

FastAPI

0.1.0 OAS3

/openapi.json

default



GET /candidate/{candidate_id} Read Candidate



Fetch the candidate using :

- **id**: Id of the candidate to be fetched

Parameters

Try it out

Name	Description
------	-------------

candidate_id * required

integer
(path)

candidate_id

Responses

Code	Description	Links
------	-------------	-------

200

Successful Response

No links

Media type

application/json

Controls Accept header.

Example Value Schema

```
{  
    "name": "string",  
    "email": "string",  
    "status": "string",  
}
```

Code	Description	Links
422	<p>"id": 0</p> <p>Validation Error</p> <p>Media type</p> <p>application/json</p>	No links

Example Value Schema

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

PUT /candidate/{candidate_id} Update Candidate



Update the candidate using:

- **id**: Id of the candidate to be updated

Parameters

Try it out

Name Description

candidate_id * required

integer
(path)

candidate_id

Request body required

application/json

Example Value Schema

```
{
    "name": "string",
    "email": "string",
```

```
        "status": "string"  
    }  
  
}
```

Responses

Code	Description	Links
200	<p>Successful Response</p> <p>Media type</p> <p>application/json</p> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>"string"</pre>	No links
422	<p>Validation Error</p> <p>Media type</p> <p>application/json</p> <p>Example Value Schema</p> <pre>{ "detail": [{ "loc": ["string", 0], "msg": "string", "type": "string" }] }</pre>	No links

DELETE /candidate/{candidate_id} Delete Candidate ^

Delete the candidate using :

- **candidate_id:** Id of the candidate to be deleted

Parameters

[Try it out](#)

Name	Description
candidate_id * required integer (path)	candidate_id

Responses

Code	Description	Links
200	Successful Response	No links
422	<p>Validation Error</p> <p>Media type</p> <p>application/json</p> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>"string"</pre>	No links

GET /candidates/ Read Candidates ^

Fetch the candidate using :

- **name**: full name of the candidate
- **email**: personal email of the candidate
- **status**: current hiring status of the candidate, Ex: "pre-hire", "active", "inactive"

Parameters**Try it out****Name** **Description**

name
string
(query)

email
string
(query)

status
string
(query)

skip
integer
(query)
Default value : 0

limit
integer
(query)
Default value : 100

Responses

Code	Description	Links
200	Successful Response	No links

Media type

Code	Description	Links
	<p>application/json</p> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>[[{ "name": "string", "email": "string", "status": "string", "id": 0 }]]</pre>	
422	Validation Error	No links
	<p>Media type</p> <p>application/json</p> <p>Example Value Schema</p> <pre>{ "detail": [{ "loc": ["string", 0], "msg": "string", "type": "string" }] }</pre>	

POST /candidate/ Create Candidate ^

Create a candidate with following information:

- **name**: full name of the candidate
- **email**: personal email of the candidate
- **status**: current hiring status of the candidate, Ex: "pre-hire", "active", "inactive"

Parameters

[Try it out](#)

No parameters

Request body required

application/json

Example Value Schema

```
{  
    "name": "string",  
    "email": "string",  
    "status": "string"  
}
```

Responses

Code	Description	Links
201	Successful Response	<i>No links</i>

Media type

application/json

Controls Accept header.

Example Value Schema

```
{  
    "name": "string",  
    "email": "string",  
    "status": "string",  
    "id": 0  
}
```

422	Validation Error	<i>No links</i>
-----	------------------	-----------------

Media type

application/json**Example Value** Schema

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

DELETE /candidate/ Delete Candidate By Email

Delete the candidate using :

- **email**: Email id of the candidate to be deleted

Parameters**Try it out**

Name	Description
------	-------------

email * required

string
(query)

Responses

Code	Description	Links
------	-------------	-------

200

Successful Response

No links

Media type

application/json

Controls Accept header.

Example Value Schema`"string"`

422

Validation Error

No links

Media type

application/json**Example Value** Schema

```
{  
    "detail": [  
        {
```

Code	Description	Links
<pre>"loc": ["string", 0], "msg": "string", "type": "string" }]</pre>		

GET /employee/{employee_id} Read Employee ^

Fetch the employee using :

- **id**: Id of the employee to be fetched

Parameters

[Try it out](#)

Name	Description
employee_id * required integer (path)	employee_id

Responses

Code	Description	Links
200	Successful Response	No links

Media type

application/json

Controls Accept header.

Example Value Schema

```
{
    "name": "string",
    "email": "string",
    "designation": "string",
    "id": 0
}
```

Code	Description	Links
422	Validation Error	No links

Media type

application/json**Example Value** Schema

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

PUT /employee/{employee_id} Update Employee ^

Update the employee using:

- **employee_id**: Id of the employee to be updated

Parameters**Try it out**

Name	Description
------	-------------

employee_id * requiredinteger
(path)**employee_id****Request body** required**application/json****Example Value** Schema

```
{
    "name": "string",
    "email": "string",
    "designation": "string"
}
```

Responses

Code	Description	Links
200	Successful Response	No links
422	Validation Error	No links

Media type

application/json

Controls Accept header.

Example Value Schema

```
"string"
```

DELETE /employee/{employee_id} Delete Employee ^

Delete the employee using :

- **employee_id**: Id of the employee to be deleted

Parameters

[Try it out](#)

Name	Description
employee_id * required integer (path)	employee_id

Responses

Code	Description	Links
200	Successful Response	No links
422	Validation Error	No links

Media type

application/json

Controls Accept header.

Example Value Schema

"string"

Media type

application/json

Example Value Schema

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

GET /employees/ Read Employees



Fetch the employee using :

- **name**: full name of the employee
- **email**: personal email of the employee
- **designation**: current hiring designation of the employee, Ex: "CEO", "Developer", "Designer"

Parameters

[Try it out](#)

Name	Description
name string (query)	<input type="text" value="name"/>
email string (query)	<input type="text" value="email"/>
designation string (query)	<input type="text" value="designation"/>
skip integer (query)	<i>Default value : 0</i> <input type="text" value="0"/>
limit integer (query)	<i>Default value : 100</i> <input type="text" value="100"/>

Responses

Code	Description	Links
200	Successful Response	No links

Media type

application/json

Code	Description	Links
	Controls Accept header.	
	Example Value Schema	
	<pre>[{ "name": "string", "email": "string", "designation": "string", "id": 0 }]</pre>	
422	Validation Error	No links
	Media type	
	application/json	
	Example Value Schema	
	<pre>{ "detail": [{ "loc": ["string", 0], "msg": "string", "type": "string" }] }</pre>	

POST /employee/ Create Employee ^

Create a candidate with following information:

- **name**: full name of the employee
- **email**: personal email of the employee
- **designation**: current hiring designation of the employee, Ex: "CEO", "Developer", "Designer"

Parameters

Try it out

No parameters

Request body required

application/json

Example Value Schema

```
{  
    "name": "string",  
    "email": "string",  
    "designation": "string"  
}
```

Responses

Code	Description	Links
201	Successful Response	<i>No links</i>
422	Validation Error	<i>No links</i>

Media type

application/json

Controls Accept header.

Example Value Schema

```
{  
    "name": "string",  
    "email": "string",  
    "designation": "string",  
    "id": 0  
}
```

Media type

application/json

Example Value Schema

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

DELETE /employee/ Delete Employee By Email



Delete the employee using :

- **email**: Email id of the employee to be deleted

Parameters

[Try it out](#)

Name	Description
------	-------------

email * required

string

(query)

email

Responses

Code	Description	Links
------	-------------	-------

200

Successful Response

No links

Media type

application/json

Controls Accept header.

Example Value Schema

"string"

422

Validation Error

No links

Media type

application/json

Example Value Schema

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

Code	Description	Links																		
	<pre>], "msg": "string", "type": "string" }]</pre>																			
GET	/interview/{interview_id} Read Interview	^																		
	Fetch the interview using :																			
	<ul style="list-style-type: none">• interview_id: Id of the interview to be fetched																			
	Parameters	Try it out																		
	<table border="1"><thead><tr><th>Name</th><th>Description</th></tr></thead><tbody><tr><td>interview_id * required integer (path)</td><td>interview_id</td></tr></tbody></table>	Name	Description	interview_id * required integer (path)	interview_id															
Name	Description																			
interview_id * required integer (path)	interview_id																			
	Responses																			
	<table border="1"><thead><tr><th>Code</th><th>Description</th><th>Links</th></tr></thead><tbody><tr><td>200</td><td>Successful Response</td><td>No links</td></tr><tr><td></td><td>Media type application/json Controls Accept header.</td><td></td></tr><tr><td></td><td>Example Value Schema</td><td></td></tr><tr><td></td><td><pre>{ "round": 0, "candidate_id": 0, "employee_id": 0, "id": 0 }</pre></td><td></td></tr><tr><td>422</td><td>Validation Error</td><td>No links</td></tr></tbody></table>	Code	Description	Links	200	Successful Response	No links		Media type application/json Controls Accept header.			Example Value Schema			<pre>{ "round": 0, "candidate_id": 0, "employee_id": 0, "id": 0 }</pre>		422	Validation Error	No links	
Code	Description	Links																		
200	Successful Response	No links																		
	Media type application/json Controls Accept header.																			
	Example Value Schema																			
	<pre>{ "round": 0, "candidate_id": 0, "employee_id": 0, "id": 0 }</pre>																			
422	Validation Error	No links																		

Code	Description	Links
	<p>Media type</p> <p>application/json</p> <p>Example Value Schema</p> <pre>{ "detail": [{ "loc": ["string", 0], "msg": "string", "type": "string" }] }</pre>	

PUT /interview/{interview_id} Update Interview



Update the interview using:

- **interview_id**: Id of the interview to be updated

Parameters

[Try it out](#)

Name Description

interview_id * required

integer
(path)

interview_id

Request body required

application/json

Example Value Schema

```
{
    "round": 0,
    "candidate_id": 0,
    "employee_id": 0
}
```

Responses

Code	Description	Links
200	Successful Response	No links
422	Validation Error	No links

DELETE /interview/{interview_id} Delete Interview ^

Delete the interview using :

- **interview_id**: Id of the interview to be deleted

Parameters

Try it out

Name	Description
interview_id * required integer (path)	interview_id

Responses

Code	Description	Links
200	Successful Response	No links
422	Validation Error	No links

Media type

application/json

Controls Accept header.

Example Value Schema

"string"

422

Validation Error

Media type

application/json

Example Value Schema

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

GET /interviews/ Read Interviews



Fetch interviews with following information:

- **round**: round number of the interview
- **candidate_id**: Id of the candidate to be interviewed
- **employee_id**: Id of the employee interviewing

Parameters

[Try it out](#)

Name	Description
------	-------------

round
integer
(query)

candidate_id
integer
(query)

employee_id
integer
(query)

skip
integer
(query)

Default value : 0

limit
integer
(query)

Default value : 100

Responses

Code	Description	Links
------	-------------	-------

200 Successful Response No links

Media type

Code	Description	Links
	Controls Accept header.	
	Example Value Schema	
	<pre>[{ "round": 0, "candidate_id": 0, "employee_id": 0, "id": 0 }]</pre>	
422	Validation Error	No links
	Media type	
	application/json	
	Example Value Schema	
	<pre>{ "detail": [{ "loc": ["string", 0], "msg": "string", "type": "string" }] }</pre>	

POST /interview/ Create Interview ^

Create an interview with following information:

- **round**: round number of the interview
- **candidate_id**: Id of the candidate to be interviewed
- **employee_id**: Id of the employee interviewing

Parameters

Try it out

No parameters

Request body required

application/json

Example Value Schema

```
{  
    "round": 0,  
    "candidate_id": 0,  
    "employee_id": 0  
}
```

Responses

Code	Description	Links
201	Successful Response	<i>No links</i>
422	Validation Error	<i>No links</i>

Media type

application/json

Controls Accept header.

Example Value Schema

```
{  
    "round": 0,  
    "candidate_id": 0,  
    "employee_id": 0,  
    "id": 0  
}
```

Media type

application/json

Example Value Schema

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

Schemas



Candidate

CandidateBase

Employee

EmployeeBase

HTTPValidationError

Interview

InterviewBase

ValidationError