

# API Specification

Task Scheduling System · Spring Boot + Java

Version

1.0

Base URL

http://localhost:8080

## 1. Overview

This document defines the six REST endpoints that form the Task Scheduling API. The API supports full CRUD operations on tasks, enforces a controlled status-transition workflow, and uses soft-deletion so that records are preserved but excluded from standard listings.

## 2. Endpoint Summary

#	Method	Endpoint	Description
1	<b>POST</b>	/tasks	Create a new task
2	<b>GET</b>	/tasks/{id}	Retrieve a task by ID
3	<b>PUT</b>	/tasks/{id}	Update an existing task
4	<b>DELETE</b>	/tasks/{id}	Soft-delete a task
5	<b>GET</b>	/tasks	List all active tasks
6	<b>PATCH</b>	/tasks/{id}/status	Transition a task's status

## 3. Data Model

Every endpoint operates on the Task resource defined below.

Field	Type	Required	Description
<b>id</b>	Long	<b>Auto</b>	System-generated unique identifier
<b>title</b>	String	<b>Yes</b>	Short name of the task (max 100 characters)
<b>description</b>	String	<b>No</b>	Optional longer description
<b>priority</b>	Enum	<b>Yes</b>	LOW, MEDIUM, or HIGH
<b>status</b>	Enum	<b>Auto</b>	PENDING, IN_PROGRESS, COMPLETED, or CANCELLED
<b>deleted</b>	Boolean	<b>Auto</b>	Soft-delete flag; defaults to false
<b>createdAt</b>	Instant	<b>Auto</b>	Timestamp set on creation
<b>updatedAt</b>	Instant	<b>Auto</b>	Timestamp updated on every change

✉ communications@rbihub.in · ☎ 080-22581122 · <https://rbihub.in>

GSTN: 29AAKCR9018A1ZB · CIN: U72900KA2021NPL178293

### Registered office

Keonics-K wing, 4th Floor, 27th Main, Sector 1, 7th Cross Road, HSR Layout, Bengaluru-560102

भारतीय रिजर्व बैंक की पूर्ण स्वामित्व वाली सहायक कंपनी

A wholly-owned subsidiary of the Reserve Bank of India

## 4. Status Transition Rules

The status field is not freely editable. Transitions are only permitted along the paths shown below. Any other transition must be rejected with a 400 Bad Request.

Current Status	Allowed Next Status	Allowed?	Notes
PENDING	IN_PROGRESS	Yes	Normal start of work
PENDING	CANCELLED	Yes	Task no longer needed
IN_PROGRESS	COMPLETED	Yes	Work finished successfully
IN_PROGRESS	CANCELLED	Yes	Abandoned mid-work
COMPLETED	—	No	Terminal state; no further transitions
CANCELLED	—	No	Terminal state; no further transitions

## 5. Endpoint Details

### 5.1 Create a New Task

**POST**
**/tasks**

Create a new task

Creates a task with the given title, description, and priority. The status is automatically set to PENDING and the deleted flag to false.

Request Body	Response Body (201 Created)
<pre>{   "title": "Implement login API",   "description": "Add JWT auth",   "priority": "HIGH" }</pre>	<pre>{   "id": 1,   "title": "Implement login API",   "description": "Add JWT auth",   "priority": "HIGH",   "status": "PENDING",   "deleted": false,   "createdAt": "2025-01-15T10:30:00Z",   "updatedAt": "2025-01-15T10:30:00Z" }</pre>

### Validation & Error Cases

Condition	HTTP Status	Error Message (example)
Title is missing or empty	400	"title is required"

Title exceeds 100 characters	<b>400</b>	"title must be at most 100 characters"
Priority is missing or invalid value	<b>400</b>	"priority must be LOW, MEDIUM, or HIGH"

## 5.2 Retrieve a Task by ID

<b>GET</b>	<b>/tasks/{id}</b>	Get a single task
------------	--------------------	-------------------

Returns the full task object for the given ID. If the ID does not exist or the task has been soft-deleted, the endpoint returns 404.

Path Parameter	Response Body (200 OK)
GET /tasks/1	{     "id": 1,     "title": "Implement login API",     "description": "Add JWT auth",     "priority": "HIGH",     "status": "PENDING",     "deleted": false,     "createdAt": "2025-01-15T10:30:00Z",     "updatedAt": "2025-01-15T10:30:00Z"   }

## Error Cases

Condition	HTTP Status	Error Message (example)
Task ID does not exist	<b>404</b>	"Task not found with id: 99"
Task exists but is soft-deleted	<b>404</b>	"Task not found with id: 5"

## 5.3 Update an Existing Task

<b>PUT</b>	<b>/tasks/{id}</b>	Update a task's fields
------------	--------------------	------------------------

Allows updating title, description, and priority. The status field must not be changed via this endpoint — use the dedicated PATCH /tasks/{id}/status endpoint instead. The updatedAt timestamp is refreshed automatically.

Request Body	Response Body (200 OK)
{     "title": "Implement login API v2",     "description": "Add OAuth 2.0",   }	{     "id": 1,     "title": "Implement login API v2",   }

✉ communications@rbihub.in • ☎ 080-22581122 • <https://rbihub.in>

GSTN: 29AAKCR9018A1ZB • CIN: U72900KA2021NPL178293

Registered office

Keonics-K wing, 4th Floor, 27th Main, Sector 1, 7th Cross Road, HSR Layout, Bengaluru-560102

भारतीय रिजर्व बैंक की पूर्ण स्वामित्व वाली सहायक कंपनी  
 A wholly-owned subsidiary of the Reserve Bank of India

<pre>     "priority": "MEDIUM"   }   </pre>	<pre>     "description": "Add OAuth 2.0",     "priority": "MEDIUM",     "status": "PENDING",     "deleted": false,     "createdAt": "2025-01-15T10:30:00Z",     "updatedAt": "2025-01-15T11:45:00Z"   }   </pre>
---	--

## Validation & Error Cases

Condition	HTTP Status	Error Message (example)
Task ID does not exist	404	"Task not found with id: 99"
Title is empty after update	400	"title is required"
Attempting to change status via this endpoint	400	"Use PATCH /tasks/{id}/status"
Invalid priority value	400	"priority must be LOW, MEDIUM, or HIGH"

## 5.4 Soft-Delete a Task

<b>DELETE</b> /tasks/{id}	Soft-delete a task
---------------------------	--------------------

Sets the deleted flag to true on the task record. The row is never physically removed from the data store. Soft-deleted tasks are invisible to GET /tasks (list) and GET /tasks/{id}.

Request	Response Body (200 OK)
<pre> DELETE /tasks/1 // No request body required   </pre>	<pre> {   "id": 1,   "title": "Implement login API",   "description": "Add JWT auth",   "priority": "HIGH",   "status": "PENDING",   "deleted": true,   "createdAt": "2025-01-15T10:30:00Z",   "updatedAt": "2025-01-15T12:00:00Z" }   </pre>

## Error Cases

Condition	HTTP Status	Error Message (example)
Task ID does not exist	404	"Task not found with id: 99"
Task is already soft-deleted	404	"Task not found with id: 5"

## 5.5 List All Active Tasks

**GET**
**/tasks**

Return all non-deleted tasks

Returns an array of all tasks where deleted is false. Soft-deleted tasks must be excluded from this response.  
 Returns an empty array if no active tasks exist — not an error.

Request	Response Body (200 OK)
<pre>GET /tasks // No parameters required</pre>	<pre>[   {     "id": 1,     "title": "Implement login API",     "priority": "HIGH",     "status": "PENDING",     "deleted": false, ...   },   {     "id": 2,     "title": "Write unit tests",     "priority": "MEDIUM",     "status": "IN_PROGRESS",     "deleted": false, ...   } ]</pre>

**Note:** A task with `deleted: true` will never appear in this list, regardless of its status.

## 5.6 Update Task Status

**PATCH**
**/tasks/{id}/status**

Transition status only

This is the only endpoint that may change a task's status. The requested transition is validated against the allowed state machine (see Section 4). Only the status and updatedAt fields are affected.

Request Body	Response Body (200 OK)
<pre>{   "status": "IN_PROGRESS" }</pre>	<pre>{   "id": 1,   "title": "Implement login API",   "description": "Add JWT auth",   "priority": "HIGH",   "status": "IN_PROGRESS",   "deleted": false,   "createdAt": "2025-01-15T10:30:00Z",   "updatedAt": "2025-01-15T14:20:00Z" }</pre>

## Validation & Error Cases

✉ [communications@rbihub.in](mailto:communications@rbihub.in) • ☎ 080-22581122 • <https://rbihub.in>

GSTN: 29AAKCR9018A1ZB • CIN: U72900KA2021NPL178293

### Registered office

Keonics-K wing, 4th Floor, 27th Main, Sector 1, 7th Cross Road, HSR Layout, Bengaluru-560102

भारतीय रिजर्व बैंक की पूर्ण स्वामित्व वाली सहायक कंपनी  
 A wholly-owned subsidiary of the Reserve Bank of India

Condition	HTTP Status	Error Message (example)
Task ID does not exist	<b>404</b>	"Task not found with id: 99"
Status value is not a valid enum	<b>400</b>	"status must be a valid value"
Transition is not permitted (e.g. PENDING → COMPLETED)	<b>400</b>	"Cannot transition from PENDING to COMPLETED"
Transition from a terminal state	<b>400</b>	"Cannot transition from COMPLETED"

## 6. Error Response Format

All error responses must share a consistent JSON structure so that clients can parse them uniformly.

Error Response Schema
<pre>{   "error": "&lt;short error type&gt;",   "message": "&lt;human-readable explanation&gt;" }</pre>

HTTP Code	error value	When to use
<b>400</b>	<b>BAD_REQUEST</b>	Malformed input, validation failure, or illegal state transition
<b>404</b>	<b>NOT_FOUND</b>	Resource ID does not exist or has been soft-deleted

## 7. Evaluation Criteria

Submissions will be assessed on the following dimensions, in no particular order of priority.

#	Criteria	What we look for
1	<b>Layered Architecture</b>	Clean separation of Controller → Service → Repository. No business logic in the controller.
2	<b>State Machine Logic</b>	Correct enforcement of allowed transitions; clear rejection of invalid ones.
3	<b>Soft-Delete Handling</b>	Deleted tasks are excluded from listings and treated as non-existent on direct lookup.
4	<b>Input Validation</b>	All required fields are checked; enums are validated before processing.
5	<b>Error Consistency</b>	Every error follows the standard shape with an appropriate HTTP status code.

**6 Code Readability**

Meaningful names, concise methods, and logical file/package organisation.

## 8. Technical Notes

**Framework:** Spring Boot (latest stable). Use `@RestController`, `@Service`, and a Repository layer.

**Data Store:** An in-memory store (e.g. a `ConcurrentHashMap`) is acceptable. SQLite via Spring Data JPA is also fine.

**Timestamps:** Use `java.time.Instant` and ISO-8601 formatting in JSON responses.

**ID Generation:** If using an in-memory store, a simple auto-incrementing `Long` counter is sufficient.

**Tests:** Please add unit tests and ensure there is good coverage covering all cases as mentioned above.