# 2.7 Finalize API Contracts and Handle Edge Cases

## 🎯 Objective

Define and finalize REST API contracts for task scheduling and status tracking, ensuring all endpoints are clearly documented, well-typed, and exposed via Swagger/OpenAPI. Also, handle edge cases such as invalid inputs, missing fields, and unsupported operations.

## 🔍 Why This Matters

Finalizing API contracts ensures:  
- Developers and clients know exactly how to interact with your service.  
- The system is robust against bad inputs and usage patterns.  
- MCA evaluation includes correctness, clarity, and usability of interfaces.

## 📑 APIs to Finalize

|  |  |  |
| --- | --- | --- |
| Method | Endpoint | Description |
| POST | /tasks | Create and submit a new task |
| GET | /tasks/{tenantId}/{taskId} | Retrieve status of a specific task |
| GET | /tasks/status/{tenantId} | (Optional) List tasks with filters |
| POST | /tasks/retry/{taskId} | (Future) Retry a failed task manually |

## ✅ Steps to Implement

1. Use @RestControllerAdvice for Global Error Handling

* Handle invalid JSON, missing fields, invalid IDs, and return clean 400/404/500 errors with structured JSON.

{  
 "error": "Task not found",  
 "status": 404,  
 "timestamp": "2025-09-29T10:00:00Z"  
}

1. 2. Add Field Validations with @Valid

* Use annotations like @NotNull, @Min, @Size in your Task DTO.

1. 3. Document Using Swagger (SpringDoc / OpenAPI)

* Use springdoc-openapi-ui dependency. Swagger UI will expose all APIs live via /swagger-ui.html or /v3/api-docs.

1. 4. Handle Edge Cases

|  |  |
| --- | --- |
| Case | Handling Strategy |
| Submitting task with missing ID | Return 400 Bad Request with error details |
| Invalid tenant/task ID in GET | Return 404 Not Found |
| Retry count exceeds maxRetries | Auto mark task as DLQ (already implemented) |
| Dependency cycle | Handled in DAG validator (DagUtils) |
| Submitting task with unknown status | Reject via enum validation / default to PENDING |

## 📌 Outcome

By completing this step, you will:  
- Finalize your public API interface  
- Be ready to write Postman collections or API manuals  
- Meet key MCA project quality criteria