4Sight-Server_APIs-v1.3-R

```
swagger: '2.0'
info:
  description: API Documentation
  version: 1.3.0
  title: 4Sight Swagger v1.3
host: localhost
basePath: /4sight/api
tags:
  - name: Login
    description: APIs to manage user Login and Logout
  - name: Widgets
    description: APIs to manage Widgets
  - name: Dashboards
    description: APIs to manage Dashboards
  - name: Search
    description: APIs for Search feature
  - name: Predictions
    description: APIs to manage Prediction models
  - name: Alerts
    description: APIs to manage Alerts
  - name: Datasets
    description: APIs to manage Datasets
  - name: Settings
    description: APIs to manage Settings
schemes:
  - http
paths:
  /isloggedin:
   get:
      tags:
        - Login
      summary: Login checking API
      description: >-
        Login checking api
      produces:
          - application/json
      parameters:
        - name: realmname
          in: query
          description: account name
          required: true
          type: string
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              is_loggedin:
                type: boolean
              client_id:
                type: string
              token:
                type: string
              account:
                type: string
              loginurl:
                type: string


  /gettoken:
   get:
      tags:
        - Login
      summary: Token
      description: >-
        To get token
      produces:
          - application/json
      parameters:
        - name: realmname
          in: query
          description: account name
          required: true
          type: string
        - name: client_id
          in: query
          description: clientID
          required: true
          type: string
        - name: code
          in: query
          description: code of an application
          required: true
          type: string
      responses:
        '400':
          description: Invalid request
          schema:
            type: object
            properties:
              message:
                type: string
                example: "Invalid request"
        '500':
          description: SQL Alchemy Error
          schema:
            type: object
            properties:
              message:
                type: string
                example: "SQL Alchemy error"
        '200':
          description: Sucess
          schema:
            type: object
            properties:
              message:
                type: string

  /logout:
    get:
        tags:
          - Login
        summary: Logout API
        description: >-
          Logout API
        produces:
            - application/json
        parameters:
          - name: token
            in: header
            description: Session Token
            required: true
            type: string
        responses:
          '204':
            description: Success
            schema:
              type: object
              properties:
                message:
                  type: string
                  example: Logout successful
          '401':
            description: >-
              Unauthorized. Invalid Login
            schema:
              type: object
              properties:
                message:
                  type: string
                  example: "Invalid login credentials"

  /settings/saveFooter:
    post:
      tags:
```

```
  - Settings
summary: Save Footer
description: >-
  API to save footer
consumes:
  - application/json
produces:
  - application/json
parameters:
  - name: token
    in: header
    description: Session Token
    required: true
    type: string
  - name: body
    in: body
    description: Request Body
    required: true
    schema:
      type: object
      properties:
        is_footer_enabled:
          type: boolean
        footer_content:
          type: string

responses:
  '201':
    description: Success
    schema:
      type: object
      properties:
        message:
          type: string
          example: Success
  '401':
    description: >-
      Unauthorized. Invalid Login Credentials
    schema:
      $ref: '#/definitions/InvalidLoginResponse'
  '400':
    description: >-
      Bad Request. Invalid Request Body
    schema:
      $ref: '#/definitions/InvalidRequestBody'
  '500':
    description: SQL Alchemy Error
    schema:
      type: object
      properties:
              message:
                type: string
  /settings/saveTheme:
    post:
      tags:
        - Settings
      summary: Save Theme
      description: >-
        API to save theme
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: body
          in: body
          description: Request Body
          required: true
          schema:
            type: object
            properties:
              theme_color:
                type: string
              selectedicon_color:
                type: string
              inselectedicon_color:
                type: string
              buttons_color:
                type: string
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              message:
                type: string
                example: Success
        '401':
          description: >-
            Unauthorized. Invalid Login Credentials
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
```

```
            Bad Request. Invalid Request Body
          schema:
            $ref: '#/definitions/InvalidRequestBody'
        '500':
          description: SQL Alchemy Error
          schema:
            type: object
            properties:
              message:
                type: string

  /settings/saveclientsecrets:
    post:
      tags:
        - Settings
      summary: Save client secrets
      description: >-
        API to save client secrets
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: account
          in: query
          description: account name
          required: true
          type: string
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              message:
                type: string
                example: Success
        '401':
          description: >-
            Unauthorized. Invalid Login Credentials
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid Request Body
          schema:
            $ref: '#/definitions/InvalidRequestBody'
        '500':
          description: SQL Alchemy Error
          schema:
            type: object
            properties:
              message:
                type: string

  /settings/saveLogo:
    post:
      tags:
        - Settings
      summary: Save Logo
      description: >-
        API to save logo
      consumes:
         - multipart/form-data
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: logo
          in: formData
          type: file
          description: The file to upload
          required: true
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              message:
                type: string
                example: Success
        '401':
          description: >-
            Unauthorized. Invalid Login Credentials
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request.
          schema:
            $ref: '#/definitions/InvalidRequestBody'
        '500':
          description: "Failed to upload Logo"
          schema:
            type: object
            properties:
              message:
                type: string
  /settings/saveRegex:
    post:
      tags:
        - Settings
      summary: Save Regex
      description: >-
        API to save regex
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: body
          in: body
          description: Request Body
          required: true
          schema:
            type: object
            properties:
              regex:
                type: string

      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              message:
                type: string
                example: Success
        '401':
          description: >-
            Unauthorized. Invalid Login Credentials
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid Request Body
          schema:
            $ref: '#/definitions/InvalidRequestBody'
        '500':
          description: SQL Alchemy Error
          schema:
            type: object
            properties:
              message:
                type: string
  /settings/getFooter:
    get:
      tags:
        - Settings
      summary: Get footer details
      description: >-
        API to get footer details
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              footer_settings:
                type: object
              is_footer_enabled:
                type: boolean
              footer_content:
                type: string
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
  /settings/getRegex:
    get:
      tags:
        - Settings
      summary: Get Regex details
      description: >-
        API to get regex details
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              regex:
                type: string
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
  /settings/getTheme:
    get:
      tags:
        - Settings
      summary: Get theme details
      description: >-
        API to get theme details
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              theme_settings:
                type: object
              theme_color:
                type: string
              selectedicon_color:
                type: string
              inselectedicon_color:
                type: string
              buttons_color:
                type: string
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
  /settings/getLogo:
    get:
      tags:
        - Settings
      summary: Get logo image
      description: >-
        API to get logo image
      produces:
        - image/*
      parameters:
        - name: accountName
          in: query
          description: account name
          required: true
          type: string
      responses:
        '200':
          description: the value is 'attachment;filename=4sight_logo.
png'
          schema:
            type: file
          headers:
            content-Disposition:
              type: string
        '400':
          description: Bad Request
          schema:
            type: object
            properties:
              message:
                type: string
                example: Invalid Query Parameters

  /dataset/createDataset:
    post:
      tags:
        - Datasets
      summary: Create new Dataset
      description: >-
        API to create new Dataset
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: body
          in: body
          description: Request Body
          required: true
          schema:
            type: object
            properties:
              index_name:
                type: string
              dataset_name:
                type: string
              description:
                type: string
              default_timestamp:
                type: string
              filters:
                type: array
                items:
                  type: string
              dataset_columns:
                type: array
                items:
                  type: object
                  properties:
                    column_name:
                      type: string
                    alias:
                      type: string
      responses:
        '201':
          description: Success
          schema:
            type: object
            properties:
              message:
                type: string
                example: Created
              dataset_id:
                type: number
                description: Created Dataset Id
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid Request Body
          schema:
            $ref: '#/definitions/InvalidRequestBody'
        '500':
          description: SQLAlchemyError
          schema:
            type: object
            properties:
              message:
                type: string
  /dataset/updateDataset:
    put:
      tags:
        - Datasets
      summary: Update Dataset
      description: >-
        API to update Dataset
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: datasetId
          in: query
          description: Datset Id to update
          required: true
          type: number
        - name: body
          in: body
          description: Request Body
          required: true
          schema:
            type: object
            properties:
              index_name:
                type: string
              dataset_name:
                type: string
              description:
                type: string
              default_timestamp:
                type: string
              filters:
                type: array
                items:
                  type: string
              dataset_columns:
                type: array
                items:
                  type: object
                  properties:
                    column_name:
                      type: string
                    alias:
                      type: string
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              message:
                type: string
                example: Updated
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid Request Body, Invalid request format,
Invalid dataset id
          schema:
            type: object
            properties:
              message:
                type: string
                example: Invalid Request Body
        '500':
          description: SQLAlchemyError
          schema:
            type: object
            properties:
              message:
                type: string
  /dataset/getDatasetCount:
    get:
      tags:
        - Datasets
      summary: Get count of Datasets
      description: >-
        API to get count of Datasets
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: searchKeyword
          in: query
          description: Search Keyword
          type: string
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              datasets_count:
                type: number
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '500':
          description: SQL Alchemy Error
          schema:
            type: object
            properties:
              message:
                type: string
  /dataset/deleteDataset:
    delete:
      tags:
        - Datasets
      summary: Delete Dataset
      description: >-
        API to delete existing dataset
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: datasetId
          in: query
          description: Dataset Id to be deleted
          required: true
          type: number
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              status:
                type: string
                example: Deleted
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Integrity Error, Bad Request. Invalid request format,
Invalid dataset id
          schema:
            type: object
            properties:
              message:
                type: string
                example: Unable to delete dataset. Dataset is in use
        '500':
          description: SQLAlchemyError
          schema:
            type: object
            properties:
              message:
                type: string

  /dataset/deleteMultipleDataset:
    delete:
      tags:
        - Datasets
      summary: Delete Dataset
      description: >-
        API to Delete dataset
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: body
          in: body
          description: Request Body
          required: true
          schema:
            type: object
            properties:
              dataset_ids:
                type: array
                items:
                  type: number
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              status:
                type: string
                example: Deleted
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '500':
          description: SQLAlchemyError
          schema:
            type: object
            properties:
              message:
                type: string

  /dataset/getDatasetDetailsById:
    get:
      tags:
        - Datasets
      summary: Get Dataset details
      description: >-
        API to get dataset details by dataset id
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: datasetId
          in: query
          description: Dataset Id
          required: true
          type: number
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              index:
                type: string
              dataset_name:
                type: string
              dataset_desc:
                type: string
              default_timestamp:
                type: string
              filters:
                type: array
                items:
                  type: string
              columns:
                type: array
                items:
                  type: object
                  properties:
                    column_id:
                      type: number
                    column_name:
                      type: string
                    alias:
                      type: string
                    isColumnInUse:
                      type: boolean
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid request format, Invalid dataset id
          schema:
            type: object
            properties:
              message:
                type: string
                example: Invalid request format
  /dataset/getDatasets:
    get:
      tags:
        - Datasets
      summary: Get Datasets
      description: >-
```

```
        API to get list of datasets
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: sortBy
          in: query
          description: Sort By Term. Example :- time or name
          type: string
        - name: sortOrder
          in: query
          description: Sort order. Example :- asc or desc
          type: string
        - name: from
          in: query
          description: Starting Offset. Note :- For first datset value
will be 0
          type: number
        - name: size
          in: query
          description: Number of datasets to fetch. If -1 is passed
then all Datsets will be fetched.
          type: number
        - name: searchKeyword
          in: query
          description: Search keyword
          type: string
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              datasets:
                type: array
                items:
                  type: object
                  properties:
                    dataset_id:
                      type: number
                    index_name:
                      type: string
                    dataset_name:
                      type: string
                    description:
                      type: string
                    default_timestamp:
                      type: string
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
  /dataset/getDatasetColumnsById:
    get:
      tags:
        - Datasets
      summary: Get Datasets Columns details
      description: >-
        API to get column details of a Dataset
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: datasetId
          in: query
          description: Dataset Id
          required: true
          type: number
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              index:
                type: string
              selectedTimeColumn:
                type: string
              columns:
                type: array
                items:
                  type: object
                  properties:
                    column_id:
                      type: number
                    column_name:
                      type: string
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid request format, Invalid dataset id
          schema:
            type: object
            properties:
              message:
                type: string
                example: Invalid request format

  /dataset/search:
    get:
      tags:
        - Datasets
      summary: Search Datasets by name
      description: >-
        API to search datasets by name
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: keyword
          in: query
          description: Search keyword
          required: true
          type: string
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              result:
                type: array
                items:
                  type: object
                  properties:
                    dataset_id:
                      type: number
                    dataset_name:
                      type: string
                    index_name:
                      type: string
                    description:
                      type: string
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '500':
          description: SQLAlchemyError
          schema:
            type: object
            properties:
              message:
                type: string
  /dataset/getIndexColumns:
    get:
      tags:
        - Datasets
      summary: Get Index Columns
      description: >-
        API to get index columns
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: index
          in: query
          description: Index Name
          required: true
          type: string
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              columns:
                type: array
                items:
                  type: string
              time_columns:
                type: array
                items:
                  type: string
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '500':
          description: NameError
          schema:
            type: object
            properties:
              message:
                type: string

  /widget/createWidget:
    post:
      tags:
        - Widgets
      summary: Create widget
      description: >-
        API to create widget
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: body
          in: body
          description: Request Body
          required: true
          schema:
            type: object
            properties:
              widget_name:
                type: string
              dataset_id:
                type: number
              filters:
                type: array
                items:
                  type: string
              dimensions:
                type: array
                items:
                  type: object
                  properties:
                    column_id:
                      type: number
              metrics:
                type: array
                items:
                  type: object
                  properties:
                    column_id:
                      type: number
                    aggregation_function:
                      type: string
                    custom_label:
                      type: string
                    filter:
                      type: array
                      items:
                        type: string
              scripts:
                type: array
                items:
                  type: object
                  properties:
                    script:
                      type: string
                    custom_label:
                      type: string
              is_timeseries:
                type: boolean
              start_time:
                type: string
              end_time:
                type: string
              sort_field:
                type: string
              sort_order:
                type: string
              sort_type:
                type: string
              customsort_field:
                type: string
              customsort_aggr:
                type: string
              number_of_datapoints:
                type: number
              chart_type:
                type: string
              legend_position:
                type: string
              xaxis_label:
                type: string
              yaxis_label:
                type: string
      responses:
        '201':
```

```
        description: Success
        schema:
          type: object
          properties:
            status:
              type: string
              example: Created
            widget_id:
              type: number
      '401':
        description: >-
          Unauthorized. Invalid Login
        schema:
          $ref: '#/definitions/InvalidLoginResponse'
      '400':
        description: >-
          Bad Request. Invalid Request Body
        schema:
          $ref: '#/definitions/InvalidRequestBody'
      '500':
        description: SQLAlchemy Error
        schema:
            type: object
            properties:
              message:
                type: string
/widget/previewWidget:
  post:
    tags:
      - Widgets
    summary: Preview widget
    description: >-
      API to preview widget
    consumes:
      - application/json
    produces:
      - application/json
    parameters:
      - name: token
        in: header
        description: Session Token
        required: true
        type: string
      - name: body
        in: body
        description: Request Body
        required: true
        schema:
          type: object
          properties:
```

```
dataset_id:
  type: number
filters:
  type: array
  items:
    type: string
dimensions:
  type: array
  items:
    type: object
    properties:
      column_id:
        type: number
metrics:
  type: array
  items:
    type: object
    properties:
      column_id:
        type: number
      aggregation_function:
        type: string
      custom_label:
        type: string
      filter:
        type: array
        items:
          type: string
scripts:
  type: array
  items:
    type: object
    properties:
      script:
        type: string
      custom_label:
        type: string
is_timeseries:
  type: boolean
start_time:
  type: string
end_time:
  type: string
sort_field:
  type: string
sort_order:
  type: string
sort_type:
  type: string
customsort_field:
```

```
                type: string
              customsort_aggr:
                type: string
              number_of_datapoints:
                type: number
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              data:
                type: object
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid Request Body
          schema:
            $ref: '#/definitions/InvalidRequestBody'
        '500':
          description: keyError
          schema:
            type: object
            properties:
              message:
                type: string
  /widget/updateWidget:
    put:
      tags:
        - Widgets
      summary: Update widget
      description: >-
        API to update widget details
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: widgetId
          in: query
          description: Widget Id
          required: true
          type: number
        - name: body
          in: body
          description: Request Body
          required: true
          schema:
            type: object
            properties:
              widget_name:
                type: string
              dataset_id:
                type: number
              filters:
                type: array
                items:
                  type: string
              dimensions:
                type: array
                items:
                  type: object
                  properties:
                    column_id:
                      type: number
              metrics:
                type: array
                items:
                  type: object
                  properties:
                    column_id:
                      type: number
                    aggregation_function:
                      type: string
                    custom_label:
                      type: string
                    filter:
                      type: array
                      items:
                        type: string
              scripts:
                type: array
                items:
                  type: object
                  properties:
                    script:
                      type: string
                    custom_label:
                      type: string
              is_timeseries:
                type: boolean
```

```
              start_time:
                type: string
              end_time:
                type: string
              sort_field:
                type: string
              sort_order:
                type: string
              sort_type:
                type: string
              customsort_field:
                type: string
              customsort_aggr:
                type: string
              number_of_datapoints:
                type: number
              chart_type:
                type: string
              legend_position:
                type: string
              xaxis_label:
                type: string
              yaxis_label:
                type: string
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              status:
                type: string
                example: Updated
              widget_id:
                type: number
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid Request Body, Invalid request format,
Invalid widegt id
          schema:
            type: object
            properties:
              message:
                type: string
                example: Invalid request format
        '500':
          description: SQLAlchemy Error
          schema:
              type: object
              properties:
                message:
                  type: string
  /widget/getWidgets:
    get:
      tags:
        - Widgets
      summary: Get widgets list
      description: >-
        API to get widgets list
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: sortBy
          in: query
          description: Sort By Term. Example :- time or name
          type: string
        - name: sortOrder
          in: query
          description: Sort order. Example :- asc or desc
          type: string
        - name: from
          in: query
          description: Starting Offset. Note :- For first widget value
will be 0
          type: number
        - name: size
          in: query
          description: Number of widgets to fetch. If -1 is passed then
all widgets will be fetched.
          type: number
        - name: searchKeyword
          in: query
          description: Search keyword
          type: string
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              widget_details:
                type: object
                properties:
                  widget_name:
                    type: string
                  widget_id:
                    type: number
                  created_by:
                    type: number
                  dataset_id:
                    type: number
                  dimensions:
                    type: array
                    items:
                      type: string
                  metrics:
                    type: array
                    items:
                      type: string
                  is_timeSeries:
                    type: boolean
                  start_time:
                    type: string
                  end_time:
                    type: string
                  sort_field:
                    type: string
                  sort_order:
                    type: string
                  chart_type:
                    type: string
                  associated_dashboards:
                    type: array
                    items:
                      type: string
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
  /widget/getWidgetDetails:
    get:
      tags:
        - Widgets
      summary: Get widget details by Id
      description: >-
        API to get widget details by Id
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: widgetId
          in: query
          description: Widget Id
          required: true
          type: number
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              widget_details:
                type: object
                properties:
                  widget_name:
                    type: string
                  dataset_id:
                    type: number
                  is_timeseries:
                    type: boolean
                  start_time:
                    type: string
                  end_time:
                    type: string
                  number_of_datapoints:
                    type: number
                  chart_type:
                    type: string
                  legend_position:
                    type: string
                  xaxis_label:
                    type: string
                  yaxis_label:
                    type: string
                  sort_field:
                    type: string
                  sort_order:
                    type: string
                  sort_type:
                    type: string
                  customsort_field:
                    type: string
                  customsort_aggr:
                    type: string
                  dimensions:
```

```
                    type: array
                    items:
                      type: object
                      properties:
                        column_id:
                          type: number
                  metrics:
                    type: array
                    items:
                      type: object
                      properties:
                        column_id:
                          type: number
                        aggregation_function:
                          type: string
                        custom_label:
                          type: string
                        filter:
                          type: array
                          items:
                            type: string
                  filters:
                    type: array
                    items:
                      type: string
                  scripts:
                    type: array
                    items:
                      type: object
                      properties:
                        script:
                          type: string
                        custom_label:
                          type: string
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid request format, Invalid widegt id
          schema:
            type: object
            properties:
              message:
                type: string
                example: Invalid request format
  /widget/getGraphData:
    post:
```

```
tags:
  - Widgets
summary: Get graph data of a Widget
description: >-
  API to get graph data of a Widget
produces:
  - application/json
parameters:
  - name: token
    in: header
    description: Session Token
    required: true
    type: string
  - name: widgetId
    in: query
    description: Widget Id
    required: true
    type: number
  - name: startTime
    in: query
    description: Start Time
    required: true
    type: string
  - name: endTime
    in: query
    description: End Time
    required: true
    type: string
  - name: body
    in: body
    description: Request Body
    schema:
      type: object
      properties:
        add_filters:
          type: array
          items:
            type: string
responses:
  '200':
    description: Success
    schema:
      type: object
      properties:
        data:
          type: object
  '401':
    description: >-
      Unauthorized. Invalid Login
    schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid time range, Invalid request format,
Invalid widget id
          schema:
            type: object
            properties:
              message:
                type: string
                example: Invalid time range
  /widget/getWidgetCount:
    get:
      tags:
        - Widgets
      summary: Get count of widgets
      description: >-
        API to get count of widgets
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: searchKeyword
          in: query
          description: Search Keyword
          type: string
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              widgets_count:
                type: number
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '500':
          description: SQL Alchemy Error
          schema:
            type: object
            properties:
              message:
                type: string
  /widget/deleteWidget:
    delete:
      tags:
        - Widgets
      summary: Delete Widget based on Id
      description: >-
        API to delete Widget based on Id
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: widgetId
          in: query
          description: Widget Id
          required: true
          type: number
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              status:
                type: string
                example: Deleted
        '400':
          description: Bad Request. Inavlid widget id, Invalid request
format
          schema:
            type: object
            properties:
              message:
                type: string
                example: widget 'widgetname' is used in dashboards. So
it cannot be deleted
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '500':
          description: SQLAlchemy Error
          schema:
            type: object
            properties:
              message:
                type: string

  /widget/deleteMultipleWidget:
    delete:
      tags:
        - Widgets
      summary: Delete Widget
      description: >-
        API to Delete widget
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: body
          in: body
          description: Request Body
          required: true
          schema:
            type: object
            properties:
              widget_ids:
                type: array
                items:
                  type: number
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              status:
                type: string
                example: Deleted
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '500':
          description: SQLAlchemyError
          schema:
            type: object
            properties:
              message:
                type: string
  /widget/getAttachedDashboards:
    get:
      tags:
        - Widgets
      summary: Get attached dashboards of a Widget
      description: >-
        API to get the attached dashboards of a Widget
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: widgetId
          in: query
          description: Widget Id
          required: true
          type: number
        - name: searchKeyword
          in: query
          description: search keyword
          type: string
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              dashboards:
                type: array
                items:
                  type: string
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: Bad Request. Invalid widget id, Invalid request
format
          schema:
            type: object
            properties:
              message:
                type: string
                example: Invalid widget id
  /widget/exportAll:
    get:
      tags:
        - Widgets
      summary: Returns the requested json file as "file download" i.e.
with content-disposition = attachment
      description: >-
        API to export of widgets
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
      responses:
        '200':
          description: the value is 'attachment;filename=widgets_config.
json'
          schema:
            type: file
          headers:
            content-Disposition:
              type: string
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
  /widget/export:
    post:
      tags:
        - Widgets
      summary: Returns the requested json file as "file download" i.e.
with content-disposition = attachment
      description: >-
        API to export of widget
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: body
          in: body
          description: Request Body
          required: true
          schema:
            type: object
            properties:
              widget_ids:
                type: array
                items:
                  type: number
      responses:
        '200':
          description: the value is 'attachment;filename=widgets_config.
json'
          schema:
            type: file
          headers:
            content-Disposition:
              type: string
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid Request Body
          schema:
            $ref: '#/definitions/InvalidRequestBody'


  /widget/import:
    post:
      tags:
        - Widgets
      summary: Get import of widget
      description: >-
        API to import of widget data
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: body
          in: body
          description: Request Body
```

```
required: true
schema:
  type: array
  items:
    type: object
    properties:
      widget_name:
        type: string
      dataset:
        type: string
      is_timeseries:
        type: string
      start_time:
        type: string
      end_time:
        type: string
      sort_order:
        type: string
      sort_type:
        type: string
      sort_field:
        type: string
      customsort_field:
        type: string
      customsort_aggr:
        type: string
      number_of_datapoints:
        type: string
      chart_type:
        type: string
      legend_position:
        type: string
      xaxis_label:
        type: string
      yaxis_label:
        type: string
      dimensions:
        type: object
        properties:
          columnname:
            type: string
      metrics:
        type: object
        properties:
          columnname:
            type: string
          aggregation_function:
            type: string
          custom_label:
            type: string
```

```
                    filters:
                      type: array
                      items:
                        type: string
                filters:
                  type: array
                  items:
                    type: string
                scripts:
                  type: array
                  items:
                    type: object
                    properties:
                      script:
                        type: string
                      custom_label:
                        type: string
      responses:
        '201':
          description: Success
          schema:
            type: object
            properties:
              status:
                type: string
                example: imported
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid Configuration File, Widget name
already exists, Invalid Dataset,
                Invalid Column name
          schema:
            type: object
            properties:
              message:
                type: string
                example: Invalid Configuration File
        '500':
          description: SQLAlchemy Error
          schema:
            type: object
            properties:
              message:
                type: string
```

```
/dashboard/createDashboard:
  post:
    tags:
      - Dashboards
    summary: Create new dashboard
    description: >-
      API to create new dashboard
    produces:
      - application/json
    consumes:
      - application/json
    parameters:
      - name: token
        in: header
        description: Session Token
        required: true
        type: string
      - name: body
        in: body
        description: Request Body
        required: true
        schema:
          type: object
          properties:
            name:
              type: string
            description:
              type: string
            start_time:
              type: string
            end_time:
              type: string
            is_custom_start_time:
              type: boolean
            refresh_interval:
              type: string
            time_range_enabled:
              type: boolean
            refresh_enabled:
              type: boolean
            edit_refresh_enabled:
              type: boolean
            filter_enabled:
              type: boolean
            filters:
              type: array
              items:
                type: string
            widgets:
              type: array
              items:
                type: object
                properties:
                  widget_id:
                    type: number
                  name:
                    type: string
                  row:
                    type: number
                  column:
                    type: number
                  height:
                    type: number
                  width:
                    type: number
                  chart_type:
                    type: string
    responses:
      '201':
        description: Success
        schema:
          type: object
          properties:
            status:
              type: string
              example: Created
            dashboard_id:
              type: number
              description: Created Dashboard Id
      '401':
        description: >-
          Unauthorized. Invalid Login
        schema:
          $ref: '#/definitions/InvalidLoginResponse'
      '400':
        description: >-
          Bad Request. Invalid Request Body
        schema:
          $ref: '#/definitions/InvalidRequestBody'
      '500':
        description: SQLAlchemy Error
        schema:
          type: object
          properties:
            message:
              type: string
/dashboard/updateDashboard:
  put:
    tags:
      - Dashboards
```

```
summary: Update dashboard
description: >-
  API to update dashboard
produces:
  - application/json
consumes:
  - application/json
parameters:
  - name: token
    in: header
    description: Session Token
    required: true
    type: string
  - name : dashboardId
    in: query
    description: Dashboard Id
    required: true
    type: number
  - name: body
    in: body
    description: Request Body
    required: true
    schema:
      type: object
      properties:
        name:
          type: string
        description:
          type: string
        time_range:
          type: string
        is_custom_time_range:
          type: boolean
        refresh_interval:
          type: string
        time_range_enabled:
          type: boolean
        refresh_enabled:
          type: boolean
        edit_refresh_enabled:
          type: boolean
        filter_enabled:
          type: boolean
        filters:
          type: array
          items:
            type: string
        widgets:
          type: array
          items:
                  type: object
                  properties:
                    widget_id:
                      type: number
                    name:
                      type: string
                    row:
                      type: number
                    column:
                      type: number
                    height:
                      type: number
                    width:
                      type: number
                    chart_type:
                      type: string
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              status:
                type: string
                example: Updated
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid Request Body, Invalid request format,
Invalid dashboard id
          schema:
            type: object
            properties:
              message:
                type: string
                example: Invalid request format
        '500':
          description: SQLAlchemy Error
          schema:
            type: object
            properties:
              message:
                type: string
  /dashboard/getDashboardCount:
    get:
      tags:
        - Dashboards
      summary: Get number of dashboards
      description: >-
        API to get number of dashboards
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: searchKeyword
          in: query
          description: Search keyword
          type: string
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              dashboards_count:
                type: number
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '500':
          description: SQLAlchemy Error
          schema:
            type: object
            properties:
              message:
                type: string
  /dashboard/getDashboards:
    get:
      tags:
        - Dashboards
      summary: Get dashboards
      description: >-
        API to get dashboards
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: sortBy
          in: query
          description: Sort By Term. Example :- time or name
          type: string
        - name: sortOrder
          in: query
          description: Sort order. Example :- asc or desc
          type: string
        - name: from
          in: query
          description: Starting Offset. Note :- For first widget value
will be 0
          type: number
        - name: size
          in: query
          description: Number of widgets to fetch. If -1 is passed then
all widgets will be fetched.
          type: number
        - name: searchKeyword
          in: query
          description: Search keyword
          type: string
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              dashboard_details:
                type: object
                properties:
                  dashboard_name:
                    type: string
                  dashboard_id:
                    type: number
                  description:
                    type: string
                  last_updated:
                    type: string
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
  /dashboard/getDashboardDetails:
    get:
      tags:
        - Dashboards
      summary: Get dashboard details
```

```yaml
description: >-
  API to get dashboard details
produces:
  - application/json
parameters:
  - name: token
    in: header
    description: Session Token
    required: true
    type: string
  - name: dashboardId
    in: query
    description: Dashboard Id
    required: true
    type: number
responses:
  '200':
    description: Success
    schema:
      type: object
      properties:
        dashboard_details:
          type: object
          properties:
            name:
              type: string
            description:
              type: string
            time_range:
              type: string
            is_custom_time_range:
              type: boolean
            refresh_interval:
              type: string
            time_range_enabled:
              type: boolean
            refresh_enabled:
              type: boolean
            edit_refresh_enabled:
              type: boolean
            filter_enabled:
              type: boolean
            filters:
              type: array
              items:
                type: string
            widgets:
              type: array
              items:
                type: object
```

```
                      properties:
                        widget_id:
                          type: number
                        name:
                          type: string
                        row:
                          type: number
                        column:
                          type: number
                        height:
                          type: number
                        width:
                          type: number
                        chart_type:
                          type: string
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: Bad Request. Invalid dashboard id, Invalid
request format
          schema:
            type: object
            properties:
              message:
                type: string
                example: Invalid dashboard id
  /dashboard/deleteDashboard:
    delete:
      tags:
        - Dashboards
      summary: Delete dashboard by id
      description: >-
        API to delete dashboard by id
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: dashboardId
          in: query
          description: Dashboard Id
          required: true
          type: number
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              status:
                type: string
                example: Deleted
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: Bad Request. Invalid dashboard id, Invalid
request format
          schema:
            type: object
            properties:
              message:
                type: string
                example: Invalid dashboard id
        '500':
          description: SQLAlchemy Error
          schema:
            type: object
            properties:
              message:
                type: string
  /dashboard/deleteMultipleDashboards:
    delete:
      tags:
        - Dashboards
      summary: Delete Dashboard
      description: >-
        API to Delete dashboard
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: body
          in: body
          description: Request Body
          required: true
          schema:
            type: object
            properties:
              dashboard_ids:
                type: array
                items:
                  type: number
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              status:
                type: string
                example: Deleted
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '500':
          description: SQLAlchemyError
          schema:
            type: object
            properties:
              message:
                type: string

  /dashboard/exportAll:
    get:
      tags:
        - Dashboards
      summary: Returns the requested json file as "file download" i.e.
with content-disposition = attachment
      description: >-
        API to export of Dashboards
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
      responses:
        '200':
          description: the value is 'attachment;
filename=dashboards_config.json'
          schema:
            type: file
          headers:
            content-Disposition:
              type: string
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
  /dashboard/export:
    post:
      tags:
        - Dashboards
      summary: Returns the requested json file as "file download" i.e.
with content-disposition = attachment
      description: >-
        API to export of Dashboard
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: body
          in: body
          description: Request Body
          required: true
          schema:
            type: object
            properties:
              dashboard_ids:
                type: array
                items:
                  type: number
      responses:
        '200':
          description: the value is 'attachment;
filename=dashboards_config.json'
          schema:
            type: file
          headers:
            content-Disposition:
              type: string
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid Request Body
          schema:
            $ref: '#/definitions/InvalidRequestBody'
  /dashboard/import:
    post:
      tags:
        - Dashboards
      summary: Get import of Dashboard
      description: >-
        API to import of Dashboard data
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: body
          in: body
          description: Request Body
          required: true
          schema:
            type: array
            items:
              type: object
              properties:
                name:
                  type: string
                description:
                  type: string
                start_time:
                  type: string
                end_time:
                  type: string
                is_custom_start_time:
                  type: string
                refresh_interval:
                  type: string
                time_range_enabled:
                  type: string
                refresh_enabled:
                  type: string
                edit_refresh_enabled:
                  type: string
                filter_enabled:
                  type: boolean
                filters:
                  type: array
                  items:
                    type: string
                widgets:
                  type: array
                  items:
                    type: object
                    properties:
                      widget_name:
                        type: string
                      name:
                        type: string
                      row:
                        type: string
                      column:
                        type: string
                      height:
                        type: string
                      width:
                        type: string
                      chart_type:
                        type: string
      responses:
        '201':
          description: Success
          schema:
            type: object
            properties:
              status:
                type: string
                example: imported
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid Configuration File, Dashboard name
already exists, Invalid Widget
          schema:
            type: object
            properties:
              message:
                type: string
                example: Invalid Configuration File
        '500':
          description: SQLAlchemy Error
          schema:
            type: object
            properties:
              message:
                type: string
  /search/getSearchCount:
    get:
      tags:
        - Search
      summary: Get number of searches
      description: >-
        API to get number of searches
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: searchKeyword
          in: query
          description: Search keyword
          type: string
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              searches_count:
                type: number
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '500':
          description: SQLAlchemy Error
          schema:
            type: object
            properties:
              message:
                type: string
  /search/getSearches:
    get:
      tags:
        - Search
      summary: Get searches
      description: >-
        API to get searches
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: sortBy
          in: query
          description: Sort By Term. Example :- time or name
          type: string
        - name: sortOrder
          in: query
          description: Sort order. Example :- asc or desc
          type: string
        - name: from
          in: query
          description: Starting Offset. Note :- For first widget value
will be 0
          type: number
        - name: size
          in: query
          description: Number of widgets to fetch. If -1 is passed then
all widgets will be fetched.
          type: number
        - name: searchKeyword
          in: query
          description: Search keyword
          type: string
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              search_details:
                type: object
                properties:
                  search_name:
                    type: string
                  search_id:
                    type: number
                  description:
                    type: string
                  last_updated:
                    type: string
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
  /search/searchByFilters:
    post:
      tags:
        - Search
      summary: Get search data by filters
      description: >-
        API to get search data by filters
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: body
          in: body
          description: Request Body
          required: true
          schema:
            type: object
            properties:
              dataset_id:
                type: number
              columns:
                type: array
                items:
                  type: string
              search_tags:
                type: array
                items:
                  type: string
              timeRange:
                type: object
                properties:
                  from:
                    type: number
                  to:
                    type: number
              orderby:
                type: string
              scroll_id:
                type: string
              clear_scroll:
                type: boolean
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              scroll_id:
                type: string
              data:
                type: array
                items:
                  type: object
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid Request Body
          schema:
            $ref: '#/definitions/InvalidRequestBody'
        '500':
          description: Elasticsearch Exception
          schema:
            type: object
            properties:
              message:
                type: string
  /search/getDatasetColumnsById:
    get:
      tags:
        - Search
      summary: Get Dataset details for Search
      description: >-
        API to get dataset column details by dataset id
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: datasetId
          in: query
          description: Dataset Id
          required: true
```

```
          type: number
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              index:
                type: string
              columns:
                type: array
                items:
                  type: object
                  properties:
                    column_id:
                      type: number
                    column_name:
                      type: string
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid request format, Invalid dataset id
          schema:
            type: object
            properties:
              message:
                type: string
                example: Invalid request format
  /search/createSearch:
    post:
      tags:
        - Search
      summary: Create new search
      description: >-
        API to create new search
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: session token
          required: true
          type: string
        - name: body
          in: body
          description: Request Body
          required: true
          schema:
            type: object
            properties:
              name:
                type: string
              description:
                type: string
              datasets:
                type: array
                items:
                  type: object
                  properties:
                    dataset_id:
                      type: number
                    columns:
                      type: array
                      items:
                        type: number
      responses:
        '201':
          description: Success
          schema:
            type: object
            properties:
              status:
                type: string
                example: Created
              search_id:
                type: number
                description: Created Model Id
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid Request Body
          schema:
            $ref: '#/definitions/InvalidRequestBody'
        '500':
          description: SQLAlchemyError
          schema:
            type: object
            properties:
              message:
                type: string
  /search/updateSearch:
    put:
      tags:
        - Search
      summary: Update search
      description: >-
        API to update search
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: search_id
          in: query
          description: Search Id to be updated
          required: true
          type: number
        - name: body
          in: body
          description: Request Body
          required: true
          schema:
            type: object
            properties:
              name:
                type: string
              description:
                type: string
              datasets:
                type: array
                items:
                  type: object
                  properties:
                    dataset_id:
                      type: number
                    columns:
                      type: array
                      items:
                        type: number
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              status:
                type: string
                example: Updated
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid Request Body, Invalid request format,
Invalid dataset id
          schema:
            type: object
            properties:
              message:
                type: string
                example: Invalid Request Body
        '500':
          description: SQLAlchemyError
          schema:
            type: object
            properties:
              message:
                type: string
  /search/getSearchDetails:
    get:
      tags:
        - Search
      summary: Get Search Details
      description: >-
        API to Get Search Details
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: search_id
          in: query
          description: Search Id to be updated
          required: true
          type: number
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              search_details:
                type: object
                properties:
                  name:
                    type: string
                  description:
                    type: string
                  time_column_enabled:
                    type: boolean
                  datasets:
                    type: array
                    items:
                      type: object
                      properties:
                        dataset_id:
                          type: number
                        columns:
                          type: array
                          items:
                            type: number
                        default_timestamp:
                          type: string
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid request format, Invalid search id
          schema:
            type: object
            properties:
              message:
                type: string
                example: Invalid request format
  /search/deleteMultipleSearches:
    delete:
      tags:
        - Search
      summary: Delete Search
      description: >-
        API to Delete search
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
```

```
          required: true
          type: string
        - name: body
          in: body
          description: Request Body
          required: true
          schema:
            type: object
            properties:
              search_ids:
                type: array
                items:
                  type: number
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              status:
                type: string
                example: Deleted
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '500':
          description: SQLAlchemyError
          schema:
            type: object
            properties:
              message:
                type: string

  /search/deleteSearch:
    delete:
      tags:
        - Search
      summary: Delete Search by Id
      description: >-
        API to Delete Search by Id
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: search_id
          in: query
          description: Search Id to be deleted
          required: true
          type: number
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              status:
                type: string
                example: Deleted
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid request format, Invalid search id
          schema:
            type: object
            properties:
              message:
                type: string
                example: Invalid request format
        '500':
          description: SQLAlchemyError
          schema:
            type: object
            properties:
              message:
                type: string
  /search/import:
    post:
      tags:
        - Search
      summary: Get import of search
      description: >-
        API to import of search data
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: body
          in: body
          description: Request Body
          required: true
          schema:
            type: array
            items:
              type: object
              properties:
                name:
                  type: string
                description:
                  type: string
                datasets:
                  type: array
                  items:
                    type: object
                    properties:
                      name:
                        type: string
                      columns:
                        type: array
                        items:
                          type: string
      responses:
        '201':
          description: Success
          schema:
            type: object
            properties:
              status:
                type: string
                example: imported
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid Configuration File, Search name
already exists, Invalid Dataset
          schema:
            type: object
            properties:
              message:
                type: string
                example: Invalid Configuration File
        '500':
          description: SQLAlchemyError
          schema:
            type: object
            properties:
              message:
                type: string
  /search/export:
    post:
      tags:
        - Search
      summary: Returns the requested json file as "file download" i.e.
with content-disposition = attachment
      description: >-
        API to export of search
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: body
          in: body
          description: Request Body
          required: true
          schema:
            type: object
            properties:
              search_ids:
                type: array
                items:
                  type: number
      responses:
        '200':
          description: the value is 'attachment;
filename=searches_config.json'
          schema:
            type: file
          headers:
            content-Disposition:
              type: string
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid Request Body
          schema:
            $ref: '#/definitions/InvalidRequestBody'
  /search/exportAll:
    get:
      tags:
        - Search
      summary: Returns the requested json file as "file download" i.e.
with content-disposition = attachment
      description: >-
        API to export of searches
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
      responses:
        '200':
          description: the value is 'attachment;
filename=searches_config.json'
          schema:
            type: file
          headers:
            content-Disposition:
              type: string
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'

  /predictionmodels/deleteMultiplePredictionModel:
    delete:
      tags:
        - Predictions
      summary: Delete Prediction Model
      description: >-
        API to Delete Prediction model
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: body
          in: body
          description: Request Body
          required: true
          schema:
            type: object
            properties:
              model_ids:
                type: array
                items:
                  type: number
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              status:
                type: string
                example: Deleted
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Integrity Error, Bad Request. Invalid request format,
Invalid model id
          schema:
            type: object
            properties:
              message:
                type: string
                example: Unable to delete Prediction Model. Model is in
use
        '500':
          description: SQLAlchemyError
          schema:
            type: object
            properties:
              message:
                type: string

  /predictionmodels/isModelRunning:
    get:
      tags:
        - Predictions
      summary: API to check whether a model is running or not
      description: >-
        API to check whether a model is running or not
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: modelId
          in: query
          description: Model Id
          required: true
          type: number
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              isModelRunning:
                type: boolean
        '400':
          description: Bad request, Invalid request format
          schema:
            type: object
            properties:
              message:
                type: string
                example: ML is not enabled
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
  /predictionmodels/isModelTrained:
    get:
      tags:
        - Predictions
      summary: API to check whether a model is trained or not
      description: >-
        API to check whether a model is trained or not
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: modelId
          in: query
          description: Model Id
          required: true
          type: number
      responses:
        '200':
          description: Success
          schema:
            type: string
            example : "True"
        '400':
          description: Bad request, Inavalid request format, Invalid
Query Parameters
          schema:
            type: object
            properties:
              message:
                type: string
                example: Inavalid request format
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
  /predictionmodels/ml/validateMLAPI:
    get:
      tags:
        - Predictions
      summary: API to validate token for consuming predictions
      description: >-
        API to validate token for consuming predictions
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: account
          in: query
          description: Account name
          required: true
          type: string
        - name: clientId
          in: query
          description: ML Client id
          required: true
          type: string
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              userDetails:
                type: object
                properties:
                  accountId:
                    type: string
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'

  /predictionmodels/createPredictionModel:
    post:
      tags:
        - Predictions
      summary: Create new prediction model
      description: >-
        API to create new prediction model
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: body
          in: body
          description: Request Body
          required: true
          schema:
            type: object
            properties:
              model_name:
                type: string
              dataset_id:
                type: number
              description:
                type: string
              algorithm:
                type: string
              training_set:
                type: number
              batch_size:
                type: number
              scheduled_interval:
                type: string
              metrices :
                type: array
                items:
                  type: string
              input_params:
                type: array
                items:
                  type: object
                  properties:
                    column_id:
                      type: number
                    function_list:
                      type: array
                      items:
                        type: string
                      example:
                        - Make Categorical
                        - Replace missing data
                    output_params:
                      type: array
                      items:
                        type: object
                        properties:
                          column_id:
                            type: number
      responses:
        '201':
          description: Success
          schema:
            type: object
            properties:
              status:
                type: string
                example: Created
              model_id:
                type: number
                description: Created Model Id
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
```

```
        '400':
          description: >-
            Bad Request. Invalid Request Body
          schema:
            $ref: '#/definitions/InvalidRequestBody'
        '500':
          description: SQLAlchemyError
          schema:
            type: object
            properties:
              message:
                type: string
  /predictionmodels/exportAll:
    get:
      tags:
        - Predictions
      summary: Returns the requested json file as "file download" i.e.
with content-disposition = attachment
      description: >-
        API to export of PredictionModels
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
      responses:
        '200':
          description: the value is 'attachment;
filename=predictionmodels_config.json'
          schema:
            type: file
          headers:
            content-Disposition:
              type: string
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
  /predictionmodels/export:
    post:
      tags:
        - Predictions
      summary: Returns the requested json file as "file download" i.e.
with content-disposition = attachment
```

```
      description: >-
        API to export of PredictionModels
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: body
          in: body
          description: Request Body
          required: true
          schema:
            type: object
            properties:
              model_ids:
                type: array
                items:
                  type: number
      responses:
        '200':
          description: the value is 'attachment;
filename=predictionmodels_config.json'
          schema:
            type: file
          headers:
            content-Disposition:
              type: string
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid Request Body
          schema:
            $ref: '#/definitions/InvalidRequestBody'
  /predictionmodels/import:
    post:
      tags:
        - Predictions
      summary: Get import of PredictionModels
      description: >-
        API to import of PredictionModels data
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: body
          in: body
          description: Request Body
          required: true
          schema:
            type: array
            items:
              type: object
              properties:
                name:
                  type: string
                description:
                  type: string
                algorithm:
                  type: string
                training_set:
                  type: number
                batch_size:
                  type: number
                scheduled_interval:
                  type: string
                metrices :
                  type: array
                  items:
                    type: string
                input_params:
                  type: array
                  items:
                    type: object
                    properties:
                      column:
                        type: string
                      function_list:
                        type: array
                        items:
                          type: string
                        example:
                          - Make Categorical
                          - Replace missing data
                output_params:
                  type: array
                  items:
                    type: object
                    properties:
                      column:
                        type: string
      responses:
        '201':
          description: Success
          schema:
            type: object
            properties:
              status:
                type: string
                example: imported
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid Configuration File, PredictionModel
name already exists, Invalid Index
          schema:
            type: object
            properties:
              message:
                type: string
                example: Invalid Configuration File
        '500':
          description: SQLAlchemyError
          schema:
            type: object
            properties:
              message:
                type: string
  /predictionmodels/getPredictionModels:
    get:
      tags:
        - Predictions
      summary: Get prediction models
      description: >-
        API to get prediction models
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: sortBy
          in: query
          description: Sort By Term. Example :- time or name
          type: string
        - name: sortOrder
          in: query
          description: Sort order. Example :- asc or desc
          type: string
        - name: from
          in: query
          description: Starting Offset. Note :- For first Prediction
model value will be 0
          type: number
        - name: size
          in: query
          description: Number of Prediction models to fetch. If -1 is
passed then all prediction models will be fetched.
          type: number
        - name: searchKeyword
          in: query
          description: Search keyword
          type: string
        - name: searchAlgorithm
          in: query
          description: Search Algorithm
          type: string
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              predictionmodels:
                type: array
                items:
                  type: object
                  properties:
                    model_name:
                      type: string
                    model_id:
                      type: number
                    model_description:
                      type: string
                    output:
                      type: string
                    last_updated:
                      type: string
        '401':
          description: >-
```

```
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '500':
          description: SQLAlchemyError
          schema:
            type: object
            properties:
              message:
                type: string
  /predictionmodels/updatePredictionModel:
    put:
      tags:
        - Predictions
      summary: Update prediction model
      description: >-
        API to update prediction model
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: modelId
          in: query
          description: Model Id to be updated
          required: true
          type: number
        - name: body
          in: body
          description: Request Body
          required: true
          schema:
            type: object
            properties:
              model_name:
                type: string
              dataset_id:
                type: number
              description:
                type: string
              algorithm:
                type: string
              training_set:
                type: number
              batch_size:
```

```
          type: number
        scheduled_interval:
          type: string
        metrices :
          type: array
          items:
            type: string
        input_params:
          type: array
          items:
            type: object
            properties:
              column_id:
                type: number
              function_list:
                type: array
                items:
                  type: string
                example:
                  - Make Categorical
                  - Replace missing data
              output_params:
                type: array
                items:
                  type: object
                  properties:
                    column_id:
                      type: number
responses:
  '200':
    description: Success
    schema:
      type: object
      properties:
        status:
          type: string
          example: Updated
  '401':
    description: >-
      Unauthorized. Invalid Login
    schema:
      $ref: '#/definitions/InvalidLoginResponse'
  '400':
    description: >-
      Bad Request. Invalid Request Body, Invalid request format
    schema:
      type: object
      properties:
        message:
          type: string
```

```
                example: Invalid Request Body
        '500':
          description: SQLAlchemyError
          schema:
            type: object
            properties:
              message:
                type: string
  /predictionmodels/getPredictionModelDetails:
    get:
      tags:
        - Predictions
      summary: Get prediction model details
      description: >-
        API to get prediction model details
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: modelId
          in: query
          description: Model Id
          type: number
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              model_details:
                type: object
                properties:
                  model_name:
                    type: string
                  dataset_id:
                    type: number
                  description:
                    type: string
                  algorithm:
                    type: string
                  training_set:
                    type: number
                  batch_size:
                    type: number
                  scheduled_interval:
                    type: string
                  metrices :
                    type: array
                    items:
                      type: string
                  input_params:
                    type: array
                    items:
                      type: object
                      properties:
                        column_id:
                          type: number
                        function_list:
                          type: array
                          items:
                            type: string
                            example:
                              - Make Categorical
                              - Replace missing data
                        output_params:
                          type: array
                          items:
                            type: object
                            properties:
                              column_id:
                                type: number
        '401':
          description: Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid request format, Invalid model id
          schema:
            type: object
            properties:
              message:
                type: string
                example: Invalid request format
        '500':
          description: SQLAlchemyError
          schema:
            type: object
            properties:
              message:
                type: string
  /predictionmodels/deletePredictionModel:
    delete:
      tags:
        - Predictions
      summary: Delete prediction model
```

```
      description: >-
        API to Delete prediction model
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: modelId
          in: query
          description: Model Id
          type: number
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              status:
                type: string
                example: Deleted
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Integrity Error, Bad Request. Invalid request format,
Invalid model id
          schema:
            type: object
            properties:
              message:
                type: string
                example: Unable to delete Prediction Model. Model is in
use
        '500':
          description: SQLAlchemyError
          schema:
            type: object
            properties:
              message:
                type: string
  /predictionmodels/getPredictionModelsCount:
    get:
      tags:
        - Predictions
      summary: Get count of prediction models
      description: >-
        API to get count of prediction models
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: searchKeyword
          in: query
          description: Search keyword
          type: string
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              predictionmodels_count:
                type: number
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '500':
          description: SQLAlchemyError
          schema:
            type: object
            properties:
              message:
                type: string
  /predictionmodels/getAvailablefunctions:
    get:
      tags:
        - Predictions
      summary: Get all available functions
      description: API to get all preprocessing functions
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              Preprocessor_functions:
                type: array
                items:
                  type: object
                  properties:
                    category_type:
                      type: string
                    function_name:
                      type: string
        '401':
          description: Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '500':
          description: SQLAlchemyError
          schema:
            type: object
            properties:
              message:
                type: string
  /predictionmodels/getAvailableAlgorithms:
    get:
      tags:
        - Predictions
      summary: Get all available algorithms
      description: API to get all ML algorithms
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              Algorithms:
                type: array
                items:
                  type: object
                  properties:
```

```
                  category_type:
                    type: string
                  algorithm_name:
                    type: string
      '401':
        description: Unauthorized. Invalid Login
        schema:
          $ref: '#/definitions/InvalidLoginResponse'
      '500':
        description: SQLAlchemyError
        schema:
          type: object
          properties:
            message:
              type: string
/predictionmodels/getAvailableMetrices:
  get:
    tags:
      - Predictions
    summary: Get all available metrices for each algorithm
    description: API to get all metrices for algorithm
    produces:
      - application/json
    parameters:
      - name: token
        in: header
        description: Session Token
        required: true
        type: string
      - name: algorithmName
        in: query
        description: Name of the algorithm
        type: string
    responses:
      '200':
        description: Success
        schema:
          type: object
          properties:
            metrices:
              type: array
              items:
                type: string
      '401':
        description: Unauthorized. Invalid Login
        schema:
          $ref: '#/definitions/InvalidLoginResponse'
      '500':
        description: SQLAlchemyError
        schema:
          type: object
          properties:
            message:
              type: string
/predictionmodels/getModelMetricesValue:
  get:
    tags:
      - Predictions
    summary: Get metric values for each model
    description: API to get all metric values for a model
    produces:
      - application/json
    parameters:
      - name: token
        in: header
        description: Session Token
        required: true
        type: string
      - name: modelId
        in: query
        description: Model Id
        type: integer
    responses:
      '200':
        description: Success
        schema:
          type: object
      '401':
        description: Unauthorized. Invalid Login
        schema:
          $ref: '#/definitions/InvalidLoginResponse'
      '400':
        description: >-
          Bad Request. Invalid request format
        schema:
          type: object
          properties:
            message:
              type: string
              example: Invalid request format
      '500':
        description: SQLAlchemyError
        schema:
          type: object
          properties:
            message:
              type: string

/predictionmodels/getPredictionModelParamsById:
  get:
```

```
tags:
  - Predictions
summary: Get model Columns details
description: >-
  API to get column details of a prediction model
produces:
  - application/json
parameters:
  - name: token
    in: header
    description: Session Token
    required: true
    type: string
  - name: modelId
    in: query
    description: model Id
    required: true
    type: number
responses:
  '200':
    description: Success
    schema:
      type: object
      properties:
        index:
          type: string
        columns:
          type: array
          items:
            type: object
            properties:
              column_id:
                type: number
              column_name:
                type: string
        output_param:
          type: array
          items:
            type: object
            properties:
              column_id:
                type: number
              column_name:
                type: string
  '401':
    description: >-
      Unauthorized. Invalid Login
    schema:
      $ref: '#/definitions/InvalidLoginResponse'
  '400':
          description: >-
            Bad Request. Invalid request format
          schema:
            type: object
            properties:
              message:
                type: string
                example: Invalid request format

  /alert/createAlert:
    post:
      tags:
        - Alerts
      summary: Create new Alert
      description: >-
        API to create new alert
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: body
          in: body
          description: Request Body. Please note - value for the fields
to, cc, bcc, subject, message has to be in encrypted form.
          required: true
          schema:
            type: object
            properties:
              alert_name:
                type: string
              dataset_id:
                type: number
              description:
                type: string
              queries:
                type: array
                items:
                  type: string
              timeRange:
                type: string
              evaluate_interval:
                type: string
              notify_interval:
                type: string
              to:
                type: string
              cc:
                type: string
              bcc:
                type: string
              subject:
                type: string
              message:
                type: string
      responses:
        '201':
          description: Success
          schema:
            type: object
            properties:
              status:
                type: string
                example: Created
              model_id:
                type: number
                description: Created Model Id
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid Request Body
          schema:
            $ref: '#/definitions/InvalidRequestBody'
        '500':
          description: SQLAlchemyError
          schema:
            type: object
            properties:
              message:
                type: string
  /alert/getAlerts:
    get:
      tags:
        - Alerts
      summary: Get Alerts
      description: >-
        API to get alerts
      produces:
        - application/json
      parameters:
        - name: token
```

```
          in: header
          description: Session Token
          required: true
          type: string
        - name: sortBy
          in: query
          description: Sort By Term. Example :- time or name
          type: string
        - name: sortOrder
          in: query
          description: Sort order. Example :- asc or desc
          type: string
        - name: from
          in: query
          description: Starting Offset. Note :- For first alert value
will be 0
          type: number
        - name: size
          in: query
          description: Number of Alertss to fetch. If -1 is passed then
all alerts will be fetched.
          type: number
        - name: searchKeyword
          in: query
          description: Search keyword
          type: string
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              alert_details:
                type: array
                items:
                  type: object
                  properties:
                    alert_name:
                      type: string
                    alert_id:
                      type: number
                    description:
                      type: string
                    last_updated:
                      type: string
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
```

```
  /alert/updateAlert:
    put:
      tags:
        - Alerts
      summary: Update Alert
      description: >-
        API to update alert
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: alert_id
          in: query
          description: Alert Id to be updated
          required: true
          type: number
        - name: body
          in: body
          description: Request Body. Please note - value for the fields
to, cc, bcc, subject, message has to be in encrypted form.
          required: true
          schema:
            type: object
            properties:
              alert_name:
                type: string
              dataset_id:
                type: number
              description:
                type: string
              queries:
                type: array
                items:
                  type: string
              timeRange:
                type: string
              evaluate_interval:
                type: string
              notify_interval:
                type: string
              to:
                type: string
              cc:
                type: string
              bcc:
                type: string
              subject:
                type: string
              message:
                type: string
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              status:
                type: string
                example: Updated
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid Request Body, Invalid alert id,
Inavalid alert id
          schema:
            type: object
            properties:
              message:
                type: string
                example: Invalid Request Body
        '500':
          description: SQLAlchemyError
          schema:
            type: object
            properties:
              message:
                type: string
  /alert/getAlertDetails:
    get:
      tags:
        - Alerts
      summary: Get prediction model details
      description: >-
        API to get prediction model details
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: alert_id
          in: query
          description: Alert Id
          type: number
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              alert_details:
                type: object
                properties:
                  alert_name:
                    type: string
                  dataset_id:
                    type: number
                  description:
                    type: string
                  queries:
                    type: string
                  timeRange:
                    type: string
                  evaluate_interval:
                    type: string
                  notify_interval:
                    type: string
                  to:
                    type: string
                  cc:
                    type: string
                  bcc:
                    type: string
                  subject:
                    type: string
                  message:
                    type: string
        '401':
          description: Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
  /alert/deleteAlert:
    delete:
      tags:
        - Alerts
      summary: Delete Alert
      description: >-
        API to Delete alert
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: alert_id
          in: query
          description: Alert Id
          type: number
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              status:
                type: string
                example: Deleted
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid request format, Invalid alert id
          schema:
            type: object
            properties:
              message:
                type: string
                example: Invalid request format
        '500':
          description: SQLAlchemyError
          schema:
            type: object
            properties:
              message:
                type: string


  /alert/deleteMultipleAlerts:
    delete:
      tags:
        - Alerts
      summary: Delete Alert
      description: >-
```

```
        API to Delete alert
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: body
          in: body
          description: Request Body
          required: true
          schema:
            type: object
            properties:
              alert_ids:
                type: array
                items:
                  type: number
      responses:
        '200':
          description: Success
          schema:
            type: object
            properties:
              status:
                type: string
                example: Deleted
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: Bad Request. Invalid request format, Invalid
alert id
          schema:
            type: object
            properties:
              message:
                type: string
                example: Invalid request format
        '500':
          description: SQLAlchemyError
          schema:
            type: object
            properties:
              message:
                type: string
```

```
/alert/getAlertsCount:
  get:
    tags:
      - Alerts
    summary: Get count of Alerts
    description: >-
      API to get count of Alerts
    produces:
      - application/json
    parameters:
      - name: token
        in: header
        description: Session Token
        required: true
        type: string
      - name: searchKeyword
        in: query
        description: Search keyword
        type: string
    responses:
      '200':
        description: Success
        schema:
          type: object
          properties:
            alerts_count:
              type: number
      '401':
        description: >-
          Unauthorized. Invalid Login
        schema:
          $ref: '#/definitions/InvalidLoginResponse'
      '500':
        description: SQLAlchemyError
        schema:
          type: object
          properties:
            message:
              type: string

/alert/import:
  post:
    tags:
      - Alerts
    summary: Get import of alert
    description: >-
      API to import of alerts
    consumes:
      - application/json
```

```
produces:
  - application/json
parameters:
  - name: token
    in: header
    description: Session Token
    required: true
    type: string
  - name: body
    in: body
    description: Request Body
    required: true
    schema:
      type: array
      items:
        type: object
        properties:
          alert_name :
            type: string
          description:
            type: string
          dataset_name:
            type: string
          queries:
            type: array
            items:
              type: string
          time_range:
            type: string
          evaluate_interval:
            type: string
          notify_interval:
            type: string
          subject:
            type: string
          email_to:
            type: string
          email_cc:
            type: string
          email_bcc:
            type: string
          message:
            type: string
responses:
  '201':
    description: Success
    schema:
      type: object
      properties:
        status:
                type: string
                example: imported
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid Configuration File, Alert name already
exists
          schema:
            type: object
            properties:
              message:
                type: string
                example: Invalid Configuration File
        '500':
          description: SQLAlchemy Error
          schema:
            type: object
            properties:
              message:
                type: string

  /alert/export:
    post:
      tags:
        - Alerts
      summary: Returns the requested json file as "file download" i.e.
with content-disposition = attachment
      description: >-
        API to export of alert
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: body
          in: body
          description: Request Body
          required: true
          schema:
            type: object
            properties:
              alert_ids:
                type: array
                items:
                  type: number
      responses:
        '200':
          description: the value is 'attachment;filename=alerts_config.
json'
          schema:
            type: file
          headers:
            content-Disposition:
              type: string
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid Request Body
          schema:
            $ref: '#/definitions/InvalidRequestBody'
  /alert/exportAll:
    get:
      tags:
        - Alerts
      summary: Returns the requested json file as "file download" i.e.
with content-disposition = attachment
      description: >-
        API to export of Alerts
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
      responses:
        '200':
          description: the value is 'attachment;filename=alerts_config.
json'
          schema:
            type: file
          headers:
            content-Disposition:
              type: string
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
  /dataset/import:
    post:
      tags:
        - Datasets
      summary: Get import of dataset
      description: >-
        API to import of datasets
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: body
          in: body
          description: Request Body
          required: true
          schema:
            type: array
            items:
              type: object
              properties:
                name:
                  type: string
                index_name:
                  type: string
                description:
                  type: string
                default_timestamp:
                  type: string
                filters:
                  type: array
                  items:
                    type: string
                columns:
                  type: array
                  items:
                    type: object
                    properties:
                      column_name:
                        type: string
                      alias:
                        type: string
      responses:
        '201':
          description: Success
          schema:
            type: object
            properties:
              status:
                type: string
                example: imported
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid Configuration File, Dataset name
already exists
          schema:
            type: object
            properties:
              message:
                type: string
                example: Invalid Configuration File
        '500':
          description: SQLAlchemy Error
          schema:
            type: object
            properties:
              message:
                type: string

  /dataset/export:
    post:
      tags:
        - Datasets
      summary: Returns the requested json file as "file download" i.e.
with content-disposition = attachment
      description: >-
        API to export of dataset
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
        - name: body
          in: body
          description: Request Body
          required: true
          schema:
            type: object
            properties:
              dataset_ids:
                type: array
                items:
                  type: number
      responses:
        '200':
          description: the value is 'attachment;
filename=datasets_config.json'
          schema:
            type: file
          headers:
            content-Disposition:
              type: string
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
        '400':
          description: >-
            Bad Request. Invalid Request Body
          schema:
            $ref: '#/definitions/InvalidRequestBody'
  /dataset/exportAll:
    get:
      tags:
        - Datasets
      summary: Returns the requested json file as "file download" i.e.
with content-disposition = attachment
      description: >-
        API to export of datasets
      consumes:
        - application/json
      produces:
        - application/json
      parameters:
        - name: token
          in: header
          description: Session Token
          required: true
          type: string
      responses:
        '200':
          description: the value is 'attachment;
filename=datasets_config.json'
          schema:
            type: file
          headers:
            content-Disposition:
              type: string
        '401':
          description: >-
            Unauthorized. Invalid Login
          schema:
            $ref: '#/definitions/InvalidLoginResponse'
definitions:
  InvalidLoginResponse:
    type: object
    properties:
      message:
        type: string
        example: Invalid Login Credentials
  InvalidRequestBody:
    type: object
    properties:
      message:
        type: string
        example: Invalid Request Body
```