Type: application/json

```
{
  "messages": [],
  "sessionId": "101",
  "version": "1.0.0",
  "statusCode": 0
}
```

```
}

```

End Session

This API call ends and removes the Interact session.

DELETE/interact/v2/sessions/{sessionId}

Parameters

Parameter	Description
sessionId	Interact Runtime Session ID

Example Result

Content-Type: application/json

```
{
  "messages": null,
  "sessionId": "101",
  "version": "1.0.0",
  "statusCode": 0
}
```

API - Offers

Description

These APIs are used to get the offers from the started Interact Runtime session.

Get Offers

This API call gets offers from the started Interact session.

GET/interact/v2/offers/{sessionId}/{interactionPointID}?
number={numberRequested}

Parameters

Parameters	Description
sessionID	Interact Runtime Session ID
interactionPointID	Interaction Point ID
number	Number of offers requested

Example Result

Content-Type: application/json

```
{
  "messages": null,
  "offerLists": [
    {
      "defaultString": "string",
      "ip": "string",
      "offers": [
        {
          "attributes": [
            {
              "n": "string",
              "t": "string",
              "v": "string"
            }
          ],
          "code": [
            "string"
          ],
          "desc": "string",
          "n": "string",
          "score": 0,
          "treatmentCode": "string"
        }
      ]
    }
  ]
}
```

```

"sessionId": "string",
"statusCode": 0,
"version": "string"
}

```

Get Offers for Multiple IPs

This API call gets offers for multiple interaction points.

POST/interact/v2/offers/{sessionId}

Body Parameter

This is used to provide the get offers for multiple IP command details.

Parameters

Parameter	Description
sessionId	Interact Runtime Session ID.

Content-Type: application/json

```

{
  "messages": [],
  "offerLists": [
    {
      "defaultString": "string",
      "ip": "string",
      "offers": [
        {
          "attributes": [
            {
              "n": "string",
              "t": "string",
              "v": {}
            }
          ]
        }
      ]
    }
  ]
}

```

```

    ],
    "code": [
      "string"
    ],
    "desc": "string",
    "n": "string",
    "score": 0,
    "treatmentCode": "string"
  }
]
}
],
"sessionId": "string",
"statusCode": 0,
"version": "string"
}

```

Example Result

Content-Type: application/json

```

{
  "messages": [],
  "offerLists": [
    {
      "defaultString": "string",
      "ip": "string",
      "offers": [
        {
          "attributes": [
            {
              "n": "string",
              "t": "string",

```



```

        "v": {}
      }
    ],
    "code": [
      "string"
    ],
    "desc": "string",
    "n": "string",
    "score": 0,
    "treatmentCode": "string"
  }
]
}
],
"sessionId": "string",
"statusCode": 0,
"version": "string"
}

```

API - Events

Description

These APIs are used to post the event to the started Interact Runtime session.

Post Event

This API call posts event to the Interact session.

POST/interact/v2/events/{sessionId}/{eventName}

Parameters

Parameters	Description
sessionId	Interact Runtime Session ID

Parameters	Description
eventName	Event Name

Body Parameter

This is used to provide the post event command details.

Content-Type: application/json

```
{
  "parameters": [
    {
      "n": "string",
      "t": "string",
      "v": "string"
    }
  ]
}
```

Example Result

Content-Type: application/json

```
{
  "messages": null,
  "sessionId": "101",
  "version": "1.0.0",
  "statusCode": 0
}
```

API - Profiles

Description

These APIs are used to get the profile data from started Interact Runtime Session.

Get Profile

This API call gets profile data from Interact session.

GET/interact/v2/profiles/{sessionID}

Parameters

Parameter	Description
sessionID	Interact Runtime session ID

Example Result

Content-Type: application/json

```
{
  "messages": null,
  "profile": [
    {
      "n": "string",
      "t": "string",
      "v": "string"
    }
  ],
  "sessionId": "string",
  "statusCode": 0,
  "version": "string"
}
```

API - Audience

Description

These APIs are used to set the audience data to the existing Interact runtime session.

Set Audience

This API call sets audience data to the Interact session.

```
PUT/interact/v2/audience/{sessionId}?
```

```
audienceLevel={audienceLevel}&audienceIDField={audienceIDField}
```

Parameters

Parameters	Description
sessionId	Interact Runtime session ID
audienceLevel	Audience Level
audienceIDField	Audience ID

Body Parameter

This is used to provide the post event command details.

Content-Type: application/json

```
{
  "parameters": [
    {
      "t": "string",
      "v": "string",
      "n": "string"
    }
  ]
}
```

Example Result

Content-Type: application/json

```
{
  "statusCode": 0,
  "sessionId": "101",
  "messages": null,
  "version": "1.0.0"
}
```

API - Batch Execute

Description

These APIs are used to execute batch commands.

Methods

POST /v2/batch/{sessionId}

Parameters

Parameters	Description
sessionId	Interact runtime session ID

Body Parameter

This is used to provide the start session command details.

Content-Type: application/json

```
{
  "endSession": true,
  "getOffers": {
    "interactionPointID": "string",
    "numberRequested": 0
  },
  "getOffersForMultipleInteractionPoints": {
    "getOfferRequests": [
```

```
{
  "dupPolicy": 0,
  "ip": "string",
  "numberRequested": 0,
  "offerAttributes": {
    "attributes": [
      {
        "n": "string",
        "t": "string",
        "v": {}
      }
    ],
    "childRequirements": [
      {}
    ],
    "numberRequested": 0
  }
},
"getProfile": true,
"getVersion": true,
"postEvent": {
  "event": "string",
  "parameters": [
    {
      "n": "string",
      "t": "string",
      "v": {}
    }
  ]
},
```

```
"setAudience": {
  "audienceID": [
    {
      "n": "string",
      "t": "string",
      "v": {}
    }
  ],
  "audienceLevel": "string",
  "parameters": [
    {
      "n": "string",
      "t": "string",
      "v": {}
    }
  ]
},
"setDebug": {
  "debug": true
},
"startSession": {
  "auIdParams": [
    {
      "n": "string",
      "t": "string",
      "v": {}
    }
  ],
  "audienceID": [
    {
      "n": "string",
      "t": "string",
```

```

    "v": {}
  }
],
"audienceLevel": "string",
"debug": true,
"ic": "string",
"parameters": [
  {
    "n": "string",
    "t": "string",
    "v": {}
  }
],
"relyOnExistingSession": true
}
}

```

Example Result

Content-Type: application/json

```

{
  "batchStatusCode": 0,
  "responses": [
    {
      "messages": [
        {
          "detailMsg": "string",
          "msg": "string",
          "msgCode": 0,
          "msgLevel": 0
        }
      ]
    }
  ],

```



```
"offerLists": [  
  {  
    "defaultString": "string",  
    "ip": "string",  
    "offers": [  
      {  
        "attributes": [  
          {  
            "n": "string",  
            "t": "string",  
            "v": {}  
          }  
        ],  
        "code": [  
          "string"  
        ],  
        "desc": "string",  
        "n": "string",  
        "score": 0,  
        "treatmentCode": "string"  
      }  
    ]  
  }  
],  
"profile": [  
  {  
    "n": "string",  
    "t": "string",  
    "v": {}  
  }  
],  
"sessionId": "string",
```

```

    "statusCode": 0,
    "version": "string"
  }
]
}

```

API - Deployment

Description

These APIs are used to get the deployment information of interactive channels.

Get Deployed Channel

This API call gets the currently active deployment information of an interactive channel.

GET/interact/v2/deployments/channels/{icldOrName}

Parameters

Parameter	Description
icldOrName	Interactive Channel ID or Name

Example Result

Content-Type: application/json

```

{
  "deployment": {
    "categoriesAndEvents": [
      {
        "categoryName": "string",
        "events": [

```

```

        {
            "eventActions": "string",
            "eventName": "string"
        }
    ]
}
],
"constraints": [
    {
        "constraintId": 0,
        "constraintName": "string",
        "offerConstraint": "string"
    }
],
"flowcharts": [
    {
        "audienceLevel": "string",
        "flowchartName": "string"
    }
],
"icName": "string",
"lastDeploymentTime": "string",
"offerThreshold": 0,
"strategies": [
    {
        "ruleGroups": [
            {
                "cellCode": "string",
                "rules": [
                    {
                        "marketWeight": 0,
                        "offerAttributes": [

```

```

        "string"
    ],
    "offerCodes": "string",
    "offerName": "string",
    "predicate": "string",
    "predicateState": "string",
    "ruleEnabled": "string",
    "zoneName": "string"
    }
]
}
],
"strategyName": "string"
}
],
"tableMapping": [
{
    "audienceLevel": "string",
    "tableInfo": [
        {
            "parentTableName": "string",
            "preLoaded": true,
            "tableName": "string"
        }
    ]
}
],
"zonesAndInteractionPoints": [
{
    "interactionPoints": [
        "string"
    ],

```

```

        "zoneName": "string"
      }
    ]
  }
}

```

Get Deployed Offers

This API call gets the deployment information of all currently deployed offers.

GET/interact/v2/deployments/offers

Example Result

Content-Type: application/json

```

{
  "offers": [
    {
      "constraintIds": [
        0
      ],
      "offerCode": [
        "string"
      ],
      "offerId": 0,
      "offerName": "offer",
      "suppressionRules": {
        "categoryName": "string",
        "eventType": "string",
        "hidden": true,
        "id": 0,
        "name": "string"
      }
    }
  ]
}

```

```
]
}
```

API - Event Pattern State

Description

These APIs are used to get the states of all event patterns and offer suppression rules belonging to the specified audience ID.

Get Event Pattern State from Cache

This API call gets the states of all event patterns and offer suppression rules belonging to the specified audience ID.

GET/interact/v2/eventpatterns/cache?

audienceLevel={audienceLevel}&audienceIDField={audienceIDFieldValue}

Parameters

Parameters	Description
audienceLevel	Audience Level
audienceIDField	Audience ID

Example Result

Content-Type: application/json

```
{
  "lastUpdateTime": 0,
  "patternStates": [
    {
      "offerId": "string",
      "offerName": "string",
      "patternId": 0,
      "patternName": "string",
```

```

    "suppression": "string",
    "values": [
      {
        "activationTime": 0,
        "attrArray": {},
        "changed": true,
        "eventCounterDeltas": [ 0 ],
        "initialEventCounters": [ 0 ],
        "lastResetTime": 0,
        "lastUpdateTime": 0,
        "status": "string"
      } ]
    } ]
  }

```

Get Event Pattern State from Database

GET/interact/v2/eventpatterns/db?

audienceLevel={audienceLevel}&audienceIDField={audienceIDFieldValue}

This API call gets the states of all event patterns and offer suppression rules belonging to the specified audience ID.

Parameters

Parameters	Description
audienceLevel	Audience Level
audienceIDField	Audience ID field

Example Result

Content-Type: application/json

```

{
  "lastUpdateTime": 0,

```

```

"patternStates": [
  {
    "offerId": "string",
    "offerName": "string",
    "patternId": 0,
    "patternName": "string",
    "suppression": "string",
    "values": [
      {
        "activationTime": 0,
        "attrArray": {},
        "changed": true,
        "eventCounterDeltas": [
          0
        ],
        "initialEventCounters": [
          0
        ],
        "lastResetTime": 0,
        "lastUpdateTime": 0,
        "status": "string"
      } ] }
] }

```

API - Statistics

Description

These APIs are used to get the JMX sweep.

Get JMX Statistics

This API call gets the JMX sweep.

GET/interact/v2/stats/jmx

Example Result

Content-Type: application/json

```
"statsJMX": [  
  
  {  
  
    "name":  
    "com.unicacorp.interact:type=api,group=Statistics",  
  
    "information": {  
  
      "WaitForFlowchartAverage": "1.0",  
  
      "LoadScoreOverrideDuration": "0",  
  
      "EndSessionCount": "0",  
  
      "SetAudienceCount": "0",  
  
      "GetProfileAverage": "0.0",  
  
      "PostEventAverage": "393.0",  
  
      "ActiveSessionCount": "4",  
  
      "PostEventCount": "1",  
  
      "ExecuteBatchAverage": "184.33333",  
  
      "ExecuteBatchCount": "6",
```

```
"SetDebugAverage": "0.0",

"StartSessionAverage": "131.0",

"GetProfileCount": "0",

"SetDebugCount": "0",

"SetAudienceAverage": "0.0",

"LoadOfferSuppressionDuration": "0",

"GetOffersCount": "1",

"LoadOffersBySQLCount": "0",

"LoadOffersBySQLAverage": "0.0",

"GetVersionCount": "0",

"EndSessionAverage": "0.0",

"StartSessionCount": "4",

"RunSegmentationDuration": "0",

"GetVersionAverage": "0.0",

"GetOffersAverage": "133.0"

}
```

```
    }  
  
  ]  
  
}
```

API - Interact Status

Description

These APIs are used to check status of Interact.

Get Version

This API call gets the version of Interact service.

GET/interact/V2/status

Example Result

Content-Type: application/json

```
{  
  "messages": null,  
  "sessionId": null,  
  "version": "1.0.0"  
  "statusCode": 0  
}
```

Rest API for Interact Design time

The following section includes the Rest APIs for Interact Design time.

API - Interactive Channel

Description

These APIs are used to perform CRUD operations on Interactive Channel.

Get All

GET /Campaign/api/interact/rest/v2/channels?page=0&size=10

Parameters

Parameters	Default Values
page	0
size	10

Output

Table 1. Response Codes

Success	200	Ok
Error	400	Bad Request
Server Error	500	Internal/Server Error

Example: Content-Type: application/json

```
{
  "content": [
    {
      "createBy": 0,
      "createDate": 0,
      "description": "string",
      "id": 0,
      "lastDeploymentTime": 0,
      "longestWaitTime": "string",
      "longestWaitTimeAvailable": true,
      "name": "string",
    }
  ]
}
```

```

    "offerThreshold": 0,
    "policyId": 0,
    "productionServerGroup": "string",
    "serverGroups": [
    "string"
    ],
    "updateBy": 0,
    "updateDate": 0,
    "waitingForDeployment": 0
  }
],
"page": {
  "hasNext": true,
  "hasPrev": true,
  "pageNumber": 0,
  "size": 0,
  "totalElements": 0,
  "totalPages": 0
}
}

```

Get

GET /Campaign/api/interact/rest/v2/channels/{icldOrName}

Output**Table 2. Response Codes**

Success	200	Ok
Error	400	Bad Request
Server Error	500	Internal/Server Error

Example: Content-Type: application/json

```
{
  "createBy": 0,
  "createDate": 0,
  "description": "string",
  "id": 0,
  "lastDeploymentTime": 0,
  "longestWaitTime": "string",
  "longestWaitTimeAvailable": true,
  "name": "string",
  "offerThreshold": 0,
  "policyId": 0,
  "productionServerGroup": "string",
  "serverGroups": [
    "string"
  ],
  "updateBy": 0,
  "updateDate": 0,
  "waitingForDeployment": 0
}
```

Create

POST /Campaign/api/interact/rest/v2/channels

Input

Body Parameter – Content-Type: application/json

```
{
  "channels": [
    {
      "createBy": 0,
      "createDate": 0,
      "description": "string",
```

```

    "id": 0,
    "lastDeploymentTime": 0,
    "longestWaitTime": "string",
    "longestWaitTimeAvailable": true,
    "name": "string",
    "offerThreshold": 0,
    "policyId": 0,
    "productionServerGroup": "string",
    "serverGroups": [
      "string"
    ],
    "updateBy": 0,
    "updateDate": 0,
    "waitingForDeployment": 0
  }
]
}

```

Output

Table 3. Response Codes

Success	201	Created
Miscellaneous	207	Status code of batch request
Error	400	Bad Request
Server Error	500	Internal/Server Error

Example: Content-Type: application/json

```
[
```

```

    {
      "headers": {},
      "body": <entity>/<error-msg>,
      "status": 201/400/500
    }
  ]

```

Update

PUT /Campaign/api/interact/rest/v2/channels/{icldOrName}

Input

Body Parameter – Content-Type: application/json

```

{
  "createBy": 0,
  "createDate": 0,
  "description": "string",
  "id": 0,
  "lastDeploymentTime": 0,
  "longestWaitTime": "string",
  "longestWaitTimeAvailable": true,
  "name": "string",
  "offerThreshold": 0,
  "policyId": 0,
  "productionServerGroup": "string",
  "serverGroups": [
    "string"
  ],
  "updateBy": 0,
  "updateDate": 0,
  "waitingForDeployment": 0
}

```


Output

Table 4. Response Codes

Success	202	Accepted
Error	400	Bad Request
Server Error	500	Internal/Server Error

Example: Content-Type: application/json

```
{
  "createBy": 0,
  "createDate": 0,
  "description": "string",
  "id": 0,
  "lastDeploymentTime": 0,
  "longestWaitTime": "string",
  "longestWaitTimeAvailable": true,
  "name": "string",
  "offerThreshold": 0,
  "policyId": 0,
  "productionServerGroup": "string",
  "serverGroups": ["string"],
  "updateBy": 0,
  "updateDate": 0,
  "waitingForDeployment": 0
}
```

Delete

DELETE /Campaign/api/interact/rest/v2/channels/{icldOrName}

Output

Table 5. Response Codes

Success	204	No Content
Error	400	Bad Request
Server Error	500	Internal/Server Error

Get All Audience Levels

GET /Campaign/api/interact/rest/v2/channels/{icldOrName}/audiencelevels

Output**Table 6. Response Codes**

Success	200	Ok
Error	400	Bad Request
Server Error	500	Internal/Server Error

Example: Content-Type: application/json

```

{
  "audienceLevels": [
    "Account",
    "Customer"
  ]
}

```

Map profile tables to IC

POST /Campaign/api/interact/rest/v2/channels/{icldOrName}/profiletables

Input

Body Parameter – Content-Type: application/json

```

{
  "audienceLevel": "string",
  "baseKeys": [
    "string"
  ]
}

```

```

],
"baseTable": true,
"dimKeys": [
  "string"
],
"joinTable": "string",
"joinType": 0,
"preLoad": true,
"tableName": "string",
"unmap": true
}

```

Table 7. Response Codes

Success	201	Created
Error	400	Bad Request
Server Error	500	Internal/Server Error

Example Request**Map Base Table**

```

{
  "audienceLevel": "Customer",
  "baseTable": true,
  "tableName": "Customer_Audience"
}

```

Map Dimension Table

```

{
  "audienceLevel": "Customer",
  "baseKeys": [
    "CID"
  ]
}

```

```

],
"baseTable": false,
"dimKeys": [
  "CID"
],
"joinTable": "Customer_Audience",
"joinType": 1,
"preLoad": false,
"tableName": "Credit_Details"
}

```

Un-map Table

```

{
  "audienceLevel": "Customer",
  "tableName": "Credit_Details",
  "unmap": true
}

```

API – Zone and Interaction Point wrapper

Description

These APIs are used to perform CRUD operations on zones and interaction points.

Get All

GET /Campaign/api/interact/rest/v2/channels/{icldOrName}/zones?
page=0&size=10

Parameters

Parameters	Default value
page	0

Parameters	Default value
size	10

Output

Table 8. Response Codes

Success	200	Ok
Error	400	Bad Request
Server Error	500	Internal/Server Error

Example: Content-Type: application/json

```
{
  "content": [
    { "description": "string",
      "id": 0,
      "interactionPoints": [
        { "defaultString": "string",
          "description": "string",
          "id": 0, "name": "string"
        }
      ],
      "learningMode": 0,
      "learningModelID": 0,
      "name": "string",
      "percentRandom": 0,
      "ruleGroupResolutionType": 0
    }
  ],
  "page": {
    "hasNext": true,
    "hasPrev": true,
  }
}
```

```

    "pageNumber": 0,
    "size": 0,
    "totalElements": 0,
    "totalPages": 0
  }
}

```

Get

GET /Campaign/api/interact/rest/v2/channels/{icldOrName}/zones/
{zoneIdOrName}

Output**Table 9. Response Codes**

Success	200	Ok
Error	400	Bad Request
Server Error	500	Internal/Server Error

Example: Content-Type: application/json

```

{
  "description": "string",
  "id": 0,
  "interactionPoints": [
    {
      "defaultString": "string",
      "description": "string",
      "id": 0,
      "name": "string"
    }
  ],
  "learningMode": 0,
  "learningModelID": 0,

```

```

"name": "string",
"percentRandom": 0,
"ruleGroupResolutionType": 0
}

```

Create

POST /Campaign/api/interact/rest/v2/channels/{icldOrName}/zones

Input

Body Parameter – Content-Type: application/json

```

{
  "zones": [
    {
      "description": "string",
      "id": 0,
      "interactionPoints": [
        {
          "defaultString": "string",
          "description": "string",
          "id": 0,
          "name": "string"
        }
      ],
      "learningMode": 0,
      "learningModelID": 0,
      "name": "string",
      "percentRandom": 0,
      "ruleGroupResolutionType": 0
    }
  ]
}

```

Output

Table 10. Response Codes

Success	201	Created
Miscellaneous	207	Status code for batch request
Error	400	Bad request
Server Error	500	Internal server error

Example: Content-Type: application/json

```
[
  {
    "headers": {},
    "body": <entity>/<error-msg>,
    "status": 201/400/500
  }
]
```

Update

PUT /Campaign/api/interact/rest/v2/channels/{icldOrName}/zones/
{zoneName}

Input

Body Parameter – Content-Type: application/json

```
{
  "description": "string",
  "id": 0,
  "interactionPoints": [
    {
      "defaultString": "string",
```



```

    "description": "string",
    "id": 0,
    "name": "string"
  }
],
"learningMode": 0,
"learningModelID": 0,
"name": "string",
"percentRandom": 0,
"ruleGroupResolutionType": 0
}

```

Output

Table 11. Response Codes

Success	202	Accepted
Error	400	Bad Request
Server Error	500	Internal/Server Error

Example: Content-Type: application/json

```

{
  "description": "string",
  "id": 0,
  "interactionPoints": [
    {
      "defaultString": "string",
      "description": "string",
      "id": 0,
      "name": "string"
    }
  ],

```

```

"learningMode": 0,
"learningModelID": 0,
"name": "string",
"percentRandom": 0,
"ruleGroupResolutionType": 0
}

```

Delete

DELETE /Campaign/api/interact/rest/v2/channels/{icldOrName}/zones/
{zoneName}

Input

Query	Parameters
onlyIP: true/false	Delete only IPs from zone

Body Parameter – Content-Type: application/json

Input IDs

```

{"ids":[0]}

```

Output

Table 12. Response Codes

Success	204	No Content
Error	400	Bad Request
Server Error	500	Internal/Server Error

Delete Multiple

DELETE /Campaign/api/interact/rest/v2/channels/{icldOrName}/zones

Input

Body Parameter – Content-Type: application/json

```
{"ids":[0]}
```

Output

Table 13. Response Codes

Success	204	No Content
Error	400	Bad Request
Server Error	500	Internal/Server Error

API - Category

Description

These APIs are used to perform CRUD operations on Categories.

Get All

GET /Campaign/api/interact/rest/v2/channels/{icldOrName}/categories?
page=0&size=10

Parameters	Values
page	0
size	10

Output

Table 14. Response Codes

Success	200	Ok
Error	400	Bad Request
Server Error	500	Internal/Server Error

Example: Content-Type: application/json

```

{
  "content": [
    { "description": "string",
      "id": 0,
      "name": "string",
      "systemDefined": true
    }
  ],
  "page": {
    "hasNext": true,
    "hasPrev": true,
    "pageNumber": 0,
    "size": 0,
    "totalElements": 0,
    "totalPages": 0
  }
}

```

Get

GET /Campaign/api/interact/rest/v2/channels/{icldOrName}/categories/
{categoryIdOrName}

Output**Table 15. Response Codes**

Success	200	Ok
Error	400	Bad Request
Server Error	500	Internal/Server Error

Example: Content-Type: application/json

```
{
  "description": "string",
  "id": 0,
  "name": "string",
  "systemDefined": true
}
```

Create

POST /Campaign/api/interact/rest/v2/channels/{icldOrName}/categories

Input

Body Parameter – Content-Type: application/json

```
{
  "categories": [
    {
      "description": "string",
      "id": 0,
      "name": "string",
      "systemDefined": true
    }
  ]
}
```

Output

Table 16. Response Codes

Success	201	Created
Miscellaneous	207	Status code for batch request
Error	400	Bad Request

Server Error	500	Internal/Server Error
--------------	-----	-----------------------

Example: Content-Type: application/json

```
[
  {
    "headers": {},
    "body": <entity>/<error-msg>,
    "status": 201/400/500
  }
]
```

Update

PUT /Campaign/api/interact/rest/v2/channels/{icldOrName}/categories/
{categoryIdOrName}

Input

Body Parameter – Content-Type: application/json

```
{
  "description": "string",
  "id": 0,
  "name": "string",
  "systemDefined": true
}
```

Output

Table 17. Response Codes

Success	202	Accepted
Error	400	Bad Request
Server Error	500	Internal/Server Error

Example: Content-Type: application/json

```
{
  "description": "string",
  "id": 0,
  "name": "string",
  "systemDefined": true
}
```

Delete

DELETE /Campaign/api/interact/rest/v2/channels/{icldOrName}/categories/
{categoryIdOrName}

Output**Table 18. Response Codes**

Success	204	No Content
Error	400	Bad Request
Server Error	500	Internal/Server Error

Delete Multiple

DELETE /Campaign/api/interact/rest/v2/channels/{icldOrName}/categories

Input

Body Parameter – Content-Type: application/json

```
{"ids": [0]}
```

Output**Table 19. Response Codes**

Success	204	No Content
Error	400	Bad Request

Server Error	500	Internal/Server Error
--------------	-----	-----------------------

API - Event

Description

These APIs are used to perform CRUD operations on Events.

Get All

GET /Campaign/api/interact/rest/v2/channels/{icldOrName}/events?
page=0&size=10

Parameters

Parameters	Default Values
page	0
size	10

Output

Table 20. Response Codes

Success	200	Ok
Error	400	Bad Request
Server Error	500	Internal/Server Error

Example: Content-Type: application/json

```
{
  "content": [
    {
      "category": {
        "description": "string",
        "id": 0,
```



```

    "name": "string",
    "systemDefined": true
  },
  "createBy": 0,
  "createDate": 0,
  "description": "string",
  "eventActions": [ {
    "actionID": 0,
    "actionValue": 0,
    "async": true,
    "condition": [ {
      "id": 0,
      "name": "string",
      "value": "string"
    } ],
    "details": "string",
    "eventActionID": 0,
    "eventType": 0,
    "executionOrder": 0,
    "inferredEvents": [0]
  } ],
  "eventType": 0,
  "executeExpression": {
    "conditions": [ {
      "id": 0,
      "name": "string",
      "value": "string"
    } ],
    "enabled": true,
    "eventActionID": 0
  },
  "expression": {

```

```
"audienceLevel": "string",
"expression": "string",
"fieldName": "string"
},
"id": 0,
"logOfferAcceptance": {
  "conditions": [ {
    "id": 0,
    "name": "string",
    "value": "string"
  } ],
  "enabled": true,
  "eventActionID": 0
},
"logOfferContact": {
  "conditions": [ {
    "id": 0,
    "name": "string",
    "value": "string"
  } ],
  "enabled": true,
  "eventActionID": 0
},
"logOfferRejection": {
  "conditions": [ {
    "id": 0,
    "name": "string",
    "value": "string"
  } ],
  "enabled": true,
  "eventActionID": 0
},
```

```

"name": "string",
"pattern": null,
"triggerResegmentation": {
  "conditions": [ {
    "id": 0,
    "name": "string",
    "value": "string"
  } ],
  "enabled": true,
  "eventActionID": 0
},
"updateBy": 0,
"updateDate": 0
}
],
"page": {
  "hasNext": true,
  "hasPrev": true,
  "pageNumber": 0,
  "size": 0,
  "totalElements": 0,
  "totalPages": 0
}
}

```

Get

GET /Campaign/api/interact/rest/v2/channels/{icldOrName}/events/
{eventName}

Output**Table 21. Response Codes**

Success	200	Ok
---------	-----	----

Error	400	Bad Request
Server Error	500	Internal/Server Error

Example: Content-Type: application/json

```
{
  "category": {
    "description": "string",
    "id": 0,
    "name": "string",
    "systemDefined": true
  },
  "createBy": 0,
  "createDate": 0,
  "description": "string",
  "eventActions": [
    {
      "actionID": 0,
      "actionValue": 0,
      "async": true,
      "condition": [
        {
          "id": 0,
          "name": "string",
          "value": "string"
        }
      ],
      "details": "string",
      "eventActionID": 0,
      "eventType": 0,
      "executionOrder": 0,
      "inferredEvents": [0]
```

```
    }
  ],
  "eventType": 0,
  "executeExpression": {
    "conditions": [
      {
        "id": 0,
        "name": "string",
        "value": "string"
      }
    ],
    "enabled": true,
    "eventActionID": 0
  },
  "expression": {
    "audienceLevel": "string",
    "expression": "string",
    "fieldName": "string"
  },
  "id": 0,
  "logOfferAcceptance": {
    "conditions": [
      {
        "id": 0,
        "name": "string",
        "value": "string"
      }
    ],
    "enabled": true,
    "eventActionID": 0
  },
  "logOfferContact": {
```

```
"conditions": [  
  {  
    "id": 0,  
    "name": "string",  
    "value": "string"  
  }  
],  
"enabled": true,  
"eventActionID": 0  
},  
"logOfferRejection": {  
  "conditions": [  
    {  
      "id": 0,  
      "name": "string",  
      "value": "string"  
    }  
  ],  
  "enabled": true,  
  "eventActionID": 0  
},  
"name": "string",  
"pattern": null,  
"triggerResegmentation": {  
  "conditions": [  
    {  
      "id": 0,  
      "name": "string",  
      "value": "string"  
    }  
  ],  
  "enabled": true,
```

```

    "eventActionID": 0
  },
  "updateBy": 0,
  "updateDate": 0
}

```

Create

POST /Campaign/api/interact/rest/v2/{icldOrName}/events

Input

Body Parameter – Content-Type: application/json

```

{
  "events": [
    {
      "category": {
        "description": "string",
        "id": 0,
        "name": "string",
        "systemDefined": true
      },
      "createBy": 0,
      "createDate": 0,
      "description": "string",
      "eventActions": [
        {
          "actionID": 0,
          "actionValue": 0,
          "async": true,
          "condition": [
            {
              "id": 0,
              "name": "string",

```

```

        "value": "string"
    }
],
"details": "string",
"eventActionID": 0,
"eventType": 0,
"executionOrder": 0,
"inferredEvents": [0]
}
],
"eventType": 0,
"executeExpression": {
    "conditions": [
        {
            "id": 0,
            "name": "string",
            "value": "string"
        }
    ],
    "enabled": true,
    "eventActionID": 0
},
"expression": {
    "audienceLevel": "string",
    "expression": "string",
    "fieldName": "string"
},
"id": 0,
"logOfferAcceptance": {
    "conditions": [
        {
            "id": 0,

```



```

        "name": "string",
        "value": "string"
    }
],
"enabled": true,
"eventActionID": 0
},
"logOfferContact": {
    "conditions": [
        {
            "id": 0,
            "name": "string",
            "value": "string"
        }
    ],
    "enabled": true,
    "eventActionID": 0
},
"logOfferRejection": {
    "conditions": [
        {
            "id": 0,
            "name": "string",
            "value": "string"
        }
    ],
    "enabled": true,
    "eventActionID": 0
},
"name": "string",
"pattern": null,
"triggerResegmentation": {

```

```

    "conditions": [
      {
        "id": 0,
        "name": "string",
        "value": "string"
      }
    ],
    "enabled": true,
    "eventActionID": 0
  },
  "updateBy": 0,
  "updateDate": 0
}
]
}

```

Output

Table 22. Response Codes

Success	201	Created
Miscellaneous	207	Status code for batch request
Error	400	Bad Request
Server Error	500	Internal/Server Error

Example: Content-Type: application/json

```

[
  {
    "headers": {},
    "body": <entity>/<error-msg>,
    "status": 201/400/500
  }
]

```

```

    }
  ]
}

```

Update

PUT /Campaign/api/interact/rest/v2/channels/{icldOrName}/events/
{eventIdOrName}

Input

Body Parameter – Content-Type: application/json

```

{
  "category": {
    "description": "string",
    "id": 0,
    "name": "string",
    "systemDefined": true
  },
  "createBy": 0,
  "createDate": 0,
  "description": "string",
  "eventActions": [
    {
      "actionID": 0,
      "actionValue": 0,
      "async": true,
      "condition": [
        {
          "id": 0,
          "name": "string",
          "value": "string"
        }
      ],
      "details": "string",

```

```

    "eventActionID": 0,
    "eventType": 0,
    "executionOrder": 0,
    "inferredEvents": [
      0
    ]
  }
],
"eventType": 0,
"executeExpression": {
  "conditions": [
    {
      "id": 0,
      "name": "string",
      "value": "string"
    }
  ],
  "enabled": true,
  "eventActionID": 0
},
"expression": {
  "audienceLevel": "string",
  "expression": "string",
  "fieldName": "string"
},
"id": 0,
"logOfferAcceptance": {
  "conditions": [
    {
      "id": 0,
      "name": "string",
      "value": "string"
    }
  ]
}

```

```

    }
  ],
  "enabled": true,
  "eventActionID": 0
},
"logOfferContact": {
  "conditions": [
    {
      "id": 0,
      "name": "string",
      "value": "string"
    }
  ],
  "enabled": true,
  "eventActionID": 0
},
"logOfferRejection": {
  "conditions": [
    {
      "id": 0,
      "name": "string",
      "value": "string"
    }
  ],
  "enabled": true,
  "eventActionID": 0
},
"name": "string",
"pattern": null,
"triggerResegmentation": {
  "conditions": [
    {

```

```

        "id": 0,
        "name": "string",
        "value": "string"
    }
],
"enabled": true,
"eventActionID": 0
},
"updateBy": 0,
"updateDate": 0
}

```

Output

Table 23. Response Codes

Success	202	Accepted
Error	400	Bad Request
Server Error	500	Internal/Server Error

Example: Content-Type: application/json

```

{
  "category": {
    "description": "string",
    "id": 0,
    "name": "string",
    "systemDefined": true
  },
  "createBy": 0,
  "createDate": 0,
  "description": "string",
  "eventActions": [

```

```
{
  "actionID": 0,
  "actionValue": 0,
  "async": true,
  "condition": [
    {
      "id": 0,
      "name": "string",
      "value": "string"
    }
  ],
  "details": "string",
  "eventActionID": 0,
  "eventType": 0,
  "executionOrder": 0,
  "inferredEvents": [
    0
  ]
},
"eventType": 0,
"executeExpression": {
  "conditions": [
    {
      "id": 0,
      "name": "string",
      "value": "string"
    }
  ],
  "enabled": true,
  "eventActionID": 0
},
```

```
"expression": {
  "audienceLevel": "string",
  "expression": "string",
  "fieldName": "string"
},
"id": 0,
"logOfferAcceptance": {
  "conditions": [
    {
      "id": 0,
      "name": "string",
      "value": "string"
    }
  ],
  "enabled": true,
  "eventActionID": 0
},
"logOfferContact": {
  "conditions": [
    {
      "id": 0,
      "name": "string",
      "value": "string"
    }
  ],
  "enabled": true,
  "eventActionID": 0
},
"logOfferRejection": {
  "conditions": [
    {
      "id": 0,
```



```

        "name": "string",
        "value": "string"
    }
],
"enabled": true,
"eventActionID": 0
},
"name": "string",
"pattern": null,
"triggerResegmentation": {
    "conditions": [
        {
            "id": 0,
            "name": "string",
            "value": "string"
        }
    ],
    "enabled": true,
    "eventActionID": 0
},
"updateBy": 0,
"updateDate": 0
}

```

Delete

DELETE /Campaign/api/interact/rest/v2/channels/{icldOrName}/events/
{eventIdOrName}

Output

Table 24. Response Codes

Success	204	No Content
Error	400	Bad Request

Server Error	500	Internal/Server Error
--------------	-----	-----------------------

Delete Multiple

DELETE /Campaign/api/interact/rest/v2/channels/{icldOrName}/events

Input

Body Parameter – Content-Type: application/json

```
{"ids": [0,0]}
```

Output

Table 25. Response Codes

Success	204	No Content
Error	400	Bad Request
Server Error	500	Internal/Server Error

API - Event Pattern

Description

These APIs are used to perform CRUD operations on Event Patterns.

Get All

GET /Campaign/api/interact/rest/v2/channels/{icldOrName}/eventpatterns?
page=0&size=10

Parameters

Parameters	Values
page	0
size	10

Output

Table 26. Response Codes

Success	200	Ok
Error	400	Bad Request
Server Error	500	Internal/Server Error

Example: Content-Type: application/json

```
{
  "content": [
    {
      "category": {
        "description": "string",
        "id": 0,
        "name": "string",
        "systemDefined": true
      },
      "createBy": 0,
      "createDate": 0,
      "description": "string",
      "eventActions": [ {
        "actionID": 0,
        "actionValue": 0,
        "async": true,
        "condition": [ {
          "id": 0,
          "name": "string",
          "value": "string"
        } ],
        "details": "string",
        "eventActionID": 0,
        "eventType": 0,

```

```
"executionOrder": 0,
"inferredEvents": [ 0 ] } ],
"eventType": 0,
"executeExpression": {
  "conditions": [ {
    "id": 0,
    "name": "string",
    "value": "string"
  } ],
  "enabled": true,
  "eventActionID": 0
},
"expression": {
  "audienceLevel": "string",
  "expression": "string",
  "fieldName": "string"
},
"id": 0,
"logOfferAcceptance": {
  "conditions": [ {
    "id": 0,
    "name": "string",
    "value": "string"
  } ],
  "enabled": true,
  "eventActionID": 0
},
"logOfferContact": {
  "conditions": [ {
    "id": 0,
    "name": "string",
    "value": "string"
```

```

    } ],
    "enabled": true,
    "eventActionID": 0
  },
  "logOfferRejection": {
    "conditions": [ {
      "id": 0,
      "name": "string",
      "value": "string"
    } ],
    "enabled": true,
    "eventActionID": 0
  },
  "name": "string",
  "pattern": {
    "async": true,
    "enabled": true,
    "endTime": 0,
    "itemEvent": [ {
      "attribute": {
        "dataType": 0,
        "name": "string",
        "value": "string"
      },
      "id": 0,
      "name": "string",
      "value": 0
    } ],
    "matchTotal": 0,
    "reset": {
      "id": 0,
      "idSelected": 0,

```

```
    "idToRemove": 0,
    "interval": 0,
    "intervalType": "string",
    "startTime": 0,
    "type": 0
  },
  "resetDuration": 0,
  "startTime": 0,
  "timeDuration": 0,
  "type": 0
},
"triggerResegmentation": {
  "conditions": [ {
    "id": 0,
    "name": "string",
    "value": "string"
  } ],
  "enabled": true,
  "eventActionID": 0
},
"updateBy": 0,
"updateDate": 0
}
],
"page": {
  "hasNext": true,
  "hasPrev": true,
  "pageNumber": 0,
  "size": 0,
  "totalElements": 0,
  "totalPages": 0
}
```

```
}

```

Get

GET /Campaign/api/interact/rest/v2/channels/{icldOrName}/eventpatterns/
{ eventpatternsIdOrName}

Output**Table 27. Response Codes**

Success	200	Ok
Error	400	Bad Request
Server Error	500	Internal/Server Error

Example: Content-Type: application/json

```
{
  "category": {
    "description": "string",
    "id": 0,
    "name": "string",
    "systemDefined": true
  },
  "createBy": 0,
  "createDate": 0,
  "description": "string",
  "eventActions": [
    {
      "actionID": 0,
      "actionValue": 0,
      "async": true,
      "condition": [
        {
          "id": 0,

```

```

        "name": "string",
        "value": "string"
    }
],
"details": "string",
"eventActionID": 0,
"eventType": 0,
"executionOrder": 0,
"inferredEvents": [
    0
]
}
],
"eventType": 0,
"executeExpression": {
    "conditions": [
        {
            "id": 0,
            "name": "string",
            "value": "string"
        }
    ],
    "enabled": true,
    "eventActionID": 0
},
"expression": {
    "audienceLevel": "string",
    "expression": "string",
    "fieldName": "string"
},
"id": 0,
"logOfferAcceptance": {

```



```
"conditions": [  
  {  
    "id": 0,  
    "name": "string",  
    "value": "string"  
  }  
],  
"enabled": true,  
"eventActionID": 0  
},  
"logOfferContact": {  
  "conditions": [  
    {  
      "id": 0,  
      "name": "string",  
      "value": "string"  
    }  
  ],  
  "enabled": true,  
  "eventActionID": 0  
},  
"logOfferRejection": {  
  "conditions": [  
    {  
      "id": 0,  
      "name": "string",  
      "value": "string"  
    }  
  ],  
  "enabled": true,  
  "eventActionID": 0  
},
```

```
"name": "string",
"pattern": {
  "async": true,
  "enabled": true,
  "endTime": 0,
  "itemEvent": [
    {
      "attribute": {
        "dataType": 0,
        "name": "string",
        "value": "string"
      },
      "id": 0,
      "name": "string",
      "value": 0
    }
  ],
  "matchTotal": 0,
  "reset": {
    "id": 0,
    "idSelected": 0,
    "idToRemove": 0,
    "interval": 0,
    "intervalType": "string",
    "startTime": 0,
    "type": 0
  },
  "resetDuration": 0,
  "startTime": 0,
  "timeDuration": 0,
  "type": 0
},
```

```

"triggerResegmentation": {
  "conditions": [
    {
      "id": 0,
      "name": "string",
      "value": "string"
    }
  ],
  "enabled": true,
  "eventActionID": 0
},
"updateBy": 0,
"updateDate": 0
}

```

Create

POST /Campaign/api/interact/rest/v2/channels/{icldOrName}/eventpatterns

Input

Body Parameter – Content-Type: application/json

```

{
  "eventPatterns": [
    {
      "category": {
        "description": "string",
        "id": 0,
        "name": "string",
        "systemDefined": true
      },
      "createBy": 0,
      "createDate": 0,
      "description": "string",

```

```
"eventActions": [  
  {  
    "actionID": 0,  
    "actionValue": 0,  
    "async": true,  
    "condition": [  
      {  
        "id": 0,  
        "name": "string",  
        "value": "string"  
      }  
    ],  
    "details": "string",  
    "eventActionID": 0,  
    "eventType": 0,  
    "executionOrder": 0,  
    "inferredEvents": [  
      0  
    ]  
  }  
],  
"eventType": 0,  
"executeExpression": {  
  "conditions": [  
    {  
      "id": 0,  
      "name": "string",  
      "value": "string"  
    }  
  ],  
  "enabled": true,  
  "eventActionID": 0
```

```
},
"expression": {
  "audienceLevel": "string",
  "expression": "string",
  "fieldName": "string"
},
"id": 0,
"logOfferAcceptance": {
  "conditions": [
    {
      "id": 0,
      "name": "string",
      "value": "string"
    }
  ],
  "enabled": true,
  "eventActionID": 0
},
"logOfferContact": {
  "conditions": [
    {
      "id": 0,
      "name": "string",
      "value": "string"
    }
  ],
  "enabled": true,
  "eventActionID": 0
},
"logOfferRejection": {
  "conditions": [
    {
```

```

        "id": 0,
        "name": "string",
        "value": "string"
    }
],
"enabled": true,
"eventActionID": 0
},
"name": "string",
"pattern": {
    "async": true,
    "enabled": true,
    "endTime": 0,
    "itemEvent": [
        {
            "attribute": {
                "dataType": 0,
                "name": "string",
                "value": "string"
            },
            "id": 0,
            "name": "string",
            "value": 0
        }
    ],
    "matchTotal": 0,
    "reset": {
        "id": 0,
        "idSelected": 0,
        "idToRemove": 0,
        "interval": 0,
        "intervalType": "string",

```

```

        "startTime": 0,
        "type": 0
    },
    "resetDuration": 0,
    "startTime": 0,
    "timeDuration": 0,
    "type": 0
},
"triggerResegmentation": {
    "conditions": [
        {
            "id": 0,
            "name": "string",
            "value": "string"
        }
    ],
    "enabled": true,
    "eventActionID": 0
},
"updateBy": 0,
"updateDate": 0
}
]
}

```

Output

Table 28. Response Codes

Success	201	Created
Miscellaneous	207	Status code for batch request
Error	400	Bad Request

Server Error	500	Internal/Server Error
--------------	-----	-----------------------

Example: Content-Type: application/json

```
[
  {
    "headers": {},
    "body": <entity>/<error-msg>,
    "status": 201/400/500
  }
]
```

Update

PUT /Campaign/api/interact/rest/v2/channels/{icldOrName}/eventpatterns/{evtpatternIdOrName}

Input

Body Parameter – Content-Type: application/json

```
{
  "category": {
    "description": "string",
    "id": 0,
    "name": "string",
    "systemDefined": true
  },
  "createBy": 0,
  "createDate": 0,
  "description": "string",
  "eventActions": [
    {
      "actionID": 0,
      "actionValue": 0,

```



```

    "async": true,
    "condition": [
      {
        "id": 0,
        "name": "string",
        "value": "string"
      }
    ],
    "details": "string",
    "eventActionID": 0,
    "eventType": 0,
    "executionOrder": 0,
    "inferredEvents": [
      0
    ]
  },
  "eventType": 0,
  "executeExpression": {
    "conditions": [
      {
        "id": 0,
        "name": "string",
        "value": "string"
      }
    ],
    "enabled": true,
    "eventActionID": 0
  },
  "expression": {
    "audienceLevel": "string",
    "expression": "string",

```

```
    "fieldName": "string"
  },
  "id": 0,
  "logOfferAcceptance": {
    "conditions": [
      {
        "id": 0,
        "name": "string",
        "value": "string"
      }
    ],
    "enabled": true,
    "eventActionID": 0
  },
  "logOfferContact": {
    "conditions": [
      {
        "id": 0,
        "name": "string",
        "value": "string"
      }
    ],
    "enabled": true,
    "eventActionID": 0
  },
  "logOfferRejection": {
    "conditions": [
      {
        "id": 0,
        "name": "string",
        "value": "string"
      }
    ]
  }
}
```

```

    ],
    "enabled": true,
    "eventActionID": 0
  },
  "name": "string",
  "pattern": {
    "async": true,
    "enabled": true,
    "endTime": 0,
    "itemEvent": [
      {
        "attribute": {
          "dataType": 0,
          "name": "string",
          "value": "string"
        },
        "id": 0,
        "name": "string",
        "value": 0
      }
    ],
    "matchTotal": 0,
    "reset": {
      "id": 0,
      "idSelected": 0,
      "idToRemove": 0,
      "interval": 0,
      "intervalType": "string",
      "startTime": 0,
      "type": 0
    },
    "resetDuration": 0,

```

```

    "startTime": 0,
    "timeDuration": 0,
    "type": 0
  },
  "triggerResegmentation": {
    "conditions": [
      {
        "id": 0,
        "name": "string",
        "value": "string"
      }
    ],
    "enabled": true,
    "eventActionID": 0
  },
  "updateBy": 0,
  "updateDate": 0
}

```

Output

Table 29. Response Codes

Success	202	Accepted
Error	400	Bad Request
Server Error	500	Internal/Server Error

Example: Content-Type: application/json

```

{
  "category": {
    "description": "string",
    "id": 0,

```

```

    "name": "string",
    "systemDefined": true
  },
  "createBy": 0,
  "createDate": 0,
  "description": "string",
  "eventActions": [
    {
      "actionID": 0,
      "actionValue": 0,
      "async": true,
      "condition": [
        {
          "id": 0,
          "name": "string",
          "value": "string"
        }
      ],
      "details": "string",
      "eventActionID": 0,
      "eventType": 0,
      "executionOrder": 0,
      "inferredEvents": [0]
    }
  ],
  "eventType": 0,
  "executeExpression": {
    "conditions": [
      {
        "id": 0,
        "name": "string",
        "value": "string"
      }
    ]
  }
}

```

```

    }
  ],
  "enabled": true,
  "eventActionID": 0
},
"expression": {
  "audienceLevel": "string",
  "expression": "string",
  "fieldName": "string"
},
"id": 0,
"logOfferAcceptance": {
  "conditions": [
    {
      "id": 0,
      "name": "string",
      "value": "string"
    }
  ],
  "enabled": true,
  "eventActionID": 0
},
"logOfferContact": {
  "conditions": [
    {
      "id": 0,
      "name": "string",
      "value": "string"
    }
  ],
  "enabled": true,
  "eventActionID": 0
}

```

```
},
"logOfferRejection": {
  "conditions": [
    {
      "id": 0,
      "name": "string",
      "value": "string"
    }
  ],
  "enabled": true,
  "eventActionID": 0
},
"name": "string",
"pattern": {
  "async": true,
  "enabled": true,
  "endTime": 0,
  "itemEvent": [
    {
      "attribute": {
        "dataType": 0,
        "name": "string",
        "value": "string"
      },
      "id": 0,
      "name": "string",
      "value": 0
    }
  ],
  "matchTotal": 0,
  "reset": {
    "id": 0,
```

```

    "idSelected": 0,
    "idToRemove": 0,
    "interval": 0,
    "intervalType": "string",
    "startTime": 0,
    "type": 0
  },
  "resetDuration": 0,
  "startTime": 0,
  "timeDuration": 0,
  "type": 0
},
"triggerResegmentation": {
  "conditions": [
    {
      "id": 0,
      "name": "string",
      "value": "string"
    }
  ],
  "enabled": true,
  "eventActionID": 0
},
"updateBy": 0,
"updateDate": 0
}

```

Delete

DELETE /Campaign/api/interact/rest/v2/{icldOrName}/eventpatterns/
{evtpatternIdOrName}

Output

Table 30. Response Codes

Success	204	No Content
Error	400	Bad Request
Server Error	500	Internal/Server Error

Delete Multiple

DELETE /Campaign/api/interact/rest/v2/channels/{icldOrName}/
eventpatterns

Input

Body Parameter – Content-Type: application/json

```
{"ids": [0,0]}
```

Output**Table 31. Response Codes**

Success	204	No Content
Error	400	Bad Request
Server Error	500	Internal/Server Error

API - Strategy

Description

These APIs are used to perform CRUD operations on the Strategy.

Get All Strategies for Interactive Channel

GET /Campaign/api/interact/rest/v2/channels/{icldOrName}/strategies?
page=0&size=50

Parameters

Parameters	Default Values
page	0
size	50

Output

Table 32. Response Codes

Success	200	Ok
Error	400	Inputs provided for the API are not valid.
Server Error	500	Internal Server Error. For more details, check the logs.

Example: Content-Type: application/json

```
{
  "content": [
    {
      "campaignCode": "string",
      "campaignId": 0,
      "campaignName": "string",
      "createBy": "string",
      "createDate": 0,
      "deleteLock": 0,
      "description": "string",
      "editLock": 0,
      "id": 0,
      "interactiveChannelId": 0,
      "interactiveChannelName": "string",
    }
  ]
}
```

```
"lastProdDepTime": 0,
"name": "string",
"ruleCount": 0,
"runEnable": 0,
"updateBy": "string",
"updateDate": 0
}

],

"page": {

"pageNumber": 0,

"totalElements": 0,

"totalPages": 0,

"size": 0,

"hasNext": false,

"hasPrev": false

}

}
```

Get Strategy by Strategy Id

GET /Campaign/api/interact/rest/v2/channels/{icldOrName}/strategies/
{strategyIdOrName}

Parameters

Parameters	Default Values
page	0
size	50

Output

Table 33. Response Codes

Success	200	Ok
Error	400	Inputs provided for the API are not valid.
Server Error	500	Internal Server Error. For more details, check the logs.

Example: Content-Type: application/json

```
{
  "campaignCode": "string",
  "campaignId": 0,
  "campaignName": "string",
  "createBy": "string",
  "createDate": 0,
  "deleteLock": 0,
  "description": "string",
  "editLock": 0,
  "id": 0,
  "interactiveChannelId": 0,
  "interactiveChannelName": "string",
```

```
"lastProdDepTime": 0,  
"name": "string",  
"ruleCount": 0,  
"runEnable": 0,  
"updateBy": "string",  
"updateDate": 0  
  
}
```

Create Strategy

POST /Campaign/api/interact/rest/v2/channels/{icldOrName}/strategies

Input

Body Parameter – Content-Type:application/json

```
{  
  "campaignCode": "string",  
  "campaignId": 0,  
  "campaignName": "string",  
  "createBy": "string",  
  "createDate": 0,  
  "deleteLock": 0,  
  "description": "string",  
  "editLock": 0,  
  "id": 0,  
  "interactiveChannelId": 0,  
  "interactiveChannelName": "string",  
  "lastProdDepTime": 0,  
  "name": "string",  
  "ruleCount": 0,  
  "runEnable": 0,  
}
```

```

"updateBy": "string",
"updateDate": 0
}

```

Output

Table 34. Response Codes

Success	201	Created
Error	400	Bad Request
Server Error	500	Internal /Server Error.

Example: Content-Type: application/json

```

{
  "campaignCode": "string",
  "campaignId": 0,
  "campaignName": "string",
  "createBy": "string",
  "createDate": 0,
  "deleteLock": 0,
  "description": "string",
  "editLock": 0,
  "id": 0,
  "interactiveChannelId": 0,
  "interactiveChannelName": "string",
  "lastProdDepTime": 0,
  "name": "string",
  "ruleCount": 0,
  "runEnable": 0,
  "updateBy": "string",
  "updateDate": 0
}

```

```
}
```

Update

PUT /Campaign/api/interact/rest/v2/channels/{icIdOrName}/strategies/
{strategyIdOrName}

Input

Body Parameter – Content-Type:application/json

```
{  
  "campaignCode": "string",  
  "campaignId": 0,  
  "campaignName": "string",  
  "createBy": "string",  
  "createDate": 0,  
  "deleteLock": 0,  
  "description": "string",  
  "editLock": 0,  
  "id": 0,  
  "interactiveChannelId": 0,  
  "interactiveChannelName": "string",  
  "lastProdDepTime": 0,  
  "name": "string",  
  "ruleCount": 0,  
  "runEnable": 0,  
  "updateBy": "string",  
  "updateDate": 0  
}
```

Output

Table 35. Response Codes

Success	202	Accepted
Error	400	Bad Request
Server Error	500	Internal /Server Error.

```
{
  "campaignCode": "string",
  "campaignId": 0,
  "campaignName": "string",
  "createBy": "string",
  "createDate": 0,
  "deleteLock": 0,
  "description": "string",
  "editLock": 0,
  "id": 0,
  "interactiveChannelId": 0,
  "interactiveChannelName": "string",
  "lastProdDepTime": 0,
  "name": "string",
  "ruleCount": 0,
  "runEnable": 0,
  "updateBy": "string",
  "updateDate": 0
}
```

Delete

DELETE /Campaign/api/interact/rest/v2/channels/{icldOrName}/strategies/
{strategyIdOrName}

Output

Table 36. Response Codes

Success	204	No Content
Error	400	Bad Request
Server Error	500	Internal /Server Error.

Copy Strategy

POST /Campaign/api/interact/rest/v2/channels/{icldOrName}/strategies/
{strategyIdOrName}

Input

Body Parameter – Content-Type:application/json

```
{
  "campaignCode": "string",
  "campaignId": 0,
  "campaignName": "string",
  "createBy": "string",
  "createDate": 0,
  "deleteLock": 0,
  "description": "string",
  "editLock": 0,
  "id": 0,
  "interactiveChannelId": 0,
  "interactiveChannelName": "string",
  "lastProdDepTime": 0,
  "name": "string",
  "ruleCount": 0,
  "runEnable": 0,
  "updateBy": "string",
  "updateDate": 0
}
```

Output

Table 37. Response Codes

Success	202	Accepted
Error	400	Bad Request
Server Error	500	Internal /Server Error.

```
{
  "campaignCode": "string",
  "campaignId": 0,
  "campaignName": "string",
  "createBy": "string",
  "createDate": 0,
  "deleteLock": 0,
  "description": "string",
  "editLock": 0,
  "id": 0,
  "interactiveChannelId": 0,
  "interactiveChannelName": "string",
  "lastProdDepTime": 0,
  "name": "string",
  "ruleCount": 0,
  "runEnable": 0,
  "updateBy": "string",
  "updateDate": 0
}
```

API - Smart Rule

Description

These APIs are used to perform CRUD operations on the Smart Rules.

Get All Smart Rules for Interactive Channel

GET /Campaign/api/interact/rest/v2/channels/{icldOrName}/strategies/
{strategyIdOrName}/rules?page=0&size=50

Parameters

Parameters	Default Values
page	0
size	50

Output

Table 38. Response Codes

Success	200	Ok
Error	400	Inputs provided for the API are not valid.
Server Error	500	Internal Server Error. For more details, check the logs.

Example: Content-Type: application/json

```
{
  "content": [
    {
      "cellCode": "string",
      "cellId": 0,
      "effectiveDate": 0,
      "eligibilityPredicate": "string",
      "eligibilityPredicateEnabled": true,
      "eligibilityPredicateId": 0,
    }
  ]
}
```

```
"enabled": true,
"expirationDate": 0,
"id": 0,
"learningMode": 0,
"learningModelId": 0,
"lmName": "string",
"maxScore": true,
"offerAttrExpressions": [
  {
    "attrId": 0,
    "attrName": "string",
    "expression": "string",
    "ruleId": 0
  }
],
"offerCode": "string",
"offerId": 0,
"offerName": "string",
"ruleGroupName": "string",
"ruleGroups": [
  {
    "description": "string",
    "id": 0,
    "name": "string"
  }
],
"ruleName": "string",
"samplePercent": 0,
"score": 0,
"scorePredicate": "string",
"scorePredicateEnabled": true,
"scorePredicateId": 0,
```

```

    "segmentId": 0,
    "segmentName": "string",
    "suppressCount": 0,
    "zoneId": 0,
    "zoneName": "string"
  }],

  "page": {

    "pageNumber": 0,

    "totalElements": 0,

    "totalPages": 0,

    "size": 0,
    "hasNext": false,

    "hasPrev": false

  }

}

```

Get Smart Rule by Rule Id

GET /Campaign/api/interact/rest/v2/channels/{icldOrName}/strategies/
 {strategyIdOrName}/rules/{ruleIdOrName}

Parameters

Parameters	Default Values
page	0

Parameters	Default Values
size	50

Output

Table 39. Response Codes

Success	200	Ok
Error	400	Inputs provided for the API are not valid.
Server Error	500	Internal Server Error. For more details, check the logs.

Example: Content-Type: application/json

```
{
  "cellCode": "string",
  "cellId": 0,
  "effectiveDate": 0,
  "eligibilityPredicate": "string",
  "eligibilityPredicateEnabled": true,
  "eligibilityPredicateId": 0,
  "enabled": true,
  "expirationDate": 0,
  "id": 0,
  "learningMode": 0,
  "learningModelId": 0,
  "lmName": "string",
  "maxScore": true,
  "offerAttrExpressions": [
    {
      "attrId": 0,
```

```

    "attrName": "string",
    "expression": "string",
    "ruleId": 0
  }
],
"offerCode": "string",
"offerId": 0,
"offerName": "string",
"ruleGroupName": "string",
"ruleGroups": [
  {
    "description": "string",
    "id": 0,
    "name": "string"
  }
],
"ruleName": "string",
"samplePercent": 0,
"score": 0,
"scorePredicate": "string",
"scorePredicateEnabled": true,
"scorePredicateId": 0,
"segmentId": 0,
"segmentName": "string",
"suppressCount": 0,
"zoneId": 0,
"zoneName": "string"
}

```

Save Rules

POST /Campaign/api/interact/rest/v2/channels/{icldOrName}/strategies/
 {strategyIdOrName}/rules

Input

Body Parameter – Content-Type:application/json

```
{
  "toAdd": [
    {
      "cellCode": "string",
      "cellId": 0,
      "effectiveDate": 0,
      "eligibilityPredicate": "string",
      "eligibilityPredicateEnabled": true,
      "eligibilityPredicateId": 0,
      "enabled": true,
      "expirationDate": 0,
      "id": 0,
      "learningMode": 0,
      "learningModelId": 0,
      "lmName": "string",
      "maxScore": true,
      "offerAttrExpressions": [
        {
          "attrId": 0,
          "attrName": "string",
          "expression": "string",
          "ruleId": 0
        }
      ],
      "offerCode": "string",
      "offerId": 0,
      "offerName": "string",
      "ruleGroupName": "string",
      "ruleGroups": [
```



```

    {
      "description": "string",
      "id": 0,
      "name": "string"
    }
  ],
  "ruleName": "string",
  "samplePercent": 0,
  "score": 0,
  "scorePredicate": "string",
  "scorePredicateEnabled": true,
  "scorePredicateId": 0,
  "segmentId": 0,
  "segmentName": "string",
  "suppressCount": 0,
  "zoneId": 0,
  "zoneName": "string"
}
],
"toDelete": [
  0
],
"toUpdate": [
  {
    "cellCode": "string",
    "cellId": 0,
    "effectiveDate": 0,
    "eligibilityPredicate": "string",
    "eligibilityPredicateEnabled": true,
    "eligibilityPredicateId": 0,
    "enabled": true,
    "expirationDate": 0,

```

```
"id": 0,
"learningMode": 0,
"learningModelId": 0,
"lmName": "string",
"maxScore": true,
"offerAttrExpressions": [
  {
    "attrId": 0,
    "attrName": "string",
    "expression": "string",
    "ruleId": 0
  }
],
"offerCode": "string",
"offerId": 0,
"offerName": "string",
"ruleGroupName": "string",
"ruleGroups": [
  {
    "description": "string",
    "id": 0,
    "name": "string"
  }
],
"ruleName": "string",
"samplePercent": 0,
"score": 0,
"scorePredicate": "string",
"scorePredicateEnabled": true,
"scorePredicateId": 0,
"segmentId": 0,
"segmentName": "string",
```

```

    "suppressCount": 0,
    "zoneId": 0,
    "zoneName": "string"
  }
]
}

```

Output

Table 40. Response Codes

Success	201	Created
Error	400	Bad Request
Server Error	500	Internal/Server Error

Delete

DELETE /Campaign/api/interact/rest/v2/channels/{icldOrName}/strategies/{strategyIdOrName}/rules

Input

Body Parameter – Content-Type:application/json

```

{
  "ids": [
    0
  ]
}

```

Output

Table 41. Response Codes

Success	204	No Content
Error	400	Bad Request

Server Error	500	Internal/Server Error
--------------	-----	-----------------------

API - Strategy Deployment

Description

These APIs are used to perform CRUD operations on the Strategy Deployment.

Get Deployment State

GET /Campaign/api/interact/rest/v2/channels/{icldOrName}/strategies/{strategyIdOrName}/deployment?deployAction=getDeploymentState

Output

Table 42. Response Codes

Success	200	Ok
Error	400	Inputs provided for the API are not valid.
Server Error	500	Internal Server Error. For more details, check the logs.

Body Parameter – Content-Type:application/json

```
{
  "campaignId": 0,
  "deployAction": "",
  "state": 0,
  "deployedState": 0,
  "dirty": false,
  "lastProdDepTime": 0,
  "lastMarkTime": 0,
  "deploymentStatusText": ""
```

```

"deploymentWaitMessage": ""
}

```

Mark for Deployment Or Un-Deployment

PUT /Campaign/api/interact/rest/v2/channels/{icIdOrName}/strategies/{strategyIdOrName}/deployment

Input

Body Parameter – Content-Type:application/json

```

{
  "campaignId": 0,
  "deployAction" : ""
}

```

Output

Table 43. Response Codes

Success	202	Accepted
Error	400	Bad Request
Server Error	500	Internal/Server Error

Body Parameter – Content-Type:application/json

```

{
  "campaignId": 0,
  "deployAction": "",
  "state": 0,
  "deployedState": 0,
  "dirty": false,
  "lastProdDepTime": 0,
  "lastMarkTime": 0,
  "deploymentStatusText": "",

```

```

"deploymentWaitMessage": ""
}

```

Cancel Mark for Deployment Or Un-Deployment

DELETE /Campaign/api/interact/rest/v2/channels/{icldOrName}/strategies/
{strategyIdOrName}/deployment

Input

Body Parameter – Content-Type:application/json

```

{
  "campaignId": 0,
  "deployAction" : ""
}

```

Output

Table 44. Response Codes

Success	204	No Content
Error	400	Bad Request
Server Error	500	Internal/Server Error

API - Smart Rule Group

Description

These APIs are used to perform CRUD operations on the Smart Rule Group.

Get Smart Rule Groups by Strategy

GET /Campaign/api/interact/rest/v2/channels/{icldOrName}/strategies/
{strategyIdOrName}/ruleGroups?page=0&size=50

Parameters

Parameters	Default Values
page	0
size	50

Output

Table 45. Response Codes

Success	200	Ok
Error	400	Inputs provided for the API are not valid.
Server Error	500	Internal Server Error. For more details, check the logs.

Example: Content-Type: application/json

```
{
  "content": [
    {
      "id": 0,
      "name": "String",
      "strategyId": 0,
      "ruleGroupType": "String",
      "description": "String",
      "createDate": 0,
      "createBy": 0,
      "updateDate": 0,
      "updateBy": 0,
      "ruleIdList": ["0","0"]
    }
  ],
  "page": {
```

```

    "pageNumber": 0,
    "totalElements": 0,
    "totalPages": 0,
    "size": 0,
    "hasNext": false,
    "hasPrev": false
  }
}

```

Get Smart Rule Groups by Rule

GET /Campaign/api/interact/rest/v2/channels/{icldOrName}/strategies/
{strategyIdOrName}/ruleGroups/{ruleIdOrName}?page=0&size=50

Parameters

Parameters	Default Values
page	0
size	50

Output

Table 46. Response Codes

Success	200	Ok
Error	400	Inputs provided for the API are not valid.
Server Error	500	Internal Server Error. For more details, check the logs.

Example: Content-Type: application/json

```

{
  "content": [

```



```

    {
      "id": 0,
      "name": "String",
      "strategyId": 0,
      "ruleGroupType": "String",
      "description": "String",
      "createDate": 0,
      "createBy": 0,
      "updateDate": 0,
      "updateBy": 0,
      "ruleIdList": ["0","0"]
    }
  ],
  "page": {
    "pageNumber": 0,
    "totalElements": 0,
    "totalPages": 0,
    "size": 0,
    "hasNext": false,
    "hasPrev": false
  }
}

```

Create

POST /Campaign/api/interact/rest/v2/channels/{icldOrName}/strategies/
 {strategyIdOrName}/ruleGroups

Input

Body Parameter – Content-Type:application/json

```

{
  "id": 0,
  "name": "String",

```

```

    "strategyId": 0,
    "ruleGroupType": "String",
    "description": "String",
    "createDate": 0,
    "createBy": 0,
    "updateDate": 0,
    "updateBy": 0,
    "ruleIdList": []
  }

```

Output

Table 47. Response Codes

Success	201	Created
Error	400	Bad Request
Server Error	500	Internal /Server Error

Example: Content-Type: application/json

```

{
  "id": 0,
  "name": "String",
  "strategyId": 0,
  "ruleGroupType": "String",
  "description": "String",
  "createDate": 0,
  "createBy": 0,
  "updateDate": 0,
  "updateBy": 0,
  "ruleIdList": []
}

```

Update

PUT /Campaign/api/interact/rest/v2/channels/{icIdOrName}/strategies/
 {strategyIdOrName}/ruleGroups/{ruleGroupIdOrName}

Input

Body Parameter – Content-Type:application/json

```
{
  "id": 0,
  "name": "String",
  "strategyId": 0,
  "ruleGroupType": "String",
  "description": "String",
  "createDate": 0,
  "createBy": 0,
  "updateDate": 0,
  "updateBy": 0,
  "ruleIdList": []
}
```

Output

Table 48. Response Codes

Success	202	Accepted
Error	400	Bad Request
Server Error	500	Internal/Server Error

```
{
  "id": 0,
  "name": "String",
  "strategyId": 0,
  "ruleGroupType": "String",
```

```

    "description": "String",
    "createDate": 0,
    "createBy": 0,
    "updateDate": 0,
    "updateBy": 0,
    "ruleIdList": []
  }

```

Delete

DELETE /Campaign/api/interact/rest/v2/channels/{icldOrName}/strategies/{strategyIdOrName}/ruleGroups/{ruleGroupIdOrName}

Output

Table 49. Response Codes

Success	204	No Content
Error	400	Bad Request
Server Error	500	Internal/Server Error

API - Interactive Channel Deployment

Description

This API is used to deploy interactive channel.

Deploy/Un-deploy

POST /Campaign/api/interact/rest/v2/deployments

Parameters

Parameters	Values
request	1: Deploy (default)

	-1: Un-deploy
includeGlobal	true (default) false
globalOnly	true false (default)

Body Parameter – Content-Type: application/json

```
{
  "icIdOrName": "string",
  "rtPassword": "string",
  "rtUsername": "string",
  "serverGroup": "string",
  "serverGroupType": 0
}
```

Output

Table 50. Response Codes

Success	200	Ok
Error	400	Bad Request
Server Error	500	Internal Server Error