

Workflow

Workflow			Show/Hide	List Operations	Expand Operations
GET	/api/workflow	On-going workflow instances			
DELETE	/api/workflow/{instanceid}/{currentbookmark}	Terminate Workflow			
GET	/api/workflow/{instanceid}/{currentbookmark}	Get Workflow			
POST	/api/workflow/counts	On-going workflow instances with count			
GET	/api/workflow/file	Get File			
POST	/api/workflow/resume/externalapi/	Resume Workflow			
POST	/api/workflow/resume/portal/	Resume Workflow			
POST	/api/workflow/start/	Start Workflow			
POST	/api/workflow/start/withparams	Start Workflow			
GET	/api/workflow/status	Workflow Status			
GET	/api/workflow/validate/{instanceid}/{currentbookmark}	Validate Current Form			
POST	/api/workflow/withcriteria	On-going workflow instances with criteria			
POST	/api/workflow/withcustomcolumn	On-going workflow instances with custom column and criteria			

İçerdiği Metotlar

1) /api/workflow/

Logo Flow üzerinde devam eden işlerin listesini döner.

Input Parameters

- **Authorization(Headers):** Login metotlarından herhangi biri kullanılarak alınan token bilgisi geçerli.
- **offset(query):** Gelen verinin kaçınıcı sayfadan itibaren listeleneceği bilgisidir.
- **limit(query):** Gelen verinin verilen offsete göre kaçar kaçar gösterileceği bilgisidir.
- **sort(query):** Gelen veride sıralama yapmak için kullanılır. asc,desc
- **withcount(query):** Toplam kayıt sayısını döndürür. Gerekecekçe kullanmayınız.

2) /api/workflow/{instanceid}/{currentbookmark}- Delete

Logo Flow üzerinde ilgili instance idli işi iptal eder.

Input Parameters

- **Authorization(Headers):** Login metotlarından herhangi biri kullanılarak alınan token bilgisi geçerli.
- **storageid(query):** Silinecek olan işin akış idsidir.
- **instanceid(path):** Silinecek işin guid id bilgisidir. Format: xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxx (UID).
- **currentbookmark(path):** Silinecek işin bulunduğu adım guid bilgisidir. Format: xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxx

3) /api/workflow/{instanceid}/{currentbookmark}- Get

Logo Flow üzerinde ilgili instance id li işi döner.

Input Parameters

- **Authorization(Headers):** Login metotlarından herhangi biri kullanılarak alınan token bilgisi geçerli.
- **storageid(query):** Silinecek olan işin akış idsidir.
- **instanceid(path):** Silinecek işin guid id bilgisidir. Format: xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxx (UID).
- **currentbookmark(path):** Silinecek işin bulunduğu adım guid bilgisidir. Format: xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxx

4) /api/workflow/count

Logo Flow üzerinde devam eden işlerin sayısını döner.

Input Parameters

- **Authorization(Headers):** Login metotlarından herhangi biri kullanılarak alınan token bilgisi geçerli.
- **keyFilter(body):** Veriyi filtreli bir şekilde getirmek için kullanılır.

```
{
  "KeyFilters": [
    {
      "Operator": "string" : eq,neq,gt,lt,lte,contains,doesnotcontain
      "Field": "string", : filtre uygulanacak alan
      "Value": {} : filtre uygulanacak alanın değeri
    }
  ],
  "Logic": "string" : birden fazla filtre içeren requestlerde filtrelerin bağlaçları. (and,or)
}
```

5) /api/workflow/file

Logo Flow üzerinde akışa eklenen doküman bilgisini ve indirme linkini döner.

Input Parameters

- **Authorization(Headers):** Login metotlarından herhangi biri kullanılarak alınan token bilgisi geçerli.
- **fileid(query):** İşe eklenen doküman id bilgisidir.

6) /api/workflow/resume/externalapi

Logo Flow üzerinde iş akışını devam ettir ve sonraki adıma ait bilgileri döner.

Input Parameters

- **Authorization(Headers):** Login metotlarından herhangi biri kullanılarak alınan token bilgisi geçerli.
- **latitude(query):** Mobil uygulamadan iletlenen iş için konum bilgisidir.(enlem)
- **longitude(query):** Mobil uygulamadan iletlenen iş için konum bilgisidir.(boylam)
- **resumeparameters(body):** Devam ettirilecek işe ait bilgiler parametre geçerli.

```
{
  "storageid": 0,
  "instanceid": "string",
  "currentbookmark": "string",
  "NextButtonName": "string",
  "ResumeParameters": [
    "string"
  ],
  "RequestLocationType": "0",
  "CommentList": [
    {
      "Id": {},
      "Comment": "string",
      "OperationType": "0"
    }
  ],
  "FileList": [
    {
      "FileId": {},
      "FileData": "string",
      "FileName": "string",
      "OperationType": "0"
    }
  ],
  "ValidateFilesAndComments": true
}
```

7) /api/workflow/start

Logo Flow üzerinde akış başlatmak için kullanılır.

Input Parameters

- **Authorization(Headers)**: Login metotlarından herhangi biri kullanılarak alınan token bilgisi geçerli.
- **storageid(query)**: Başlatılmak istenen akış id bilgisidir.
- **description(query)**: Akış başlatılırken açıklama eklenmek istenirse girilir.
- **publishtype(query)**: Akışın yayın modu başlatılmak isteniyorsa 1, geliştirim modu başlatılmak isteniyorsa 0 gönderilir.
- **latitude(query)**: Mobil uygulamadan başlatılan iş için konum bilgisidir.(enlem)
- **longitude(query)**: Mobil uygulamadan başlatılan iş için konum bilgisidir.(boylam)

8) /api/workflow/start/withparams

Logo Flow üzerinde detay parametreler ile akış başlatmak için kullanılır.

Input Parameters

- **Authorization(Headers)**: Login metotlarından herhangi biri kullanılarak alınan token bilgisi geçerli.
- **startupParameters(body)**: Akış başlatırken kullanılabilecek parametreler gönderilir.

```

{
  "storageid": 0,
  "description": "string",
  "storagePublishType": "0",
  "StartupParameters": [
    "string"
  ],
  "CommentList": [-- yorum
    {
      "Id": {},
      "Comment": "string",
      "OperationType": "0"
    }
  ],
  "FileList": [ --dosya
    {
      "FileId": {},
      "FileData": "string",
      "FileName": "string",
      "OperationType": "0"
    }
  ]
}

```

9) /api/workflow/status

Logo Flow üzerinde başlatılan işlerin durumunu döner.

Input Parameters

- **Authorization(Headers)**: Login metotlarından herhangi biri kullanılarak alınan token bilgisi geçerli.

10) /api/workflow/validate/{instanceid}/{currentbookmark}

Logo Flow üzerinde ilgili instance id li işin validasyon bilgilerini döner.

Input Parameters

- **Authorization(Headers)**: Login metotlarından herhangi biri kullanılarak alınan token bilgisi geçerli.
- **storageid(query)**: Sorgulanacak işin akış idsidir.
- **instanceid(path)**: Sorgulanacak işin guid id bilgisidir. Format: xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxx (UUID).
- **currentbookmark(path)**: Sorgulanacak işin bulunduğu adım guid bilgisidir. Format: xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxx
- **includefileandcomment(query)**: Sorgulanacak işte dosya ve yorum zorunluğunun bulunup bulunmadığının bilgisini döner.

11) /api/workflow/withcriteria

Logo Flow üzerinde belirtilen kriterlerde devam eden işlerin listesini döner.

Input Parameters

- **Authorization(Headers)**: Login metotlarından herhangi biri kullanılarak alınan token bilgisi geçerli.
- **offset(query)**: Gelen verinin kaçınıcı sayfadan itibaren listeleneceği bilgisidir.
- **limit(query)**: Gelen verinin verilen offsete göre kaçar kaçar gösterileceği bilgisidir.
- **sort(query)**: Gelen veride sıralama yapmak için kullanılır. asc,desc
- **withcount(query)**: Toplam kayıt sayısını döndürür. Gereksiz kullanmayınız.
- **keyFilter(body)**: Veriyi filtreli bir şekilde getirmek için kullanılır.

```
{
  "KeyFilters": [
    {
      "Operator": "string" : eq,neq,gt,lt,lte,contains,doesnotcontain
      "Field": "string", : filtre uygulanacak alan
      "Value": {} : filtre uygulanacak alanın değeri
    }
  ],
  "Logic": "string" : birden fazla filtre içeren requestlerde filtrelerin bağlaçları. (and,or)
}
```

12) /api/workflow/withcustomcolumn

Logo Flow üzerinde devam eden işlerin listesini tanımlı özel alanları ile birlikte döner.

Input Parameters

- **Authorization(Headers)**: Login metotlarından herhangi biri kullanılarak alınan token bilgisi geçerli.
- **firstversionid(query)**: İlgili akışın firstversionid bilgisi geçerli.
- **publishType(query)**: Akışın yayın modu başlatılmak isteniyorsa 1, geliştirim modu başlatılmak isteniyorsa 0 gönderilir.
- **offset(query)**: Gelen verinin kaçınıcı sayfadan itibaren listeleneceği bilgisidir.
- **limit(query)**: Gelen verinin verilen offsete göre kaçar kaçar gösterileceği bilgisidir.
- **sort(query)**: Gelen veride sıralama yapmak için kullanılır. asc,desc
- **withcount(query)**: Toplam kayıt sayısını döndürür. Gerekecekçe kullanmayınız.
- **keyFilter(body)**: Veriyi filtreli bir şekilde getirmek için kullanılır.

```
{
  "KeyFilters": [
    {
      "Operator": "string" : eq,neq,gt,lt,lte,contains,doesnotcontain
      "Field": "string", : filtre uygulanacak alan
      "Value": {} : filtre uygulanacak alanın değeri
    }
  ],
  "Logic": "string" : birden fazla filtre içeren requestlerde filtrelerin bağlaçları. (and,or)
}
```