Image Management Application API Specification

This specification details the APIs to support the upload, view, and delete operations for an application that manages images. The images are stored in an Oracle database and uploaded to AWS S3, with CRUD operations supported on the metadata.

# 1. Upload Image API

Endpoint: POST /images/upload

Description: Uploads a batch of images from the Oracle database to AWS S3. Updates the database with S3 URLs and upload statuses.

Request Payload: Not applicable (batch upload triggered internally or by scheduled job).

Response:

{  
 "batchId": "12345",  
 "status": "SUCCESS",  
 "uploadedCount": 100,  
 "failedCount": 0,  
 "failedRecords": []  
}

Notes:

- Images are fetched from the database.

- Failed records (if any) are logged for reprocessing.

# 2. View All Metadata API

Endpoint: GET /images

Description: Fetches metadata for all images stored in the Oracle database. Supports pagination and sorting.

Query Parameters:

| Parameter | Type | Required | Description |  
|--------------|--------|----------|------------------------------|  
| `page` | int | No | Page number (default: 0) |  
| `size` | int | No | Page size (default: 10) |  
| `sortBy` | string | No | Sort field (default: isiDate)|  
| `order` | string | No | Sort order (`asc` or `desc`) |

Response:

{  
 "page": 0,  
 "size": 10,  
 "totalElements": 1000,  
 "totalPages": 100,  
 "data": [  
 {  
 "isiBarCode": "123456789",  
 "isiFileType": "JPEG",  
 "isiFileSizeKb": 2048,  
 "isiFileCompressed": false,  
 "isiDateCreated": "2024-12-05T10:00:00Z",  
 "s3Url": "https://s3.amazonaws.com/bucketname/image1.jpg"  
 },  
 ...  
 ]  
}

Notes:

- The `s3Url` field provides a link to the uploaded image in S3.

# 3. View Single Image Metadata API

Endpoint: GET /images/{isiBarCode}

Description: Fetches metadata for a specific image using its barcode.

Path Parameters:

| Parameter | Type | Description |  
|----------------|--------|----------------------------|  
| `isiBarCode` | string | Unique identifier (barcode)|

Response:

{  
 "isiBarCode": "123456789",  
 "isiFileType": "JPEG",  
 "isiFileSizeKb": 2048,  
 "isiFileCompressed": false,  
 "isiDateCreated": "2024-12-05T10:00:00Z",  
 "s3Url": "https://s3.amazonaws.com/bucketname/image1.jpg"  
}

Notes:

- If the image metadata is not found, the API returns a `404 Not Found` status.

# 4. Delete Image API

Endpoint: DELETE /images/{isiBarCode}

Description: Deletes an image and its metadata from both the Oracle database and S3.

Path Parameters:

| Parameter | Type | Description |  
|----------------|--------|----------------------------|  
| `isiBarCode` | string | Unique identifier (barcode)|

Response:

{  
 "message": "Image successfully deleted.",  
 "isiBarCode": "123456789"  
}

Notes:

- Ensures atomic deletion of the image metadata and S3 object.

- Returns a `404 Not Found` if the image does not exist.

# 5. Update Metadata API

Endpoint: PUT /images/{isiBarCode}

Description: Updates metadata for a specific image in the Oracle database.

Path Parameters:

| Parameter | Type | Description |  
|----------------|--------|----------------------------|  
| `isiBarCode` | string | Unique identifier (barcode)|

Request Payload:

{  
 "isiFileType": "PNG",  
 "isiFileCompressed": true,  
 "isiDateCreated": "2024-12-06T10:00:00Z"  
}

Response:

{  
 "message": "Metadata successfully updated.",  
 "isiBarCode": "123456789"  
}

Notes:

- Validate all fields before updating.

- Return a `404 Not Found` if the image does not exist.