# **QeuBox Low Level Design Document**

## **Screen Name: Order Etl Hdr**

### **Database Table Name: order\_etl\_hdr**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| order\_etl\_hdr\_sno | smallint - (Numeric Precision: 16) | order\_etl\_hdr\_pkey (PRIMARY KEY), |
| delivery\_partner\_sno | smallint - (Numeric Precision: 16) | order\_etl\_hdr\_delivery\_partner\_sno\_fkey (FOREIGN KEY), |
| partner\_code | character varying | null |
| staging\_table\_name | character varying | null |
| is\_active | boolean | null |
| file\_name\_prefix | character varying | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Order Etl Hdr Sno | hidden | number | true | Manual Entry |
| Delivery Partner Sno | listbox | number | true | API feed list box selection |
| Partner Code | textbox | string | true | Manual Entry |
| Staging Table Name | textbox | string | false | Manual Entry |
| Is Active | checkbox | boolean | false | Manual Entry |
| File Name Prefix | textbox | string | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_order\_etl\_hdrs | get | /api/v1/get\_order\_etl\_hdrs |
| create\_order\_etl\_hdr | post | /api/v1/create\_order\_etl\_hdr |
| update\_order\_etl\_hdr | put | /api/v1/update\_order\_etl\_hdr |
| delete\_order\_etl\_hdr | delete | /api/v1/delete\_order\_etl\_hdr |
| get\_delivery\_partners | get | /api/v1/get\_delivery\_partners |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_order\_etl\_hdrs | {} | {"data":{"file\_name\_prefix":"character varying","is\_active":"boolean","delivery\_partner\_sno":"smallint","partner\_code":"character varying","staging\_table\_name":"character varying","order\_etl\_hdr\_sno":"smallint"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_order\_etl\_hdr | {"data":{"file\_name\_prefix":"character varying","is\_active":"boolean","delivery\_partner\_sno":"smallint","partner\_code":"character varying","staging\_table\_name":"character varying","order\_etl\_hdr\_sno":"smallint"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_order\_etl\_hdr | {"data":{"file\_name\_prefix":"character varying","is\_active":"boolean","delivery\_partner\_sno":"smallint","partner\_code":"character varying","staging\_table\_name":"character varying","order\_etl\_hdr\_sno":"smallint"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_order\_etl\_hdr | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_delivery\_partners | {"id":"integer"} | {"data":"array of delivery\_partner"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| custom\_delete\_order\_etl\_hdr\_by\_id | DELETE FROM partner\_order.order\_etl\_hdr WHERE <custom\_conditions>; |
| update\_order\_etl\_hdr | UPDATE partner\_order.order\_etl\_hdr SET order\_etl\_hdr\_sno = quote\_nullable(i\_order\_etl\_hdr\_sno), delivery\_partner\_sno = quote\_nullable(i\_delivery\_partner\_sno), partner\_code = quote\_nullable(i\_partner\_code), staging\_table\_name = quote\_nullable(i\_staging\_table\_name), is\_active = quote\_nullable(i\_is\_active), file\_name\_prefix = quote\_nullable(i\_file\_name\_prefix) WHERE order\_etl\_hdr\_sno = quote\_nullable(i\_order\_etl\_hdr\_sno); |
| create\_order\_etl\_hdr | INSERT INTO partner\_order.order\_etl\_hdr (t.order\_etl\_hdr\_sno, t.delivery\_partner\_sno, t.partner\_code, t.staging\_table\_name, t.is\_active, t.file\_name\_prefix) VALUES (i\_order\_etl\_hdr\_sno, i\_delivery\_partner\_sno, i\_partner\_code, i\_staging\_table\_name, i\_is\_active, i\_file\_name\_prefix); |
| custom\_search\_order\_etl\_hdr\_by\_id | SELECT t.order\_etl\_hdr\_sno, t.delivery\_partner\_sno, t.partner\_code, t.staging\_table\_name, t.is\_active, t.file\_name\_prefix FROM partner\_order.order\_etl\_hdr t WHERE <custom\_search\_conditions>; |
| get\_order\_etl\_hdr\_by\_id | SELECT t.order\_etl\_hdr\_sno, t.delivery\_partner\_sno, t.partner\_code, t.staging\_table\_name, t.is\_active, t.file\_name\_prefix FROM partner\_order.order\_etl\_hdr t WHERE t.order\_etl\_hdr\_sno = quote\_nullable(i\_order\_etl\_hdr\_sno); |
| get\_all\_order\_etl\_hdrs | SELECT t.order\_etl\_hdr\_sno, t.delivery\_partner\_sno, t.partner\_code, t.staging\_table\_name, t.is\_active, t.file\_name\_prefix FROM partner\_order.order\_etl\_hdr t ; |
| delete\_order\_etl\_hdr\_by\_id | DELETE FROM partner\_order.order\_etl\_hdr WHERE order\_etl\_hdr\_sno = quote\_nullable(i\_order\_etl\_hdr\_sno); |
| custom\_update\_order\_etl\_hdr | UPDATE partner\_order.order\_etl\_hdr SET order\_etl\_hdr\_sno = quote\_nullable(i\_order\_etl\_hdr\_sno), delivery\_partner\_sno = quote\_nullable(i\_delivery\_partner\_sno), partner\_code = quote\_nullable(i\_partner\_code), staging\_table\_name = quote\_nullable(i\_staging\_table\_name), is\_active = quote\_nullable(i\_is\_active), file\_name\_prefix = quote\_nullable(i\_file\_name\_prefix) WHERE <custom\_conditions>; |
| unique\_order\_etl\_hdr\_exists | SELECT 1 FROM partner\_order.order\_etl\_hdr WHERE <unique\_conditions> LIMIT 1; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Etl Table Column**

### **Database Table Name: etl\_table\_column**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| etl\_table\_column\_sno | smallint - (Numeric Precision: 16) | etl\_table\_column\_pkey (PRIMARY KEY), |
| order\_etl\_hdr\_sno | smallint - (Numeric Precision: 16) | etl\_table\_column\_order\_etl\_table\_sno\_fkey (FOREIGN KEY), |
| source\_column | character varying | null |
| staging\_column | character varying | null |
| data\_type | character varying | null |
| is\_required | boolean | null |
| description | text | null |
| created\_at | timestamp without time zone | null |
| updated\_at | timestamp without time zone | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Etl Table Column Sno | hidden | number | true | Manual Entry |
| Order Etl Hdr Sno | listbox | number | true | API feed list box selection |
| Source Column | textbox | string | false | Manual Entry |
| Staging Column | textbox | string | false | Manual Entry |
| Data Type | textbox | string | false | Manual Entry |
| Is Required | checkbox | boolean | false | Manual Entry |
| Description | textbox | string | false | Manual Entry |
| Created At | textbox | string | false | Manual Entry |
| Updated At | textbox | string | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_etl\_table\_columns | get | /api/v1/get\_etl\_table\_columns |
| create\_etl\_table\_column | post | /api/v1/create\_etl\_table\_column |
| update\_etl\_table\_column | put | /api/v1/update\_etl\_table\_column |
| delete\_etl\_table\_column | delete | /api/v1/delete\_etl\_table\_column |
| get\_order\_etl\_hdrs | get | /api/v1/get\_order\_etl\_hdrs |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_etl\_table\_columns | {} | {"data":{"source\_column":"character varying","updated\_at":"timestamp without time zone","is\_required":"boolean","data\_type":"character varying","created\_at":"timestamp without time zone","description":"text","staging\_column":"character varying","etl\_table\_column\_sno":"smallint","order\_etl\_hdr\_sno":"smallint"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_etl\_table\_column | {"data":{"source\_column":"character varying","updated\_at":"timestamp without time zone","is\_required":"boolean","data\_type":"character varying","created\_at":"timestamp without time zone","description":"text","staging\_column":"character varying","etl\_table\_column\_sno":"smallint","order\_etl\_hdr\_sno":"smallint"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_etl\_table\_column | {"data":{"source\_column":"character varying","updated\_at":"timestamp without time zone","is\_required":"boolean","data\_type":"character varying","created\_at":"timestamp without time zone","description":"text","staging\_column":"character varying","etl\_table\_column\_sno":"smallint","order\_etl\_hdr\_sno":"smallint"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_etl\_table\_column | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_order\_etl\_hdrs | {"id":"integer"} | {"data":"array of order\_etl\_hdr"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| get\_all\_etl\_table\_columns | SELECT t.etl\_table\_column\_sno, t.order\_etl\_hdr\_sno, t.source\_column, t.staging\_column, t.data\_type, t.is\_required, t.description, t.created\_at, t.updated\_at, f\_order\_etl\_hdr.staging\_table\_name FROM partner\_order.etl\_table\_column t LEFT JOIN partner\_order.order\_etl\_hdr f\_order\_etl\_hdr ON t.order\_etl\_hdr\_sno = f\_order\_etl\_hdr.order\_etl\_hdr\_sno ; |
| custom\_delete\_etl\_table\_column\_by\_id | DELETE FROM partner\_order.etl\_table\_column WHERE <custom\_conditions>; |
| update\_etl\_table\_column | UPDATE partner\_order.etl\_table\_column SET etl\_table\_column\_sno = quote\_nullable(i\_etl\_table\_column\_sno), order\_etl\_hdr\_sno = quote\_nullable(i\_order\_etl\_hdr\_sno), source\_column = quote\_nullable(i\_source\_column), staging\_column = quote\_nullable(i\_staging\_column), data\_type = quote\_nullable(i\_data\_type), is\_required = quote\_nullable(i\_is\_required), description = quote\_nullable(i\_description), created\_at = quote\_nullable(i\_created\_at), updated\_at = quote\_nullable(i\_updated\_at) WHERE etl\_table\_column\_sno = quote\_nullable(i\_etl\_table\_column\_sno); |
| get\_etl\_table\_column\_by\_id | SELECT t.etl\_table\_column\_sno, t.order\_etl\_hdr\_sno, t.source\_column, t.staging\_column, t.data\_type, t.is\_required, t.description, t.created\_at, t.updated\_at, f\_order\_etl\_hdr.staging\_table\_name FROM partner\_order.etl\_table\_column t LEFT JOIN partner\_order.order\_etl\_hdr f\_order\_etl\_hdr ON t.order\_etl\_hdr\_sno = f\_order\_etl\_hdr.order\_etl\_hdr\_sno WHERE t.etl\_table\_column\_sno = quote\_nullable(i\_etl\_table\_column\_sno); |
| custom\_update\_etl\_table\_column | UPDATE partner\_order.etl\_table\_column SET etl\_table\_column\_sno = quote\_nullable(i\_etl\_table\_column\_sno), order\_etl\_hdr\_sno = quote\_nullable(i\_order\_etl\_hdr\_sno), source\_column = quote\_nullable(i\_source\_column), staging\_column = quote\_nullable(i\_staging\_column), data\_type = quote\_nullable(i\_data\_type), is\_required = quote\_nullable(i\_is\_required), description = quote\_nullable(i\_description), created\_at = quote\_nullable(i\_created\_at), updated\_at = quote\_nullable(i\_updated\_at) WHERE <custom\_conditions>; |
| custom\_search\_etl\_table\_column\_by\_id | SELECT t.etl\_table\_column\_sno, t.order\_etl\_hdr\_sno, t.source\_column, t.staging\_column, t.data\_type, t.is\_required, t.description, t.created\_at, t.updated\_at, f\_order\_etl\_hdr.staging\_table\_name FROM partner\_order.etl\_table\_column t LEFT JOIN partner\_order.order\_etl\_hdr f\_order\_etl\_hdr ON t.order\_etl\_hdr\_sno = f\_order\_etl\_hdr.order\_etl\_hdr\_sno WHERE <custom\_search\_conditions>; |
| create\_etl\_table\_column | INSERT INTO partner\_order.etl\_table\_column (t.etl\_table\_column\_sno, t.order\_etl\_hdr\_sno, t.source\_column, t.staging\_column, t.data\_type, t.is\_required, t.description, t.created\_at, t.updated\_at) VALUES (i\_etl\_table\_column\_sno, i\_order\_etl\_hdr\_sno, i\_source\_column, i\_staging\_column, i\_data\_type, i\_is\_required, i\_description, i\_created\_at, i\_updated\_at); |
| delete\_etl\_table\_column\_by\_id | DELETE FROM partner\_order.etl\_table\_column WHERE etl\_table\_column\_sno = quote\_nullable(i\_etl\_table\_column\_sno); |
| unique\_etl\_table\_column\_exists | SELECT 1 FROM partner\_order.etl\_table\_column WHERE <unique\_conditions> LIMIT 1; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Etl Job**

### **Database Table Name: etl\_job**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| etl\_job\_sno | integer | etl\_job\_tracking\_pkey (PRIMARY KEY), |
| order\_etl\_hdr\_sno | integer | order\_etl\_hdr\_sno\_id\_fkey (FOREIGN KEY), |
| file\_name | character varying | null |
| file\_load\_timestamp | timestamp without time zone | null |
| etl\_start\_timestamp | timestamp without time zone | null |
| etl\_end\_timestamp | timestamp without time zone | null |
| status | character varying | null |
| error\_message | text | null |
| record\_count | integer | null |
| processed\_by | character varying | null |
| last\_updated | timestamp without time zone | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Etl Job Sno | hidden | number | true | Manual Entry |
| Order Etl Hdr Sno | listbox | number | true | API feed list box selection |
| File Name | textbox | string | false | Manual Entry |
| File Load Timestamp | textbox | string | false | Manual Entry |
| Etl Start Timestamp | textbox | string | false | Manual Entry |
| Etl End Timestamp | textbox | string | false | Manual Entry |
| Status | textbox | string | false | Manual Entry |
| Error Message | textbox | string | false | Manual Entry |
| Record Count | textbox | number | false | Manual Entry |
| Processed By | textbox | string | false | Manual Entry |
| Last Updated | textbox | string | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_etl\_jobs | get | /api/v1/get\_etl\_jobs |
| create\_etl\_job | post | /api/v1/create\_etl\_job |
| update\_etl\_job | put | /api/v1/update\_etl\_job |
| delete\_etl\_job | delete | /api/v1/delete\_etl\_job |
| get\_order\_etl\_hdrs | get | /api/v1/get\_order\_etl\_hdrs |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_etl\_jobs | {} | {"data":{"etl\_job\_sno":"integer","record\_count":"integer","error\_message":"text","last\_updated":"timestamp without time zone","etl\_end\_timestamp":"timestamp without time zone","processed\_by":"character varying","file\_name":"character varying","file\_load\_timestamp":"timestamp without time zone","etl\_start\_timestamp":"timestamp without time zone","status":"character varying","order\_etl\_hdr\_sno":"integer"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_etl\_job | {"data":{"etl\_job\_sno":"integer","record\_count":"integer","error\_message":"text","last\_updated":"timestamp without time zone","etl\_end\_timestamp":"timestamp without time zone","processed\_by":"character varying","file\_name":"character varying","file\_load\_timestamp":"timestamp without time zone","etl\_start\_timestamp":"timestamp without time zone","status":"character varying","order\_etl\_hdr\_sno":"integer"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_etl\_job | {"data":{"etl\_job\_sno":"integer","record\_count":"integer","error\_message":"text","last\_updated":"timestamp without time zone","etl\_end\_timestamp":"timestamp without time zone","processed\_by":"character varying","file\_name":"character varying","file\_load\_timestamp":"timestamp without time zone","etl\_start\_timestamp":"timestamp without time zone","status":"character varying","order\_etl\_hdr\_sno":"integer"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_etl\_job | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_order\_etl\_hdrs | {"id":"integer"} | {"data":"array of order\_etl\_hdr"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| create\_etl\_job | INSERT INTO partner\_order.etl\_job (t.etl\_job\_sno, t.order\_etl\_hdr\_sno, t.file\_name, t.file\_load\_timestamp, t.etl\_start\_timestamp, t.etl\_end\_timestamp, t.status, t.error\_message, t.record\_count, t.processed\_by, t.last\_updated) VALUES (i\_etl\_job\_sno, i\_order\_etl\_hdr\_sno, i\_file\_name, i\_file\_load\_timestamp, i\_etl\_start\_timestamp, i\_etl\_end\_timestamp, i\_status, i\_error\_message, i\_record\_count, i\_processed\_by, i\_last\_updated); |
| get\_all\_etl\_jobs | SELECT t.etl\_job\_sno, t.order\_etl\_hdr\_sno, t.file\_name, t.file\_load\_timestamp, t.etl\_start\_timestamp, t.etl\_end\_timestamp, t.status, t.error\_message, t.record\_count, t.processed\_by, t.last\_updated, f\_order\_etl\_hdr.staging\_table\_name FROM partner\_order.etl\_job t LEFT JOIN partner\_order.order\_etl\_hdr f\_order\_etl\_hdr ON t.order\_etl\_hdr\_sno = f\_order\_etl\_hdr.order\_etl\_hdr\_sno ; |
| delete\_etl\_job\_by\_id | DELETE FROM partner\_order.etl\_job WHERE etl\_job\_sno = quote\_nullable(i\_etl\_job\_sno); |
| get\_etl\_job\_by\_id | SELECT t.etl\_job\_sno, t.order\_etl\_hdr\_sno, t.file\_name, t.file\_load\_timestamp, t.etl\_start\_timestamp, t.etl\_end\_timestamp, t.status, t.error\_message, t.record\_count, t.processed\_by, t.last\_updated, f\_order\_etl\_hdr.staging\_table\_name FROM partner\_order.etl\_job t LEFT JOIN partner\_order.order\_etl\_hdr f\_order\_etl\_hdr ON t.order\_etl\_hdr\_sno = f\_order\_etl\_hdr.order\_etl\_hdr\_sno WHERE t.etl\_job\_sno = quote\_nullable(i\_etl\_job\_sno); |
| custom\_update\_etl\_job | UPDATE partner\_order.etl\_job SET etl\_job\_sno = quote\_nullable(i\_etl\_job\_sno), order\_etl\_hdr\_sno = quote\_nullable(i\_order\_etl\_hdr\_sno), file\_name = quote\_nullable(i\_file\_name), file\_load\_timestamp = quote\_nullable(i\_file\_load\_timestamp), etl\_start\_timestamp = quote\_nullable(i\_etl\_start\_timestamp), etl\_end\_timestamp = quote\_nullable(i\_etl\_end\_timestamp), status = quote\_nullable(i\_status), error\_message = quote\_nullable(i\_error\_message), record\_count = quote\_nullable(i\_record\_count), processed\_by = quote\_nullable(i\_processed\_by), last\_updated = quote\_nullable(i\_last\_updated) WHERE <custom\_conditions>; |
| custom\_delete\_etl\_job\_by\_id | DELETE FROM partner\_order.etl\_job WHERE <custom\_conditions>; |
| update\_etl\_job | UPDATE partner\_order.etl\_job SET etl\_job\_sno = quote\_nullable(i\_etl\_job\_sno), order\_etl\_hdr\_sno = quote\_nullable(i\_order\_etl\_hdr\_sno), file\_name = quote\_nullable(i\_file\_name), file\_load\_timestamp = quote\_nullable(i\_file\_load\_timestamp), etl\_start\_timestamp = quote\_nullable(i\_etl\_start\_timestamp), etl\_end\_timestamp = quote\_nullable(i\_etl\_end\_timestamp), status = quote\_nullable(i\_status), error\_message = quote\_nullable(i\_error\_message), record\_count = quote\_nullable(i\_record\_count), processed\_by = quote\_nullable(i\_processed\_by), last\_updated = quote\_nullable(i\_last\_updated) WHERE etl\_job\_sno = quote\_nullable(i\_etl\_job\_sno); |
| unique\_etl\_job\_exists | SELECT 1 FROM partner\_order.etl\_job WHERE <unique\_conditions> LIMIT 1; |
| custom\_search\_etl\_job\_by\_id | SELECT t.etl\_job\_sno, t.order\_etl\_hdr\_sno, t.file\_name, t.file\_load\_timestamp, t.etl\_start\_timestamp, t.etl\_end\_timestamp, t.status, t.error\_message, t.record\_count, t.processed\_by, t.last\_updated, f\_order\_etl\_hdr.staging\_table\_name FROM partner\_order.etl\_job t LEFT JOIN partner\_order.order\_etl\_hdr f\_order\_etl\_hdr ON t.order\_etl\_hdr\_sno = f\_order\_etl\_hdr.order\_etl\_hdr\_sno WHERE <custom\_search\_conditions>; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Etl Record**

### **Database Table Name: etl\_record**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| etl\_record\_sno | integer | row\_processing\_status\_pkey (PRIMARY KEY), |
| etl\_job\_sno | integer | row\_processing\_status\_job\_id\_fkey (FOREIGN KEY), |
| staging\_table\_name | character varying | null |
| row\_id | integer | null |
| processing\_status | character varying | null |
| processing\_timestamp | timestamp without time zone | null |
| error\_message | text | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Etl Record Sno | hidden | number | true | Manual Entry |
| Etl Job Sno | listbox | number | false | API feed list box selection |
| Staging Table Name | textbox | string | false | Manual Entry |
| Row Id | textbox | number | false | Manual Entry |
| Processing Status | textbox | string | false | Manual Entry |
| Processing Timestamp | textbox | string | false | Manual Entry |
| Error Message | textbox | string | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_etl\_records | get | /api/v1/get\_etl\_records |
| create\_etl\_record | post | /api/v1/create\_etl\_record |
| update\_etl\_record | put | /api/v1/update\_etl\_record |
| delete\_etl\_record | delete | /api/v1/delete\_etl\_record |
| get\_etl\_jobs | get | /api/v1/get\_etl\_jobs |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_etl\_records | {} | {"data":{"etl\_job\_sno":"integer","error\_message":"text","processing\_timestamp":"timestamp without time zone","processing\_status":"character varying","etl\_record\_sno":"integer","staging\_table\_name":"character varying","row\_id":"integer"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_etl\_record | {"data":{"etl\_job\_sno":"integer","error\_message":"text","processing\_timestamp":"timestamp without time zone","processing\_status":"character varying","etl\_record\_sno":"integer","staging\_table\_name":"character varying","row\_id":"integer"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_etl\_record | {"data":{"etl\_job\_sno":"integer","error\_message":"text","processing\_timestamp":"timestamp without time zone","processing\_status":"character varying","etl\_record\_sno":"integer","staging\_table\_name":"character varying","row\_id":"integer"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_etl\_record | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_etl\_jobs | {"id":"integer"} | {"data":"array of etl\_job"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| custom\_delete\_etl\_record\_by\_id | DELETE FROM partner\_order.etl\_record WHERE <custom\_conditions>; |
| create\_etl\_record | INSERT INTO partner\_order.etl\_record (t.etl\_record\_sno, t.etl\_job\_sno, t.staging\_table\_name, t.row\_id, t.processing\_status, t.processing\_timestamp, t.error\_message) VALUES (i\_etl\_record\_sno, i\_etl\_job\_sno, i\_staging\_table\_name, i\_row\_id, i\_processing\_status, i\_processing\_timestamp, i\_error\_message); |
| custom\_update\_etl\_record | UPDATE partner\_order.etl\_record SET etl\_record\_sno = quote\_nullable(i\_etl\_record\_sno), etl\_job\_sno = quote\_nullable(i\_etl\_job\_sno), staging\_table\_name = quote\_nullable(i\_staging\_table\_name), row\_id = quote\_nullable(i\_row\_id), processing\_status = quote\_nullable(i\_processing\_status), processing\_timestamp = quote\_nullable(i\_processing\_timestamp), error\_message = quote\_nullable(i\_error\_message) WHERE <custom\_conditions>; |
| unique\_etl\_record\_exists | SELECT 1 FROM partner\_order.etl\_record WHERE <unique\_conditions> LIMIT 1; |
| get\_etl\_record\_by\_id | SELECT t.etl\_record\_sno, t.etl\_job\_sno, t.staging\_table\_name, t.row\_id, t.processing\_status, t.processing\_timestamp, t.error\_message, f\_etl\_job.file\_name FROM partner\_order.etl\_record t LEFT JOIN partner\_order.etl\_job f\_etl\_job ON t.etl\_job\_sno = f\_etl\_job.etl\_job\_sno WHERE t.etl\_record\_sno = quote\_nullable(i\_etl\_record\_sno); |
| delete\_etl\_record\_by\_id | DELETE FROM partner\_order.etl\_record WHERE etl\_record\_sno = quote\_nullable(i\_etl\_record\_sno); |
| get\_all\_etl\_records | SELECT t.etl\_record\_sno, t.etl\_job\_sno, t.staging\_table\_name, t.row\_id, t.processing\_status, t.processing\_timestamp, t.error\_message, f\_etl\_job.file\_name FROM partner\_order.etl\_record t LEFT JOIN partner\_order.etl\_job f\_etl\_job ON t.etl\_job\_sno = f\_etl\_job.etl\_job\_sno ; |
| update\_etl\_record | UPDATE partner\_order.etl\_record SET etl\_record\_sno = quote\_nullable(i\_etl\_record\_sno), etl\_job\_sno = quote\_nullable(i\_etl\_job\_sno), staging\_table\_name = quote\_nullable(i\_staging\_table\_name), row\_id = quote\_nullable(i\_row\_id), processing\_status = quote\_nullable(i\_processing\_status), processing\_timestamp = quote\_nullable(i\_processing\_timestamp), error\_message = quote\_nullable(i\_error\_message) WHERE etl\_record\_sno = quote\_nullable(i\_etl\_record\_sno); |
| custom\_search\_etl\_record\_by\_id | SELECT t.etl\_record\_sno, t.etl\_job\_sno, t.staging\_table\_name, t.row\_id, t.processing\_status, t.processing\_timestamp, t.error\_message, f\_etl\_job.file\_name FROM partner\_order.etl\_record t LEFT JOIN partner\_order.etl\_job f\_etl\_job ON t.etl\_job\_sno = f\_etl\_job.etl\_job\_sno WHERE <custom\_search\_conditions>; |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Swiggy Staging**

### **Database Table Name: swiggy\_staging**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| staging\_sno | integer | swiggy\_staging\_pkey (PRIMARY KEY), |
| qbox\_remote\_location | character varying | null |
| restaurant\_code | character varying | null |
| partner\_code | character varying | null |
| order\_status\_cd | integer | null |
| order\_by | integer | null |
| partner\_purchase\_order\_id | character varying | null |
| sku\_unique\_code | character varying | swiggy\_staging\_sku\_unique\_code\_key (UNIQUE), |
| created\_time | timestamp without time zone | null |
| etl\_job\_sno | integer | swiggy\_staging\_etl\_job\_sno\_fkey (FOREIGN KEY), |
| partner\_food\_code | character varying | null |
| meal\_time\_code | character varying | null |
| transaction\_date | date | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Staging Sno | hidden | number | true | Manual Entry |
| Qbox Remote Location | textbox | string | false | Manual Entry |
| Restaurant Code | textbox | string | false | Manual Entry |
| Partner Code | textbox | string | false | Manual Entry |
| Order Status Cd | textbox | number | false | Manual Entry |
| Order By | textbox | number | false | Manual Entry |
| Partner Purchase Order Id | textbox | string | false | Manual Entry |
| Sku Unique Code | textbox | string | false | Manual Entry |
| Created Time | textbox | string | false | Manual Entry |
| Etl Job Sno | listbox | number | true | API feed list box selection |
| Partner Food Code | textbox | string | false | Manual Entry |
| Meal Time Code | textbox | string | false | Manual Entry |
| Transaction Date | timestamp | string | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_swiggy\_stagings | get | /api/v1/get\_swiggy\_stagings |
| create\_swiggy\_staging | post | /api/v1/create\_swiggy\_staging |
| update\_swiggy\_staging | put | /api/v1/update\_swiggy\_staging |
| delete\_swiggy\_staging | delete | /api/v1/delete\_swiggy\_staging |
| get\_etl\_jobs | get | /api/v1/get\_etl\_jobs |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_swiggy\_stagings | {} | {"data":{"transaction\_date":"date","created\_time":"timestamp without time zone","sku\_unique\_code":"character varying","order\_status\_cd":"integer","qbox\_remote\_location":"character varying","etl\_job\_sno":"integer","restaurant\_code":"character varying","staging\_sno":"integer","meal\_time\_code":"character varying","order\_by":"integer","partner\_food\_code":"character varying","partner\_purchase\_order\_id":"character varying","partner\_code":"character varying"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_swiggy\_staging | {"data":{"transaction\_date":"date","created\_time":"timestamp without time zone","sku\_unique\_code":"character varying","order\_status\_cd":"integer","qbox\_remote\_location":"character varying","etl\_job\_sno":"integer","restaurant\_code":"character varying","staging\_sno":"integer","meal\_time\_code":"character varying","order\_by":"integer","partner\_food\_code":"character varying","partner\_purchase\_order\_id":"character varying","partner\_code":"character varying"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_swiggy\_staging | {"data":{"transaction\_date":"date","created\_time":"timestamp without time zone","sku\_unique\_code":"character varying","order\_status\_cd":"integer","qbox\_remote\_location":"character varying","etl\_job\_sno":"integer","restaurant\_code":"character varying","staging\_sno":"integer","meal\_time\_code":"character varying","order\_by":"integer","partner\_food\_code":"character varying","partner\_purchase\_order\_id":"character varying","partner\_code":"character varying"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_swiggy\_staging | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_etl\_jobs | {"id":"integer"} | {"data":"array of etl\_job"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| custom\_delete\_swiggy\_staging\_by\_id | DELETE FROM partner\_order.swiggy\_staging WHERE <custom\_conditions>; |
| custom\_search\_swiggy\_staging\_by\_id | SELECT t.staging\_sno, t.qbox\_remote\_location, t.restaurant\_code, t.partner\_code, t.order\_status\_cd, t.order\_by, t.partner\_purchase\_order\_id, t.sku\_unique\_code, t.created\_time, t.etl\_job\_sno, t.partner\_food\_code, t.meal\_time\_code, t.transaction\_date, f\_etl\_job.file\_name FROM partner\_order.swiggy\_staging t LEFT JOIN partner\_order.etl\_job f\_etl\_job ON t.etl\_job\_sno = f\_etl\_job.etl\_job\_sno WHERE <custom\_search\_conditions>; |
| delete\_swiggy\_staging\_by\_id | DELETE FROM partner\_order.swiggy\_staging WHERE staging\_sno = quote\_nullable(i\_staging\_sno); |
| update\_swiggy\_staging | UPDATE partner\_order.swiggy\_staging SET staging\_sno = quote\_nullable(i\_staging\_sno), qbox\_remote\_location = quote\_nullable(i\_qbox\_remote\_location), restaurant\_code = quote\_nullable(i\_restaurant\_code), partner\_code = quote\_nullable(i\_partner\_code), order\_status\_cd = quote\_nullable(i\_order\_status\_cd), order\_by = quote\_nullable(i\_order\_by), partner\_purchase\_order\_id = quote\_nullable(i\_partner\_purchase\_order\_id), sku\_unique\_code = quote\_nullable(i\_sku\_unique\_code), created\_time = quote\_nullable(i\_created\_time), etl\_job\_sno = quote\_nullable(i\_etl\_job\_sno), partner\_food\_code = quote\_nullable(i\_partner\_food\_code), meal\_time\_code = quote\_nullable(i\_meal\_time\_code), transaction\_date = quote\_nullable(i\_transaction\_date) WHERE staging\_sno = quote\_nullable(i\_staging\_sno); |
| custom\_update\_swiggy\_staging | UPDATE partner\_order.swiggy\_staging SET staging\_sno = quote\_nullable(i\_staging\_sno), qbox\_remote\_location = quote\_nullable(i\_qbox\_remote\_location), restaurant\_code = quote\_nullable(i\_restaurant\_code), partner\_code = quote\_nullable(i\_partner\_code), order\_status\_cd = quote\_nullable(i\_order\_status\_cd), order\_by = quote\_nullable(i\_order\_by), partner\_purchase\_order\_id = quote\_nullable(i\_partner\_purchase\_order\_id), sku\_unique\_code = quote\_nullable(i\_sku\_unique\_code), created\_time = quote\_nullable(i\_created\_time), etl\_job\_sno = quote\_nullable(i\_etl\_job\_sno), partner\_food\_code = quote\_nullable(i\_partner\_food\_code), meal\_time\_code = quote\_nullable(i\_meal\_time\_code), transaction\_date = quote\_nullable(i\_transaction\_date) WHERE <custom\_conditions>; |
| get\_all\_swiggy\_stagings | SELECT t.staging\_sno, t.qbox\_remote\_location, t.restaurant\_code, t.partner\_code, t.order\_status\_cd, t.order\_by, t.partner\_purchase\_order\_id, t.sku\_unique\_code, t.created\_time, t.etl\_job\_sno, t.partner\_food\_code, t.meal\_time\_code, t.transaction\_date, f\_etl\_job.file\_name FROM partner\_order.swiggy\_staging t LEFT JOIN partner\_order.etl\_job f\_etl\_job ON t.etl\_job\_sno = f\_etl\_job.etl\_job\_sno ; |
| create\_swiggy\_staging | INSERT INTO partner\_order.swiggy\_staging (t.staging\_sno, t.qbox\_remote\_location, t.restaurant\_code, t.partner\_code, t.order\_status\_cd, t.order\_by, t.partner\_purchase\_order\_id, t.sku\_unique\_code, t.created\_time, t.etl\_job\_sno, t.partner\_food\_code, t.meal\_time\_code, t.transaction\_date) VALUES (i\_staging\_sno, i\_qbox\_remote\_location, i\_restaurant\_code, i\_partner\_code, i\_order\_status\_cd, i\_order\_by, i\_partner\_purchase\_order\_id, i\_sku\_unique\_code, i\_created\_time, i\_etl\_job\_sno, i\_partner\_food\_code, i\_meal\_time\_code, i\_transaction\_date); |
| unique\_swiggy\_staging\_exists | SELECT 1 FROM partner\_order.swiggy\_staging WHERE <unique\_conditions> LIMIT 1; |
| get\_swiggy\_staging\_by\_id | SELECT t.staging\_sno, t.qbox\_remote\_location, t.restaurant\_code, t.partner\_code, t.order\_status\_cd, t.order\_by, t.partner\_purchase\_order\_id, t.sku\_unique\_code, t.created\_time, t.etl\_job\_sno, t.partner\_food\_code, t.meal\_time\_code, t.transaction\_date, f\_etl\_job.file\_name FROM partner\_order.swiggy\_staging t LEFT JOIN partner\_order.etl\_job f\_etl\_job ON t.etl\_job\_sno = f\_etl\_job.etl\_job\_sno WHERE t.staging\_sno = quote\_nullable(i\_staging\_sno); |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |

## **Screen Name: Zomato Staging**

### **Database Table Name: zomato\_staging**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| staging\_sno | integer | zomato\_staging\_pkey (PRIMARY KEY), |
| qbox\_remote\_location | character varying | null |
| restaurant\_code | character varying | null |
| partner\_code | character varying | null |
| order\_status\_cd | integer | null |
| order\_by | integer | null |
| partner\_purchase\_order\_id | character varying | null |
| sku\_unique\_code | character varying | zomato\_staging\_sku\_unique\_code\_key (UNIQUE), |
| created\_time | timestamp without time zone | null |
| etl\_job\_sno | integer | zomato\_staging\_etl\_job\_sno\_fkey (FOREIGN KEY), |
| partner\_food\_code | character varying | null |
| meal\_time\_code | character varying | null |
| transaction\_date | date | null |

### **Screen Input Details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Control Name | Control Type | Data Type | Mandatory | Data Source |
| Staging Sno | hidden | number | true | Manual Entry |
| Qbox Remote Location | textbox | string | false | Manual Entry |
| Restaurant Code | textbox | string | false | Manual Entry |
| Partner Code | textbox | string | false | Manual Entry |
| Order Status Cd | textbox | number | false | Manual Entry |
| Order By | textbox | number | false | Manual Entry |
| Partner Purchase Order Id | textbox | string | false | Manual Entry |
| Sku Unique Code | textbox | string | false | Manual Entry |
| Created Time | textbox | string | false | Manual Entry |
| Etl Job Sno | listbox | number | true | API feed list box selection |
| Partner Food Code | textbox | string | false | Manual Entry |
| Meal Time Code | textbox | string | false | Manual Entry |
| Transaction Date | timestamp | string | false | Manual Entry |

### **Screen API Details:**

|  |  |  |
| --- | --- | --- |
| API Name | Method | API URI |
| get\_zomato\_stagings | get | /api/v1/get\_zomato\_stagings |
| create\_zomato\_staging | post | /api/v1/create\_zomato\_staging |
| update\_zomato\_staging | put | /api/v1/update\_zomato\_staging |
| delete\_zomato\_staging | delete | /api/v1/delete\_zomato\_staging |
| get\_etl\_jobs | get | /api/v1/get\_etl\_jobs |

### 

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Request Payload | Response Payload | Request Headers |
| get\_zomato\_stagings | {} | {"data":{"transaction\_date":"date","created\_time":"timestamp without time zone","sku\_unique\_code":"character varying","order\_status\_cd":"integer","qbox\_remote\_location":"character varying","etl\_job\_sno":"integer","restaurant\_code":"character varying","staging\_sno":"integer","meal\_time\_code":"character varying","order\_by":"integer","partner\_food\_code":"character varying","partner\_purchase\_order\_id":"character varying","partner\_code":"character varying"}} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| create\_zomato\_staging | {"data":{"transaction\_date":"date","created\_time":"timestamp without time zone","sku\_unique\_code":"character varying","order\_status\_cd":"integer","qbox\_remote\_location":"character varying","etl\_job\_sno":"integer","restaurant\_code":"character varying","staging\_sno":"integer","meal\_time\_code":"character varying","order\_by":"integer","partner\_food\_code":"character varying","partner\_purchase\_order\_id":"character varying","partner\_code":"character varying"}} | {"message":"Record created","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| update\_zomato\_staging | {"data":{"transaction\_date":"date","created\_time":"timestamp without time zone","sku\_unique\_code":"character varying","order\_status\_cd":"integer","qbox\_remote\_location":"character varying","etl\_job\_sno":"integer","restaurant\_code":"character varying","staging\_sno":"integer","meal\_time\_code":"character varying","order\_by":"integer","partner\_food\_code":"character varying","partner\_purchase\_order\_id":"character varying","partner\_code":"character varying"}} | {"message":"Record updated","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| delete\_zomato\_staging | {"id":"integer"} | {"message":"Record deleted","status":"success"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |
| get\_etl\_jobs | {"id":"integer"} | {"data":"array of etl\_job"} | {"Authorization":"Bearer <token>","Accept":"application/json","X-Request-ID":"unique-request-id","User-Agent":"MyApp/1.0.0","Content-Type":"application/json","X-API-Version":"v1"} |

### **Database Script Details:**

|  |  |
| --- | --- |
| Script Name | Database Script |
| unique\_zomato\_staging\_exists | SELECT 1 FROM partner\_order.zomato\_staging WHERE <unique\_conditions> LIMIT 1; |
| get\_zomato\_staging\_by\_id | SELECT t.staging\_sno, t.qbox\_remote\_location, t.restaurant\_code, t.partner\_code, t.order\_status\_cd, t.order\_by, t.partner\_purchase\_order\_id, t.sku\_unique\_code, t.created\_time, t.etl\_job\_sno, t.partner\_food\_code, t.meal\_time\_code, t.transaction\_date, f\_etl\_job.file\_name FROM partner\_order.zomato\_staging t LEFT JOIN partner\_order.etl\_job f\_etl\_job ON t.etl\_job\_sno = f\_etl\_job.etl\_job\_sno WHERE t.staging\_sno = quote\_nullable(i\_staging\_sno); |
| update\_zomato\_staging | UPDATE partner\_order.zomato\_staging SET staging\_sno = quote\_nullable(i\_staging\_sno), qbox\_remote\_location = quote\_nullable(i\_qbox\_remote\_location), restaurant\_code = quote\_nullable(i\_restaurant\_code), partner\_code = quote\_nullable(i\_partner\_code), order\_status\_cd = quote\_nullable(i\_order\_status\_cd), order\_by = quote\_nullable(i\_order\_by), partner\_purchase\_order\_id = quote\_nullable(i\_partner\_purchase\_order\_id), sku\_unique\_code = quote\_nullable(i\_sku\_unique\_code), created\_time = quote\_nullable(i\_created\_time), etl\_job\_sno = quote\_nullable(i\_etl\_job\_sno), partner\_food\_code = quote\_nullable(i\_partner\_food\_code), meal\_time\_code = quote\_nullable(i\_meal\_time\_code), transaction\_date = quote\_nullable(i\_transaction\_date) WHERE staging\_sno = quote\_nullable(i\_staging\_sno); |
| get\_all\_zomato\_stagings | SELECT t.staging\_sno, t.qbox\_remote\_location, t.restaurant\_code, t.partner\_code, t.order\_status\_cd, t.order\_by, t.partner\_purchase\_order\_id, t.sku\_unique\_code, t.created\_time, t.etl\_job\_sno, t.partner\_food\_code, t.meal\_time\_code, t.transaction\_date, f\_etl\_job.file\_name FROM partner\_order.zomato\_staging t LEFT JOIN partner\_order.etl\_job f\_etl\_job ON t.etl\_job\_sno = f\_etl\_job.etl\_job\_sno ; |
| custom\_search\_zomato\_staging\_by\_id | SELECT t.staging\_sno, t.qbox\_remote\_location, t.restaurant\_code, t.partner\_code, t.order\_status\_cd, t.order\_by, t.partner\_purchase\_order\_id, t.sku\_unique\_code, t.created\_time, t.etl\_job\_sno, t.partner\_food\_code, t.meal\_time\_code, t.transaction\_date, f\_etl\_job.file\_name FROM partner\_order.zomato\_staging t LEFT JOIN partner\_order.etl\_job f\_etl\_job ON t.etl\_job\_sno = f\_etl\_job.etl\_job\_sno WHERE <custom\_search\_conditions>; |
| custom\_update\_zomato\_staging | UPDATE partner\_order.zomato\_staging SET staging\_sno = quote\_nullable(i\_staging\_sno), qbox\_remote\_location = quote\_nullable(i\_qbox\_remote\_location), restaurant\_code = quote\_nullable(i\_restaurant\_code), partner\_code = quote\_nullable(i\_partner\_code), order\_status\_cd = quote\_nullable(i\_order\_status\_cd), order\_by = quote\_nullable(i\_order\_by), partner\_purchase\_order\_id = quote\_nullable(i\_partner\_purchase\_order\_id), sku\_unique\_code = quote\_nullable(i\_sku\_unique\_code), created\_time = quote\_nullable(i\_created\_time), etl\_job\_sno = quote\_nullable(i\_etl\_job\_sno), partner\_food\_code = quote\_nullable(i\_partner\_food\_code), meal\_time\_code = quote\_nullable(i\_meal\_time\_code), transaction\_date = quote\_nullable(i\_transaction\_date) WHERE <custom\_conditions>; |
| create\_zomato\_staging | INSERT INTO partner\_order.zomato\_staging (t.staging\_sno, t.qbox\_remote\_location, t.restaurant\_code, t.partner\_code, t.order\_status\_cd, t.order\_by, t.partner\_purchase\_order\_id, t.sku\_unique\_code, t.created\_time, t.etl\_job\_sno, t.partner\_food\_code, t.meal\_time\_code, t.transaction\_date) VALUES (i\_staging\_sno, i\_qbox\_remote\_location, i\_restaurant\_code, i\_partner\_code, i\_order\_status\_cd, i\_order\_by, i\_partner\_purchase\_order\_id, i\_sku\_unique\_code, i\_created\_time, i\_etl\_job\_sno, i\_partner\_food\_code, i\_meal\_time\_code, i\_transaction\_date); |
| custom\_delete\_zomato\_staging\_by\_id | DELETE FROM partner\_order.zomato\_staging WHERE <custom\_conditions>; |
| delete\_zomato\_staging\_by\_id | DELETE FROM partner\_order.zomato\_staging WHERE staging\_sno = quote\_nullable(i\_staging\_sno); |

### **Security Details:**

|  |  |
| --- | --- |
| Security Area | Details |
| Authorization | Roles : [Admin] , [Super Admin] , [System Admin] |
| Authentication | Bearer Token received upon succesful login passed as Request Header parameter in each API. |