

AstraFlow API Documentation (Comprehensive)

Last Updated: 2025-06-23 20:27:23

This document provides detailed information on all available endpoints, authentication methods, error codes, rate limits, SDK usage, webhooks, and best practices for the AstraFlow RESTful API.

Authentication

AstraFlow uses OAuth 2.0 Bearer Tokens for API authentication. Obtain a token via POST /oauth/token and include it in the Authorization header: Authorization: Bearer <token>.

Rate Limiting

Clients are limited to 100 requests per minute. Exceeding this limit returns HTTP 429. Headers:

X-RateLimit-Limit: Number of requests allowed per window.

X-RateLimit-Remaining: Requests left in the current window.

X-RateLimit-Reset: UTC epoch seconds when limit resets.

Error Codes

Code	HTTP Status	Description
400	Bad Request	Malformed request syntax or invalid parameters
401	Unauthorized	Invalid or missing authentication token
403	Forbidden	Insufficient permissions for this resource
404	Not Found	Resource does not exist
429	Too Many Requests	Rate limit exceeded
500	Internal Server Error	Unexpected server error

Tasks

GET /api/v1/tasks

Retrieve list of tasks with filtering and pagination.

Parameters

Parameter	Type	Description	Required
status	string	Filter tasks by status (active, pending, completed)	No
page	integer	Page number	No
per_page	integer	Items per page	No

Response Schema

Field	Type	Description
-------	------	-------------

tasks	array	List of task objects
id	string	Task identifier
name	string	Task name
status	string	Task status
page	integer	Current page
total	integer	Total tasks

Example Request: GET /api/v1/tasks?status=active&page;=1&per_page=20

Example Response: { "tasks": [{"id":"123","name":"Sample","status":"active"}], "page":1, "total":50 }

POST /api/v1/tasks

Create a new task in the system.

Parameters

Parameter	Type	Description	Required
name	string	Name of the task	Yes
due_date	string	Task due date in ISO 8601	No
priority	string	Task priority (low, medium, high)	No

Response Schema

Field	Type	Description
id	string	Generated task ID
name	string	Task name
status	string	Initial status (pending)
due_date	string	ISO 8601 due date
priority	string	Assigned priority

Example Request: POST /api/v1/tasks { "name":"New", "due_date":"2025-07-01T12:00:00Z", "priority":"high" }

Example Response: {
 "id":"124","name":"New","status":"pending","due_date":"2025-07-01T12:00:00Z","priority":"high" }

GET /api/v1/tasks/{id}

Retrieve details for a specific task.

Parameters

Parameter	Type	Description	Required
id	string	Unique task identifier	Yes

Response Schema

Field	Type	Description
-------	------	-------------

id	string	Task identifier
name	string	Task name
description	string	Task description
status	string	Task status
created_at	string	ISO 8601 timestamp
updated_at	string	ISO 8601 timestamp

Example Request: GET /api/v1/tasks/123

Example Response: { "id":"123","name":"Sample","description":"...", "status":"active","created_at":"2025-01-01T10:00:00Z","updated_at":"2025-06-01T12:30:00Z" }

PUT /api/v1/tasks/{id}

Update properties of an existing task.

Parameters

Parameter	Type	Description	Required
id	string	Task ID	Yes
name	string	New task name	No
status	string	New status	No
due_date	string	Updated due date	No

Response Schema

Field	Type	Description
id	string	Task ID
status	string	Updated status
updated_at	string	Timestamp of update

Example Request: PUT /api/v1/tasks/123 { "status":"completed" }

Example Response: { "id":"123","status":"completed","updated_at":"2025-06-20T15:45:00Z" }

DELETE /api/v1/tasks/{id}

Delete a task permanently.

Parameters

Parameter	Type	Description	Required
id	string	Task ID	Yes

Response Schema

Field	Type	Description
message	string	Confirmation message

Example Request: DELETE /api/v1/tasks/123

Example Response: { "message": "Task deleted successfully" }

Users

GET /api/v1/users

List all users with pagination and filters.

Parameters

Parameter	Type	Description	Required
role	string	Filter by role	No
active	boolean	Filter active users	No

Response Schema

Field	Type	Description
users	array	List of user objects
id	string	User identifier
email	string	User email
created_at	string	Creation timestamp

Example Request: GET /api/v1/users?role=admin&active=true

Example Response: { "users":[{"id":"u123","email":"a@b.com"}],"page":1,"total":10 }

POST /api/v1/users

Create a new user account.

Parameters

Parameter	Type	Description	Required
email	string	User email address	Yes
password	string	Plaintext password	Yes
role	string	Assigned role	No

Response Schema

Field	Type	Description
id	string	New user ID
email	string	User email
created_at	string	Account creation timestamp

Example Request: POST /api/v1/users { "email":"new@af.com","password":"secret" }

Example Response: { "id":"u124","email":"new@af.com","created_at":"2025-06-22T09:15:00Z" }

Webhooks

AstraFlow supports webhooks for real-time event notifications. Register a webhook via POST /api/v1/webhooks with a target URL.

Webhook Registration Parameters

Parameter	Type	Description	Required
url	string	Target endpoint for events	Yes
event	string	Event type (task.created, task.updated)	Yes

SDK Usage Examples

Python SDK Example

```
```python from astraflow_sdk import AstraClient client = AstraClient(token='YOUR_TOKEN_HERE') tasks = client.get_tasks(status='active') print(tasks) ```
```

## JavaScript SDK Example

```
```javascript import AstraClient from 'astraflow-sdk'; const client = new AstraClient('YOUR_TOKEN'); client.getTasks({ status: 'active' }).then(console.log); ```
```