

# API Web app Sogeti :

http://localhost:49197/Service1.svc/JSON

## TASKS

### Get tasks

URL : /Tasks

METHOD : GET

PARAM :

```
?offset=int (number of items to skip for pagination)
&limit=int (number of items to return for pagination)
&sort=string (column to sort by)
&dir=int (1 for ascending order, 0 for descending order)
&id=int (search by task id) OPTIONAL
&label=string (search by task label) OPTIONAL
&location=string (search by task location) OPTIONAL
&priority=string (search by task priority) OPTIONAL
&company=string (search by agent company) OPTIONAL
&job=string (search by agent job) OPTIONAL
&name=string (search by agent name) OPTIONAL
&idA=int (search by agent id) OPTIONAL
&searchG=string (global search on all columns) OPTIONAL
```

RETURN :

```
{
  "tasks" :
    [
      {
        "id" : int, (task id)
        "label" : string, (task name)
        "location" : string, (task location)
        "priority" : string, (task priority)
        "company" : string, (agent company)
        "job" : string, (agent job)
        "name" : string, (agent name)
        "idA" : int (agent id)
      }
    ]
  "total" : int (number total of tasks)
}
```

## Get a specific task

URL : /Tasks/{id}

METHOD : GET

PARAM :

```
id : int (id of the task)
```

RETURN :

```
{
  "id" : int, (task id)
  "label" : string, (task name)
  "location" : string, (task location)
  "priority" : string, (task priority)
  "company" : string, (agent company)
  "job" : string, (agent job)
  "name" : string, (agent name)
  "idA" : int (agent id)
}
```

## Get task filters

URL : /Tasks/Filters

METHOD : GET

PARAM :

```
null
```

RETURN :

```
[
  {
    "name" : string, (name of the filter)
    "type" : string (type of the filter)
  }
]
```

## Add a task

URL : /Tasks

**METHOD :** POST

**PARAM :**

```
{
  "label" : string, (task name)
  "location" : string, (task location)
  "priority" : string, (task priority)
  "idA" : int (agent id)
}
```

**RETURN :** 1 success 0 fail

## Edit a task

**URL :** /Tasks

**METHOD :** PUT

**PARAM :**

```
{
  "id" : int, (task id)
  "label" : string, (task name)
  "location" : string, (task location)
  "priority" : string, (task priority)
  "idA" : int (agent id)
}
```

**RETURN :** 1 success 0 fail

## Delete a task

**URL :** /Tasks/{id}

**METHOD :** DELETE

**PARAM :**

```
id : int (id of the task)
```

**RETURN :** 1 success 0 fail

---

# AGENTS

## Get agents

URL : /Agents

METHOD : GET

PARAM :

```
?offset=int (number of items to skip for pagination)
&limit=int (number of items to return for pagination)
&sort=string (column to sort by)
&dir=int (1 for ascending order, 0 for descending order)
&id=int (search by agent id) OPTIONAL
&FirstName=string (search by agent firstname) OPTIONAL
&LastName=string (search by agent lastname) OPTIONAL
&company=string (search by agent company) OPTIONAL
&job=string (search by agent job) OPTIONAL
&searchG=string (global search on all columns) OPTIONAL
```

RETURN :

```
{
  "agents" :
    [
      {
        "id" : int, (agent id)
        "FirstName" : string, (agent firstname)
        "LastName" : string, (agent lastname)
        "company" : string, (agent company)
        "job" : string (agent job)
      }
    ]
  "total" : int (agents total number)
}
```

## Get a specific agent

URL : /Agents/{id}

METHOD : GET

PARAM :

```
id : int (id of the agent)
```

**RETURN :**

```
{
  "id" : int, (agent id)
  "FirstName" : string, (agent firstname)
  "LastName" : string, (agent lastname)
  "company" : string, (agent company)
  "job" : string (agent job)
}
```

## Get agent filters

**URL :** /Agents/Filters

**METHOD :** GET

**PARAM :**

```
null
```

**RETURN :**

```
[
  {
    "name" : string, (name of the filter)
    "type" : string (type of the filter)
  }
]
```

## Add an agent

**URL :** /Agents

**METHOD :** POST

**PARAM :**

```
{
  "FirstName" : string, (agent firstname)
  "LastName" : string, (agent lastname)
  "company" : string, (agent company)
  "job" : string (agent job)
}
```

**RETURN :** 1 success 0 fail

## Edit an agent

URL : /Agents

METHOD : PUT

PARAM :

```
{  
  "id" : int, (agent id)  
  "FirstName" : string, (agent firstname)  
  "LastName" : string, (agent lastname)  
  "company" : string, (agent company)  
  "job" : string (agent job)  
}
```

RETURN : 1 success 0 fail

## Delete an agent

URL : /Agents/{id}

METHOD : DELETE

PARAM :

```
id : int (id of the agent)
```

RETURN : 1 success 0 fail