

TaskFlow Documentation

1. Project Summary

TaskFlow is a smart, minimalist task manager built using Flask (backend), MySQL (database), and HTML/CSS/JavaScript (frontend). It enhances productivity by helping users create, manage, and track their daily tasks efficiently, with seamless user authentication and integrated AI-powered suggestions for task completion. It includes a modern UI with support for dark and light themes, mobile responsiveness, and a GitHub Copilot-style AI assistant panel for enhanced user experience. Created with microservice architecture.

2. Features

Core Features:

- Add, edit, delete, and mark tasks as completed or pending
- Organize tasks by due date and status
- Task sorting, searching
- Pagination Support
- List table allows searching (by default shows 10 entries per page but can be increased)

AI Integration:

AI Feature 1: GitHub Copilot-Style Assistant Panel

Purpose:

Provides contextual help and suggestions to users in natural language, acting like an AI assistant embedded in the interface.

How It Works:

- A floating panel appears at the bottom right of the screen.
- Users can type any question or command into the panel (e.g., “Suggest a task for productivity” or “What’s a good habit to track?”).
- On clicking “Send,” it sends a request to the backend endpoint.
- The backend uses a text-generation AI model (Flan-T5) to return helpful responses.
- The reply is shown in the assistant panel for the user.

AI Feature 2: Inline Task Autocompletion

Purpose:

Helps users by **auto-completing the task title** while typing, reducing effort and improving speed.

How It Works:

- As the user types in the task title input, after 3+ characters, a request is sent to /autocomplete.
- The backend uses an AI model to generate a meaningful task suggestion.
- The suggestion is rendered in real-time as a **ghosted text** (like a hint).
- If the user presses Tab, the suggestion gets filled into the input field.

Day/Night Mode:

- Toggle between day/light mode
- Adaptable to device system and preferences

Mobile Responsive:

- Fully responsive layout using Bootstrap 5
- Optimized for mobile, tablet, and desktop screens

Authentication:

- User registration with secure password hashing (Flask-Bcrypt)
- Login/logout sessions using Flask
- User-specific task data (multi-user system)

3. JWT Authentication

The backend supports JWT-based authentication for secure API access.

- After logging in, a JWT token is issued.
- The token must be included in the Authorization header (as Bearer <token>) for protected endpoints.
- Tokens are verified on every request to secure access to user-specific resources.

4. API Endpoints

Authentication APIs:

1. **/api/register** (POST): {username, email, password} → 201 Created / 400 Bad Request
2. **/api/login** (POST): {email, password} → {token, user_id} / 401 Unauthorized

Task APIs (Require JWT):

1. **/api/tasks** (GET): → List of user's tasks
2. **/api/tasks** (POST): {title, description, due_date} → 201 Created
3. **/api/tasks/<task_id>** (PUT): {title?, description?, status?} → 200 OK / 404 Not Found
4. **/api/tasks/<task_id>** (DELETE): → 204 No Content

AI Integration APIs:

1. `/api/ai/suggest-task` (POST): `{input_text} → {suggestion}`
2. `/api/ai/assistant` (POST): `{context} → AI assistant panel response`

5. Local Deployment

- Install Python packages: ***pip install -r /path/to/requirements.txt***
- Run MySQL server and create the table: <https://github.com/sharmaaditi16/taskflow-integrated/blob/main/tasksmanger.sql>
- Start Flask App: `python app.py`
- Open in browser: <http://localhost:5000>

6. Tech Stack

Frontend:

- HTML5, CSS, JAVASCRIPT
- BOOTSTRAP5

Backend:

- Python(Flask)
- Flask-Bcrypt, Flask-JWT, Flask-CORS

AI Integration/Providers:

- Hugging Face Transformers (for AI task suggestions)
- REST API Integration

Database(RDBMS):

- MySQL

7. Other Tools/Implementation:

- JWT for secure API authentication
- GitHub for version control
- Netlify for frontend deployment
- Vercel for backend deployment

8. Validation with Gladiator

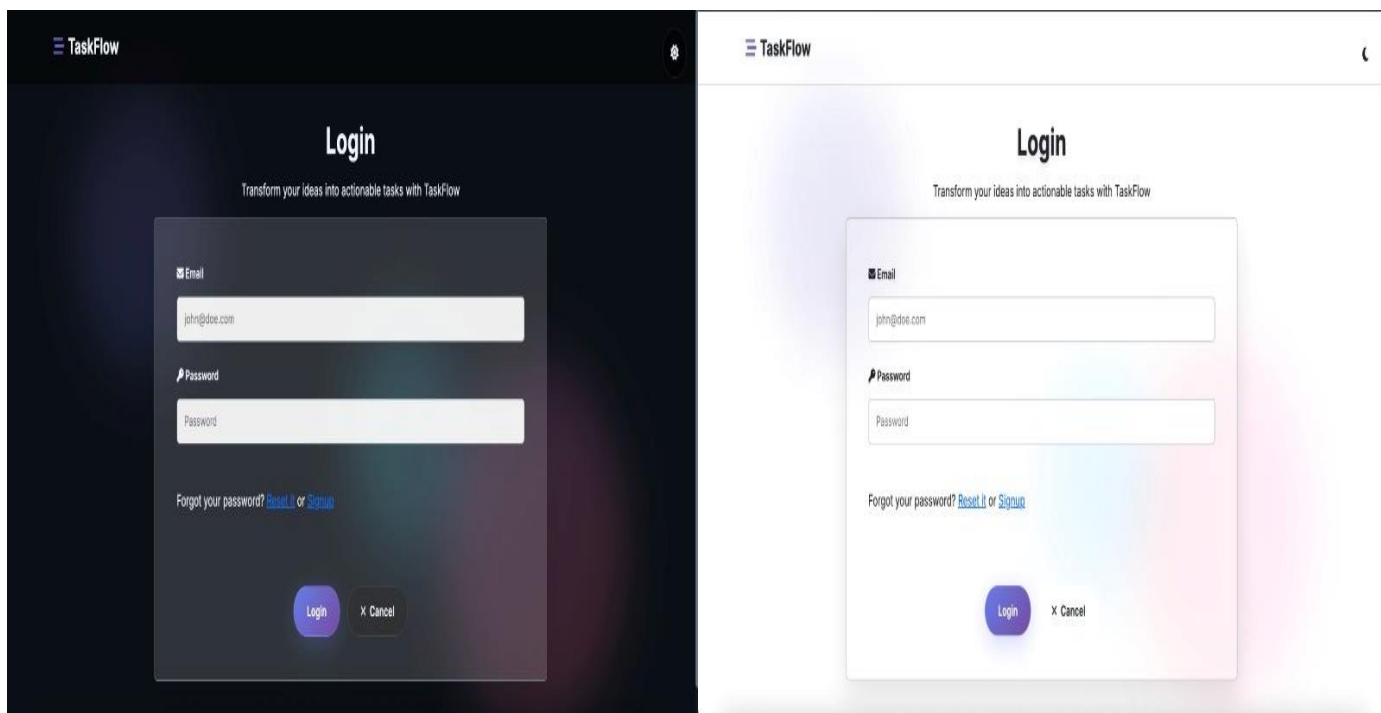
- Ensure that users fill required fields properly before submission — improves data quality and user experience.
- Task validation done using fields: title, description, status, due_date

- User validation done using fields: email, password, username.

9.Important Links

- Github Repo: <https://github.com/sharmaaditi16/taskflow-integrated>
- Frontend deployment(Netlify): <https://taskflowaditi.netlify.app/login.html>
- Backend deployment(Vercel): <https://task-flow-eight-theta.vercel.app/>
- Demo:
<https://drive.google.com/file/d/1AwwRO9yd7THS4ZW49qqmnCai1eu2jgsW/view?usp=sharing>

10. App Screenshots



TaskFlow

Login

Transform your ideas into actionable tasks with TaskFlow

Email

john@doe.com

Password

Password

Forgot your password? [Reset it](#) or [Signup](#)

Login

× Cancel

TaskFlow

Login

Transform your ideas into actionable tasks with TaskFlow

Email

john@doe.com

Password

Password

Forgot your password? [Reset it](#) or [Signup](#)

Login

× Cancel

TaskFlow

8

Signup

Transform your ideas into actionable tasks with TaskFlow

Email

john@doe.com

Username

john.doe

Password

Password

Confirm Password

Password

Already have an account? [Login](#)

Signup

× Cancel

TaskFlow

Signup

Transform your ideas into actionable tasks with TaskFlow

Email

john@doe.com

Username

john.doe

Password

Password

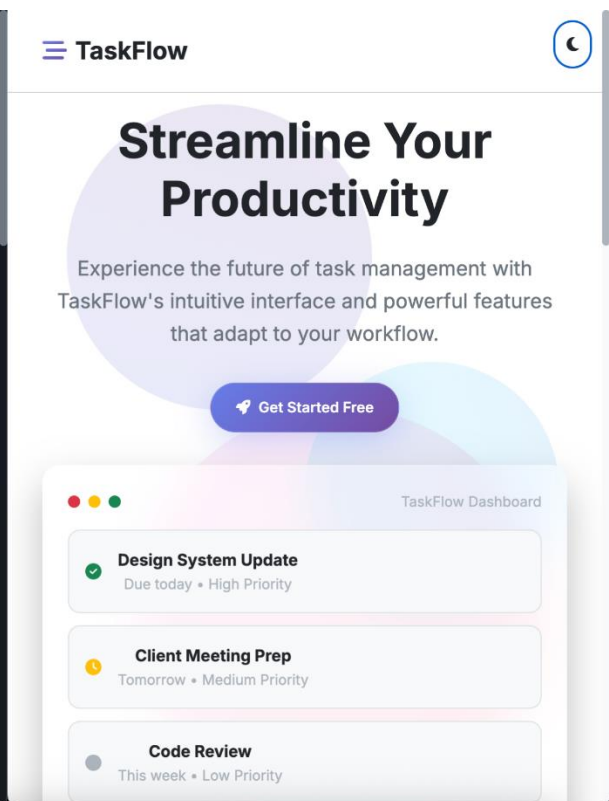
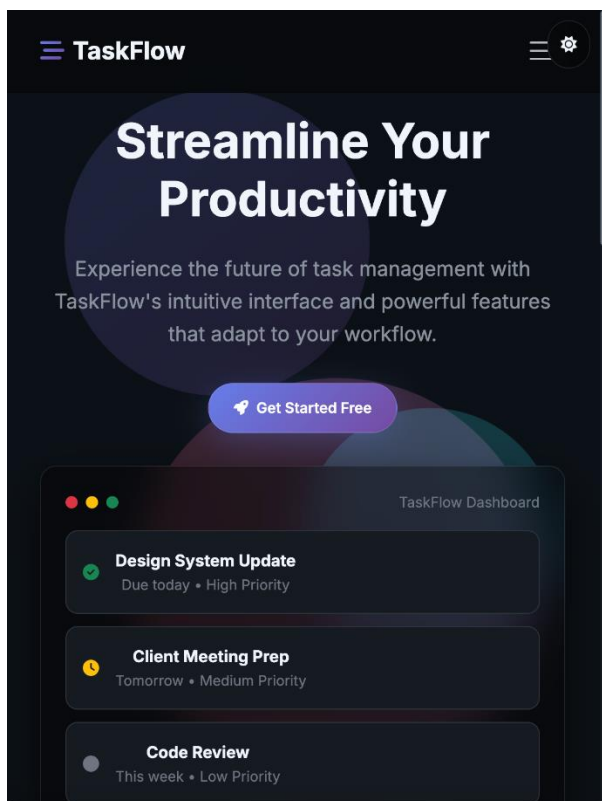
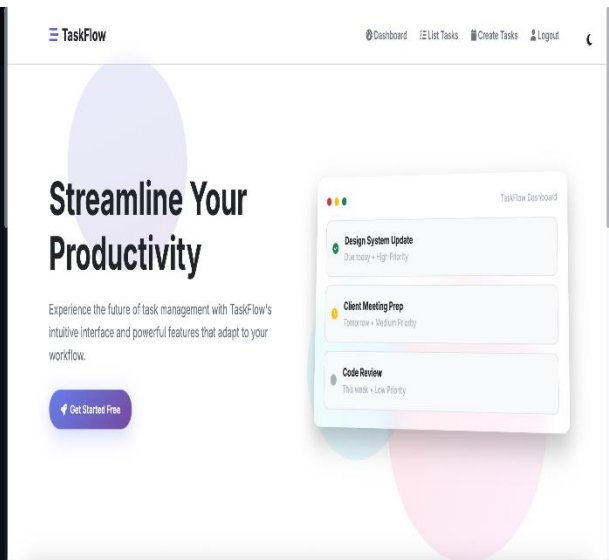
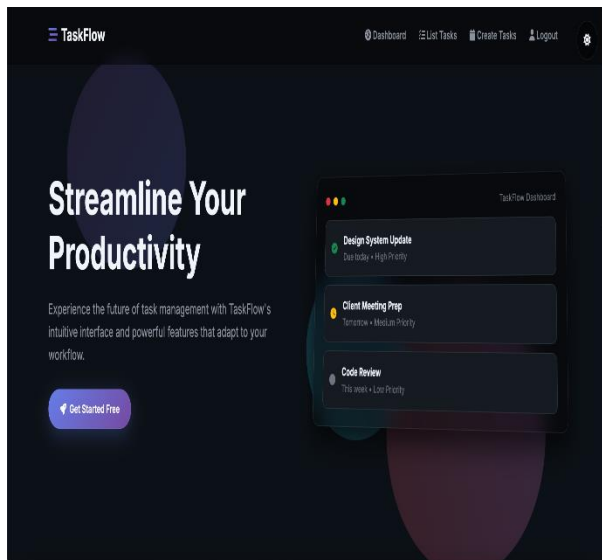
Confirm Password

Password

Already have an account? [Login](#)

Signup

× Cancel



TaskFlow Dashboard List Tasks Create Tasks Logout

10 entries per page Search:

ID	Title	Description	Due Date	Created Date	Actions
23	Book tickets again 2	Check for any issues. 2	Wed, 16 Jul 2025 03:00:00 GMT	2025-06-27T23:12:40	
21	Order supplies new again 21	Coordinate with the team.	Fri, 08 Aug 2025 00:00:00 GMT	2025-06-27T23:12:40	
22	Finish report	Needs urgent attention.	Fri, 29 Aug 2025 00:00:00 GMT	2025-06-27T23:12:40	
24	Read book	Needs management approval.	Sat, 12 Jul 2025 00:00:00 GMT	2025-06-27T23:12:40	
25	Update resume	Check for any issues.	Wed, 20 Aug 2025 00:00:00 GMT	2025-06-27T23:12:40	
26	Submit assignment	Low priority, can be done later.	Fri, 15 Sep 2025 00:00:00 GMT	2025-06-27T23:12:40	
27	Book tickets	Complete the task before deadline.	Sun, 23 Aug 2025 00:00:00 GMT	2025-06-27T23:12:40	
28	Pay bills	Needs urgent attention.	Wed, 09 Jul 2025 00:00:00 GMT	2025-06-27T23:12:40	
29	Clean the house	Low priority, can be done later.	Tue, 05 Aug 2025 00:00:00 GMT	2025-06-27T23:12:40	
30	Fix the bug	Check for any issues.	Sat, 12 Jul 2025 00:00:00 GMT	2025-06-27T23:12:40	
ID	Title	Description	Due Date	Created Date	Actions

Showing 1 to 10 of 447 entries

TaskFlow Dashboard List Tasks Create Tasks Logout

10 entries per page Search:

ID	Title	Description	Due Date	Created Date	Actions
20	Book tickets again 2	Check for any issues. 2	Wed, 16 Jul 2025 03:00:00 GMT	2025-06-27T23:12:40	
21	Order supplies new again 21	Coordinate with the team.	Fri, 08 Aug 2025 00:00:00 GMT	2025-06-27T23:12:40	
22	Finish report	Needs urgent attention.	Fri, 29 Aug 2025 00:00:00 GMT	2025-06-27T23:12:40	
24	Read book	Needs management approval.	Sat, 12 Jul 2025 00:00:00 GMT	2025-06-27T23:12:40	
25	Update resume	Check for any issues.	Wed, 20 Aug 2025 00:00:00 GMT	2025-06-27T23:12:40	
26	Submit assignment	Low priority, can be done later.	Fri, 15 Sep 2025 00:00:00 GMT	2025-06-27T23:12:40	
27	Book tickets	Complete the task before deadline.	Sun, 23 Aug 2025 00:00:00 GMT	2025-06-27T23:12:40	
28	Pay bills	Needs urgent attention.	Wed, 09 Jul 2025 00:00:00 GMT	2025-06-27T23:12:40	
29	Clean the house	Low priority, can be done later.	Tue, 05 Aug 2025 00:00:00 GMT	2025-06-27T23:12:40	
30	Fix the bug	Check for any issues.	Sat, 12 Jul 2025 00:00:00 GMT	2025-06-27T23:12:40	
ID	Title	Description	Due Date	Created Date	Actions

Showing 1 to 10 of 447 entries

API TESTING:

Create User

https://task-flow-eight-theta.vercel.app/create-user Save Share

POST **https://task-flow-eight-theta.vercel.app/create-user** Send

Params Authorization Headers (8) **Body** Scripts Tests Settings Cookies Beautify

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL ☐ JSON

```

1 {
2   "username" : "user4",
3   "email" : "user4@gmail.com",
4   "password" : "password"
5 }
```

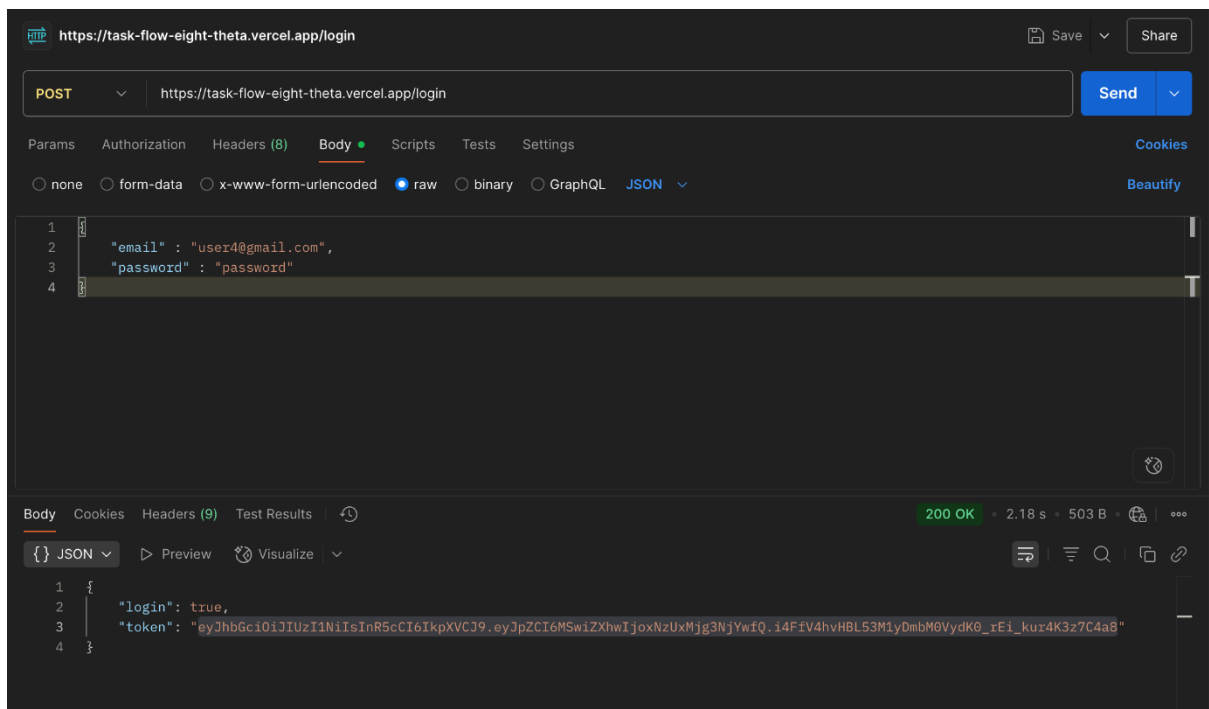
Body Cookies Headers (9) Test Results **200 OK** • 1.41 s • 400 B

JSON Preview Visualize

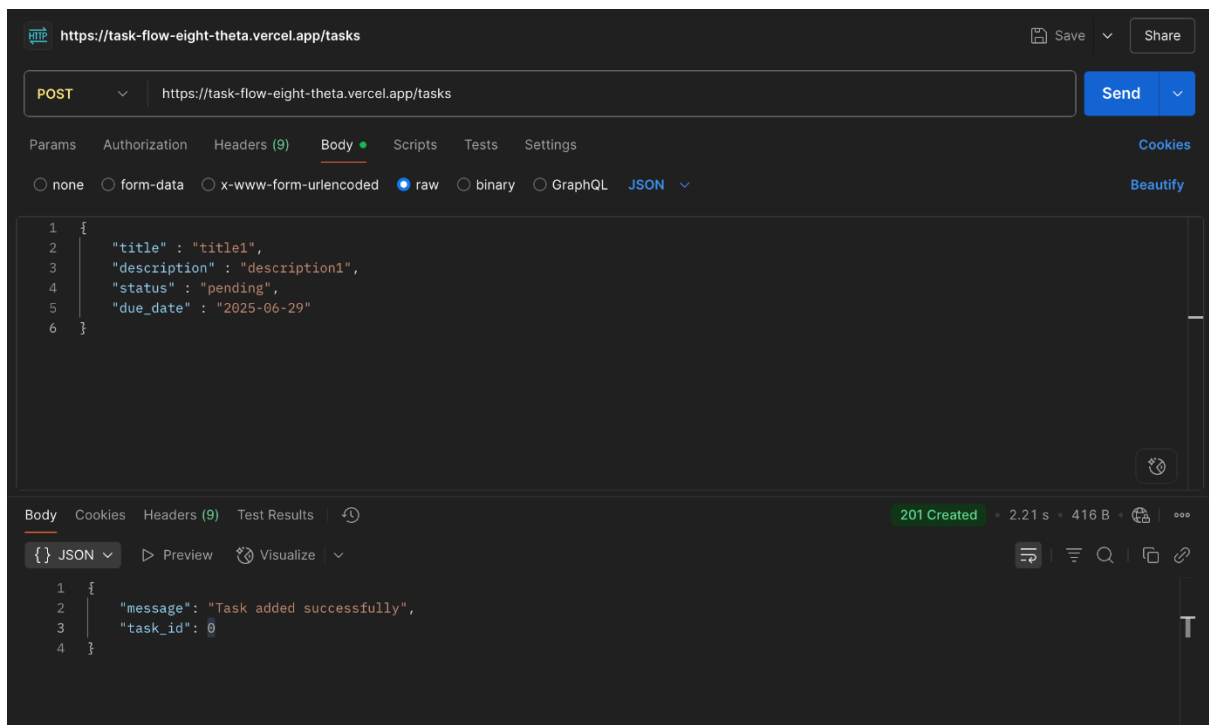
```

1 {
2   "message": "User added successfully!"
3 }
```

User Login



Create Task



Update Task

https://task-flow-eight-theta.vercel.app/tasks

PUT https://task-flow-eight-theta.vercel.app/tasks Send

Params Authorization Headers (9) Body Scripts Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify

```
1 {
2   "id" : 1,
3   "title" : "title3",
4   "description" : "description2",
5   "status" : "pending",
6   "due_date" : "2025-06-29",
7   "created_by" : 4
8 }
```

Body Cookies Headers (9) Test Results 200 OK • 2.21 s • 401 B

{ JSON Preview Visualize

```
1 {
2   "message": "Task updated successfully"
3 }
```

Delete Task

https://task-flow-eight-theta.vercel.app/tasks/5

DELETE https://task-flow-eight-theta.vercel.app/tasks/5 Send

Params Authorization Headers (7) Body Scripts Tests Settings Cookies

Headers 6 hidden


	Key	Value	Description		Bulk Edit	Presets
<input checked="" type="checkbox"/>	token	eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZi6MS...				
	Key	Value	Description		Bulk Edit	Presets

Body Cookies Headers (9) Test Results 200 OK • 2.37 s • 411 B

{ JSON Preview Visualize

```
1 {
2   "message": "Task with ID 5 deleted successfully"
3 }
```

Get Task by ID

 <https://task-flow-eight-theta.vercel.app/tasks/5>

Save

Share

GET

https://task-flow-eight-theta.vercel.app/tasks/5

Send

Params

Authorization

Headers (7)

Body

Scripts

Tests

Settings

Cookies

Headers

6 hidden


	Key	Value	Description	...	Bulk Edit	Presets
<input checked="" type="checkbox"/>	token	eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6MS...				
	Key	Value	Description			

Body

Cookies

Headers (10)

Test Results

200 OK • 2.18 s • 511 B •  ...

{}

JSON

Preview

Visualize

1

2

3

4

5

6

7

8

9

10

{

"task": {

"created_at": 1,

"description": "description1",

"due_date": "Sun, 29 Jun 2025 00:00:00 GMT",

"id": 5,

"status": "pending",

"title": "title1"

}

}