# Document Upload API Documentation

## Base URL

http://localhost:2222/api/documents

## 1. ****Upload Document****

### Endpoint:

POST /api/documents/upload

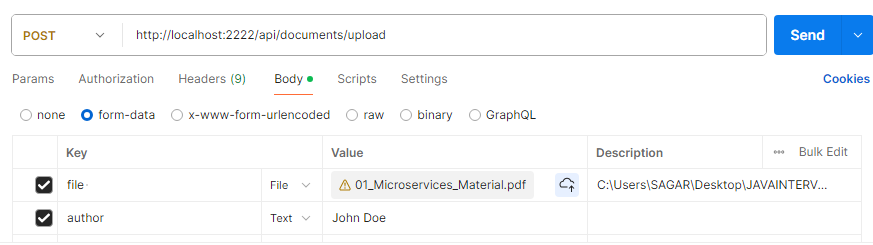
### Description:

This endpoint allows users to upload a document. The document will be ingested, and the author information will be stored.

### Request Parameters:

* **file** (MultipartFile): The document file to be uploaded (required).
* **author** (String): The name of the author of the document (required).

### Request Body Example:



### Response:

* **Status Code**: 200 OK
* **Body**: The response contains the document details, including an ID, file name, upload date, and author.

Example:

json

{

"id": 1,

"fileName": "document.pdf",

"author": "John Doe",

"uploadDate": "2025-04-14T10:15:30",

"fileSize": 5242880

}

### Error Responses:

* **400 Bad Request**: If the file or author is missing.
* **500 Internal Server Error**: If there is an error processing the file.

## 2. ****Filter Documents by Author****

### Endpoint:

GET /api/documents/filter

### Description:

This endpoint filters and retrieves documents based on the author's name.

### Request Parameters:

* **author** (String): The name of the author to filter by (required).

### Example Request:

GET /api/documents/filter?author=John

### Response:

* **Status Code**: 200 OK
* **Body**: A list of documents that match the given author.

Example:

json

[

{

"id": 1,

"fileName": "document1.pdf",

"author": "John Doe",

"uploadDate": "2025-04-14T10:15:30",

"fileSize": 5242880

},

{

"id": 2,

"fileName": "document2.pdf",

"author": "John Doe",

"uploadDate": "2025-04-13T11:20:45",

"fileSize": 1048576

}

]

### Error Responses:

* **400 Bad Request**: If the author parameter is missing or invalid.
* **404 Not Found**: If no documents are found for the given author.

## 3. ****Get Documents with Pagination and Sorting****

### Endpoint:

GET /api/documents

### Description:

This endpoint retrieves all documents. You can filter by author, paginate, and sort the results.

### Request Parameters:

* **author** (String, optional): Filter documents by a specific author (optional).
* **page** (int, default: 0): The page number to retrieve (default is 0).
* **size** (int, default: 10): The number of documents per page (default is 10).
* **sort** (String[], default: uploadDate,desc): The sorting criteria (e.g., uploadDate,asc or fileName,desc).

### Example Request:

GET /api/documents?page=1&size=5&sort=uploadDate,asc

### Response:

* **Status Code**: 200 OK
* **Body**: A paginated list of documents with sorting applied.

Example:

json

{

"content": [

{

"id": 1,

"fileName": "document1.pdf",

"author": "John Doe",

"uploadDate": "2025-04-14T10:15:30",

"fileSize": 5242880

},

{

"id": 2,

"fileName": "document2.pdf",

"author": "John Doe",

"uploadDate": "2025-04-13T11:20:45",

"fileSize": 1048576

}

],

"pageable": {

"sort": {

"sorted": true,

"unsorted": false,

"empty": false

},

"pageSize": 5,

"pageNumber": 1,

"offset": 5,

"unpaged": false,

"paged": true

},

"totalPages": 2,

"totalElements": 10,

"last": false,

"first": false,

"numberOfElements": 5

}

### Error Responses:

* **400 Bad Request**: If the sort parameter is invalid.
* **404 Not Found**: If no documents match the given filters.

## 4. ****Get Document by ID****

### Endpoint:

GET /api/documents/{id}

### Description:

This endpoint retrieves a specific document by its ID.

### Path Parameter:

* **id** (Long): The ID of the document to retrieve (required).

### Example Request:

GET /api/documents/1

### Response:

* **Status Code**: 200 OK
* **Body**: The details of the document.

Example:

json

{

"id": 1,

"fileName": "document.pdf",

"author": "John Doe",

"uploadDate": "2025-04-14T10:15:30",

"fileSize": 5242880

}

### Error Responses:

* **404 Not Found**: If the document with the given ID does not exist.