

Tool Catalogue RESTful API

Quick recap of Catalogue Requirements

Functional Requirements

- The page must display grid of tools.
- Tools must be displayed as image, name and price.
- User must be able to filter tools by categories.
- User should be able to “make tool favorite”.
- Page should have simple navigation menu.

Non-Functional Requirements

- Design should be complete and unambiguous.
- Proper HTML tags should be used.
- Classes should be used inside html.
- CSS must have different file.

Entities

Tool Catalogue must have 3 entities:

- Tool itself, with attributes – id, name, price, image_uri, category, is_favorite, description.
- Category – id, name.
- User – id, username, password, email.

Functional Requirements of API

Must be able to:

- View all tools with sorting, filtering and pagination options.
- View a single tool.
- Create a new tool.

- Update existing tool.
- Delete a tool.
- View all categories.
- Add a new category.
- Update category.
- Delete category.
- Mark tool as favorite.
- Unmark tool as favorite.

Non-Functional Requirements

- REST principles should apply.
- Proper HTTP codes should be used.
- Authentication should be done using tokens(JWT).
- Caching should be used where appropriate.

Richardson Maturity Model

Separate URLs will be used for different resources. HTTP verbs will be used properly. Responses will include hypermedia controls.

API Design

/api/tools

HTTP Method	URL	Brief Description	Response Codes
GET	/api/tools	Lists all tools	200
GET	/api/tools/{id}	Get tool by id	200, 404
POST	/api/tools	Create a new tool	201, 400
PUT	/api/tools/{id}	Replace a tool	200, 400, 404
PATCH	/api/tools/{id}	Update part of tool	200, 400, 404
DELETE	/api/tools/{id}	Delete a tool	204, 404

Some supported query parameters:

- ?category={..}
- ?min_price={..}&max_price={..}
- ?page={..}&size={..}
- ?sort={..}

/api/categories

HTTP Method	URL	Brief Description	Response Codes
GET	/api/categories	Lists all categories	200
GET	/api/categories/{id}	Get category by id	200, 404
POST	/api/categories	Create a new categorie	201, 400
PUT	/api/categories/{id}	Replace a category	200, 400, 404
DELETE	/api/categories/{id}	Deletes a category	204, 404

/api/tools/{id}/favorite

HTTP Method	URL	Brief Description	Response Codes
POST	/api/tools/{id}/favorite	likes	200, 404
DELETE	/api/tools/{id}/favorite	unlikes	200, 404

Status Codes

200 – successful request

201 – created successfully

204 – successfully deleted

400 – client sent invalid data

401 – authentication required

403 – client not allowed

404 – not found

500 – server error

Authentication

User must be authenticated to be able to change some data – use POST, PUT, DELETE requests.

To authenticate user must provide authentication token inside request header. “Authorization: Bearer <TOKEN>”

Errors

Errors should follow the following structure

```
{
  "error": {
    "code": 404,
    "message": "Tool not found",
    "details": "No tool with id=123"
  }
}
```

Pagination

All endpoints must support page and size. Response will include pagination data like below:

```
1 {
2   "content": [...],
3   "page": 1,
4   "size": 10,
5   "totalPages": 5,
6   "totalItems": 50
7 }
```

Caching

Caching should apply to:

- GET /api/tools
- GET /api/tools/{id}
- GET /api/categories

In order to optimize. When a new tool or category is changed or added cache should be cleared.