FastAPI [0.1.0] [OAS3]

/openapi.json

default ⌃

GET	/ Root	⌄

POST	/login/ Login	⌃

Login is required to use all functions.

After logging in, a token is generated and it has an expiration date.

The token is deleted when it expires. After it re-login is required.

If send login requests when already logging , expiration date is renewed.

Parameters   [Try it out]

No parameters

Request body required   application/json ⌄

Example Value | Schema

```
{
  "email": "example@example.com",
  "password": "examplepassword"
}
```

Responses

Code	Description	Links
200	Successful Response	*No links*

Media type

application/json ⌄

Controls `Accept` header.

Example Value | Schema

```
{
  "message": "You are already logged in."
}
```

| 422 | Validation Error | *No links* |

Media type

application/json ⌄

Example Value | Schema

```
{
  "detail": [
    {
      "loc": [
        "string",
        0
      ],
      "msg": "string",
      "type": "string"
    }
  ]
}
```

POST	/password-reset/ Forgot Password	⌃

If there is a valid account , the password of that account is changed to a new one .

In order to ensure security , the name information verification of the account is required.

Parameters   [Try it out]

No parameters

Request body ^{required}

```json
{
  "email": "john@example.com",
  "name": "John",
  "new_password": "ExampleNewPassword10"
}
```

Responses

Code	Description	Links
200	Successful Response	No links

Media type

application/json

Controls `Accept` header.

Example Value | Schema

```json
{
  "message": "If your information is correct, your password has been changed."
}
```

Code	Description	Links
422	Validation Error	No links

Media type

application/json

Example Value | Schema

```json
{
  "detail": [
    {
      "loc": [
        "string",
        0
      ],
      "msg": "string",
      "type": "string"
    }
  ]
}
```

POST `/register/` Register

If email doesnt exists , an account is created according to coming parameters

Parameters   Try it out

No parameters

Request body ^{required}   application/json

Example Value | Schema

```json
{
  "email": "example@example.com",
  "password": "examplepassword",
  "name": "John",
  "surname": "Joe"
}
```

Responses

Code	Description	Links
200	Successful Response	No links

Media type

application/json

Controls `Accept` header.

Example Value | Schema

```
{
  "message": "Account created."
}
```

422   Validation Error

Media type

| application/json   ▼ |

Example Value | Schema

```
{
  "detail": [
    {
      "loc": [
        "string",
        0
      ],
      "msg": "string",
      "type": "string"
    }
  ]
}
```

POST **/create-project/** Create Project   ⌃

It creates project. Project must have some features such as :

:project_name

:status : 'active' or 'archived'. Otherwise the project will not be created

:owners : it contains emails as a string in a list . If email doesnt have @ symbol the project will not be created

:viewers : it contains emails as a string in a list . If email doesnt have @ symbol the project will not be created

When the project will be saved, utc time uses for 'created' value in database.

Parameters   Try it out

Name	Description
secret string (cookie)	secret

Request body required   application/json ▼

Example Value | Schema

```
{
  "project_name": "Example Project Name",
  "status": "active",
  "owners": [
    "john@example.com",
    "marry@example.com"
  ],
  "viewers": [
    "jessica@example.com"
  ]
}
```

Responses

Code	Description	Links
200	Successful Response	No links

Media type

| application/json   ▼ |

Controls `Accept` header.

Example Value | Schema

```
{
  "message": "Project created."
}
```

422   Validation Error

Media type

| application/json   ▼ |

Example Value | Schema

```
{
  "detail": [
    {
      "loc": [
        "string",
        0
      ],
      "msg": "string",
      "type": "string"
    }
  ]
}
```

POST `/create-task/` Create Task ⌃

It creates task under the project .

Cant find task with same name on same project.

Only project owners can create task under the project.

Parameters

<div align="right">

Try it out

</div>

Name	Description
secret **string** *(cookie)*	secret

Request body required   application/json ⌄

Example Value | Schema

```
{
  "project_name": "Example Project Name",
  "title": "Example Title",
  "content": "Example Content",
  "owners": [
    "john@example.com",
    "marry@example.com"
  ],
  "viewers": [
    "jessica@example.com"
  ],
  "startdate": "2022-09-21T20:15:10",
  "enddate": "2022-10-21T20:15:10"
}
```

Responses

Code	Description	Links
200	Successful Response	*No links*

Media type

application/json ⌄

Controls `Accept` header.

Example Value | Schema

```
{
  "message": "You do not have  access to create a task"
}
```

| 422 | Validation Error | *No links* |

Media type

application/json ⌄

Example Value | Schema

```
{
  "detail": [
    {
      "loc": [
        "string",
        0
      ],
      "msg": "string",
      "type": "string"
    }
  ]
}
```

GET `/get-{get_type}-{project_or_task}/` Get Owner Projects ⌃

It returns their owner/viewer projects/tasks .

get_type = owners or viewers

project_or_task = projects or tasks

Parameters

<div style="text-align: right;">Try it out</div>

Name	Description
get_type * required string *(path)*	get_type
project_or_task * required string *(path)*	project_or_task
secret string *(cookie)*	secret

Responses

Code	Description	Links
200	Successful Response	*No links*

Media type

application/json

Controls `Accept` header.

Example Value | Schema

```
{
  "message": {
    "0": {
      "project_name": "Example Proje Name",
      "status": "active",
      "created": "2022-09-21T16:57:51.694000",
      "owners": [
        "jessica@example.com",
        "dany@example.com"
      ],
      "viewers": [
        "viewers@gmail.com",
        "lusy@example.com",
        "captain@example.com"
      ]
    },
    "1": {
      "project_name": "Example Proje Name",
      "title": "Example Title",
      "content": "Example Content",
      "status": "Created",
      "created": "2022-09-21T17:02:14.544000",
      "owners": [
        "jessica@example.com",
        "dany@example.com"
      ],
      "viewers": [
        "viewers@gmail.com",
```

| 422 | Validation Error | *No links* |

Media type

application/json

Example Value | Schema

```
{
  "detail": [
    {
      "loc": [
        "string",
        0
      ],
      "msg": "string",
      "type": "string"
    }
  ]
}
```

POST /make-comment/ Make Comment ⌃

User can comment on task under the project but user must be in owners or viewers list in task.

Parameters

<div style="text-align: right;">Try it out</div>

Name	Description
secret string *(cookie)*	secret

Request body ^{required}   application/json ⌄

Example Value | Schema

```
{
  "project_name": "Example Project Name",
  "title": "Example Title",
  "comment": "Example Comment"
}
```

Responses

Code	Description	Links
200	Successful Response	*No links*

Media type

application/json ⌄

Controls `Accept` header.

Example Value | Schema

```
{
  "message": "Comment created."
}
```

| 422 | Validation Error | *No links* |

Media type

application/json ⌄

Example Value | Schema

```
{
  "detail": [
    {
      "loc": [
        "string",
        0
      ],
      "msg": "string",
      "type": "string"
    }
  ]
}
```

GET `/get-comments/{project_name}/{title}` Get Comments   ⌃

Returns all user comments on specific project and task

Parameters   [Try it out]

Name	Description
project_name * required string *(path)*	project_name
title * required string *(path)*	title
secret string *(cookie)*	secret

Responses

Code	Description	Links
200	Successful Response	*No links*

Media type

application/json ▾

Controls Accept header.

Example Value | Schema

```
{
  "message": {
    "message": {
      "1": {
        "project_name": "Example Proje",
        "title": "Example Title",
        "created": "2022-09-21T17:13:35.427000",
        "from": "john@example.com",
        "comment": "example comment 1"
      },
      "2": {
        "project_name": "Example Proje",
        "title": "Example Title",
        "created": "2022-09-21T19:22:09.259000",
        "from": "john@example.com",
        "comment": "example comment 2"
      }
    }
  }
}
```

422   Validation Error   *No links*

Media type

application/json ▾

Example Value | Schema

```
{
  "detail": [
    {
      "loc": [
        "string",
        0
      ],
      "msg": "string",
      "type": "string"
    }
  ]
}
```

Schemas ⌃

Comment ▾ {
    project_name*   string
                       title: Project Name
                       example: Example Project Name
    title*   string
                       title: Title
                       example: Example Title
    comment*   string
                       title: Comment
                       example: Example Comment
}
example: OrderedMap { "project_name": "Example Project Name", "title": "Example Title", "comment": "Example Comment" }

CommentResponse ▾ {
    message*   string
                       title: Message
                       example: Comment created.
}

GetComments ▾ {
    message
                     Message ❯ {...}
}

HTTPValidationError ▾ {
    detail
                     Detail ❯ [...]
}

PasswordReset ▾ {
    email*   string
                       title: Email
                       example: example@example.com
    name*   string
                       title: Name
                       example: John
    new_password*   string
                       title: New Password
                       example: ExampleNewPassword10
}

example: OrderedMap { "email": "john@example.com", "name": "John", "new_password": "ExampleNewPassword10" }

PasswordResetResponse ˅ {
    message*   string
                                  title: Message
                                  example: If your information is correct, your password has been changed.

}

Project ˅ {
    project_name*   string
                                  title: Project Name
                                  example: Example Project Name
    status   string
                                  title: Status
                                  default: active
                                  example: archived
    owners*   Owners ˃ [...]
    viewers*   Viewers ˃ [...]

}
example: OrderedMap { "project_name": "Example Project Name", "status": "active", "owners": List ["john@example.com", "marry@example.com"], "viewers": List ["jessica@example.com"] }

ProjectResponse ˅ {
    message*   string
                                  title: Message
                                  example: Project created.

}

ProjectsDetails ˅ {
    message   Message ˃ {...}

}

Task ˅ {
    project_name*   string
                                  title: Project Name
                                  example: Example Project Name
    title*   string
                                  title: Title
                                  example: Example Title
    content*   string
                                  title: Content
                                  example: Example Content
    owners*   Owners ˃ [...]
    viewers*   Viewers ˃ [...]
    startdate   string($date-time)
                                  title: Startdate
                                example: 2022-09-21T20:15:10
    enddate   string($date-time)
                                  title: Enddate
                                example: 2022-10-21T20:15:10

}

TaskResponse ˅ {
    message*   string
                                  title: Message
                                  example: You do not have access to create a task

}

UserLogin ˅ {
    email*   string
                                  title: Email
                                  example: example@example.com
    password*   string
                                  title: Password
                                  example: examplePassword

}
example: OrderedMap { "email": "example@example.com", "password": "examplepassword" }

UserLoginResponse ˅ {
    message*   string
                                  title: Message
                                  example: You are already logged in.

}

UserRegister ˅ {
    email*   string
                                  title: Email
                                  example: example@example.com
    password*   string
                                  title: Password
                                  example: examplePassword
    name*   string

```
                          title: Name
                          example: John
       surname*           string
                          title: Surname
                          example: John

}
example: OrderedMap { "email": "example@example.com", "password": "examplepassword", "name": "John", "surname": "Joe" }
```

UserRegisterResponse ˅ {
```
       message*           string
                          title: Message
                          example: Account created.

}
```

ValidationError ˅ {
```
       loc*
                          Location > [...]
       msg*               string
                          title: Message
       type*              string
                          title: Error Type

}
```