Warning Open

ATQ Test Orchestration Director

REST Interface for Test queuing and orchestration

Version

Paths

/databind/list

GET /databind/list

databind

Summary

list databind

Description

List of uploaded and available files

Responses

Code	Description	Schema
200	OK	⇄ ▼**Mediatype identifier: application/atq.databind.upload+json; type=collection; view=default**[*AtqDatabindUploadCollection is the media type for an array of AtqDatabindUpload (default view)* ▶**Mediatype identifier: application/atq.databind.upload+json; view=default** { }]
204	No Content	

Try this operation

/databind/upload

POST /databind/upload

databind

Summary

upload databind

Description

Upload new zipped file for later usage with a Task

Parameters

Name	Located in	Required	Schema

payload	body	Yes	⇄ ▼**UploadPayload {**
			file: ▶ undefined *
			}

Responses

Code	Description	Schema

200	The file was uploaded succesfully	⇄ ▼**Mediatype identifier: application/atq.databind.upload+json; view=default {**
		User upload files response (default view)
		id: ▶ string
		}

415	The file doesn't have a valid extension	⇄ ▼**Mediatype identifier: application/atq.databind.upload+json; view=error {**
		User upload files response (error view)
		error: ▶ string
		}

500	Response when there are an error uploading the file	⇄ ▼**Mediatype identifier: application/atq.databind.upload+json; view=error {**
		User upload files response (error view)
		error: ▶ string
		}

Try this operation

/monitoring/ping

GET /monitoring/ping

monitoring

Summary

ping monitoring

Description

Endpoint for pinging and healthcheck purposes

Responses

Code	Description
200	Pong

Try this operation

/swarm/

GET /swarm/

swarm

Summary

status swarm

Description

Response with the details of the swarm

Responses

Code	Description	Schema
200	Details of the Docker Swarm cluster	⇄ ▼**Mediatype identifier: application/atq.swarm+json; view=default {** *Swarm Details (default view)* joinTokens: ▶**JoinTokens { }** }
503	Docker Swarm context Error Message	⇄ ▼**Mediatype identifier: application/atq.swarm+json; view=error {** *Swarm Details (error view)* error: ▶ string }

Try this operation

/task/

PUT /task/

task

Summary

create task

Description

Creates a new Task in the Swarm according with the config provided in the JSON body

Parameters

Name	Located in	Required		Schema

payload	body	Yes	⇄	▼**TaskPayload {**
				delay: integer
				master: ▶**ServicePayload { }**
				name: ▶ string *
				waitCommand: ▶**WaitCommand { }**
				worker: ▶**ServicePayload { }**
				}

Responses

Code	Description		Schema
200	Task creation in progress	⇄	▼**Mediatype identifier: application/atq.task+json; view=default {** *Task description (default view)* id: ▶ string status: ▶ string }
417	The Task definition has errors or it's not complete	⇄	▼**Mediatype identifier: application/atq.task+json; view=default {** *Task description (default view)* id: ▶ string status: ▶ string }

Try this operation

/task/{id}

```
DELETE /task/{id}
```

task

Summary

delete task

Description

Deletes the Task specified and its components

Parameters

Name	Located in	Required	Schema
id	path	Yes	⇄ `string`

Responses

Code	Description
204	Successfuly deleted
404	The given ID doesn't not exist
500	Docker Engine error deleting the Task generated container infrastructure

Try this operation

GET /task/{id}

task

Summary

inspect task

Description

Get Task's details

Parameters

Name	Located in	Description	Required	Schema
id	path	Task's UUID	Yes	⇄ string

Responses

Code	Description	Schema
200	Successful response containing Task data in JSON format	⇄ ▼**Mediatype identifier: application/atq.task+json; view=default {** *Task description (default view)* id: ▶ string status: ▶ string }
404	The given ID doesn't not exist	
500	Response when the Task has not been created correctly	⇄ ▼**Mediatype identifier: application/atq.task+json; view=default {** *Task description (default view)* id: ▶ string status: ▶ string }

Try this operation

Models

AtqDatabindUpload

```
▼Mediatype identifier: application/atq.databind.upload+json; view=default
{
⇄  User upload files response (default view)
   id: ▶ string
}
```

AtqDatabindUploadCollection

```
   ▼Mediatype identifier: application/atq.databind.upload+json;
  type=collection; view=default[
⇄    AtqDatabindUploadCollection is the media type for an array of
     AtqDatabindUpload (default view)
      ▶Mediatype identifier: application/atq.databind.upload+json; view=default
     { }
  ]
```

AtqDatabindUploadError

```
   ▼Mediatype identifier: application/atq.databind.upload+json; view=error {
     User upload files response (error view)
⇄
     error: ▶ string
   }
```

AtqSwarm

```
   ▼Mediatype identifier: application/atq.swarm+json; view=default {
     Swarm Details (default view)
⇄
     joinTokens: ▶JoinTokens { }
   }
```

AtqSwarmError

```
   ▼Mediatype identifier: application/atq.swarm+json; view=error {
     Swarm Details (error view)
⇄
     error: ▶ string
   }
```

AtqTask

```
   ▼Mediatype identifier: application/atq.task+json; view=default {
     Task description (default view)
⇄    id:     ▶ string
     status: ▶ string
   }
```

JoinTokens

```
   ▼JoinTokens {
     Docker Swarm Join Tokens
⇄    manager: string
     worker:  string
   }
```

ServicePayload

```
   ▼ServicePayload {
     alias:    ▶ string *
     args:     ▶[]
     fileid:   ▶ string
⇄    image:    ▶ string *
```

```
    replicas: ► integer
    tty:      ► boolean
  }
```

TaskPayload

```
  ▼ TaskPayload {
    delay:       integer
    master:      ►ServicePayload { }
⇄   name:        ► string *
    waitCommand: ►WaitCommand { }
    worker:      ►ServicePayload { }
  }
```

UploadPayload

```
  ▼ UploadPayload {
⇄   file: ► undefined *
  }
```

WaitCommand

```
  ▼ WaitCommand {
   Definition of a command to be executed
    command:        ► string
⇄   expectedResult: ► string
    timeout:        ► integer
  }
```