

Software Architecture Final Project (Habitica Clone)

Version 0.0.1

Paths

/account/register

POST /account/register

Parameters

Name	Located in	Required	Schema
body	body	Yes	<div><div>▼</div><div>{ email: string name: string }</div></div>

Responses

Code	Description	Schema
200	Success	<div><div>↔</div><div>{ email: string name: string }</div></div>
400	Error	

Try this operation

/account/login

POST /account/login

Parameters

Name	Located in	Required	Schema
body	body	Yes	\Rightarrow ∇ { email: string }

Responses

Code	Description	Schema
200	Success	\Rightarrow ∇ { email: string name: string }

400	Error
-----	-------

[Try this operation](#)

/habit

GET /habit

Description

Retrieves a list of habits for the currently logged in user

Responses

Code	Description	Schema
200	Success	\Rightarrow ∇ [Habit { }]

400	Error
-----	-------

[Try this operation](#)

POST /habit

Description

Create a new habit

Parameters

Name	Located in	Description	Required	Schema
body	body	The habit data	No	<div><div>⇒</div><div><div>▼HabitForm {</div><div>title: string</div><div>description: string</div><div>kind: string</div><div>difficulty: string</div><div>score: number</div><div>}</div></div></div>

Responses

Code	Description	Schema
200	Success	<div><div>⇒</div><div><div>▼Habit {</div><div>id: string</div><div>account_email: string</div><div>title: string</div><div>description: string</div><div>kind: string</div><div>difficulty: string</div><div>score: number</div><div>}</div></div></div>
400	Error	

Try this operation

/habit/{id}

GET /habit/{id}

Description

Retrieves a single habit for the currently logged in user

Parameters

Name	Located in	Required	Schema
id	path	Yes	⇔ string

Responses

Code	Description	Schema
200	Success	<div>⇔ ▼Habit { id: string account_email: string title: string description: string kind: string difficulty: string score: number }</div>
400	Error	

Try this operation

PUT /habit/{id}

Description

Update a habit

Parameters

Name	Located in	Required	Schema
id	path	Yes	⇒ string
<hr/>			
			▼HabitForm {
			title: string
			description: string
body	body	Yes	⇒ kind: string
			difficulty: string
			score: number
			}

Responses

Code	Description	Schema
		▼Habit {
		id: string
		account_email: string
		title: string
200	Success	⇒ description: string
		kind: string
		difficulty: string
		score: number
		}
<hr/>		
400	Error	

Try this operation

DELETE /habit/{id}

Description

Deletes the habit

Parameters

Name	Located in	Required	Schema
id	path	Yes	⇔ string

Responses

Code	Description
204	Success
400	Error

Try this operation

/habit/{id}/do

POST /habit/{id}/do

Description

Mark a habit as done one more time (increase score)

Parameters

Name	Located in	Required	Schema
id	path	Yes	⇒ string

Responses

Code	Description	Schema
200	Success	⇒ <pre>▼Habit { id: string account_email: string title: string description: string kind: string difficulty: string score: number }</pre>
400	Error	

[Try this operation](#)

/task

GET /task

Description

Retrieve tasks of logged in user

Responses

Code	Description	Schema
200	Success	⇒ <pre>▼ [►Task { }]</pre>
400	Error	

[Try this operation](#)

POST /task

Description

Create a new task for logged in user

Parameters

Name	Located in	Required	Schema
body	body	Yes	<div><div>⇒</div><div><div>▼TaskForm {</div><div>title: string</div><div>description: string</div><div>due_date: string</div><div>done_date: string</div><div>reminder_time: string</div><div>reminder_days: number</div><div>}</div></div></div>

Responses

Code	Description	Schema
200	Success	<div><div>⇒</div><div><div>▼Task {</div><div>id: string</div><div>account: string</div><div>title: string</div><div>description: string</div><div>due_date: string</div><div>done_date: string</div><div>reminder_time: string</div><div>reminder_days: string</div><div>}</div></div></div>
400	Error	

Try this operation

/task/{id}

GET /task/{id}

Description

Update a task

Parameters

Name	Located in	Required	Schema
id	path	Yes	⇔ string

Responses

Code	Description	Schema
200	Success	<div>⇔<div>▼Task {<div>id:string account:string title:string description:string due_date:string done_date:string reminder_time:string reminder_days:string</div>}</div></div>
400	Error	

Try this operation

PUT /task/{id}

Description

Update a task

Parameters

Name	Located in	Required	Schema
id	path	Yes	⇒ string
body	body	Yes	⇒ ▼ TaskForm { title: string description: string due_date: string done_date: string reminder_time: string reminder_days: number }

Responses

Code	Description	Schema
200	Success	⇒ ▼ Task { id: string account: string title: string description: string due_date: string done_date: string reminder_time: string reminder_days: string }
400	Error	

Try this operation

DELETE /task/{id}

Description

Deletes the task

Parameters

Name	Located in	Required	Schema
id	path	Yes	⇔ string

Responses

Code	Description
204	Success
400	Error

Try this operation

/task/{id}/do

POST /task/{id}/do

Description

Mark a task as complete

Parameters

Name	Located in	Required	Schema
id	path	Yes	⇔ string

Responses

Code	Description	Schema
200	Success	⇔ ▼Task { id: string account: string title: string description: string due_date: string done_date: string reminder_time: string reminder_days: string }
400	Error	

Try this operation

/reports

GET /reports

Description

Returns a report of habits and tasks of all users

Responses

Code	Description	Schema
200	Success	<pre>⇒ ▼AdminReport { habitsReport: ▶ { } properties: undefined }</pre>
400	Error	<pre>⇒ ▼ErrorResponse { message: string * }</pre>

Try this operation

Models

Habit

```
⇒ ▼Habit {
  id: string
  account_email: string
  title: string
  description: string
  kind: string
  difficulty: string
  score: number
}
```

HabitForm

```
⇒ ▼HabitForm {
  title: string
  description: string
  kind: string
  difficulty: string
  score: number
}
```

Task

```
▼Task {  
  id: string  
  account: string  
  title: string  
  description: string  
⇒ due_date: string  
  done_date: string  
  reminder_time: string  
  reminder_days: string  
}
```

TaskForm

```
▼TaskForm {  
  title: string  
  description: string  
  due_date: string  
⇒ done_date: string  
  reminder_time: string  
  reminder_days: number  
}
```

ErrorResponse

```
▼ErrorResponse {  
⇒ message: string *  
}
```

AdminReport

```
▼AdminReport {  
⇒ habitsReport: ► { }  
  properties: undefined  
}
```

HelloWorldResponse