

Assumptions & Tests documentation

Parts Web API

Stanisław Matuszewski

September 27, 2023

This documentation provides a guide for understanding, configuring, and running a .NET 6 PartsWebAPI solution. The solution utilizes external NuGet packages:

- EntityFrameworkCore 7.0.11
- AutoMapper 12.0.1

Application connects to a SQL Server database using the code-first approach. The database schema includes two tables: *DestinationTypes* and *Parts*, which are connected by a one-to-many relationship. The Web API exposes five endpoints to perform basic CRUD operations on these database entities.

The program's functionalities are achieved through the utilization of two classes: *Part.cs* and *DestinationType.cs*.

```
public class Part
{
    public Guid Id { get; set; }

    [StringLength(7)]
    public string Name { get; set; }

    [StringLength(7)]
    public string Component { get; set; }

    [StringLength(2)]
    public DestinationType? Destination { get; set; }
}
```

```
public class DestinationType
{
    [Key, StringLength(2)]
    public string DestinationTypeId { get; set; }

    public int Order { get; set; }

    public ICollection<Part> Parts { get; set; }
}
```

Please note that some simplifications in this project code and documentation are a result of time constraints and the non-commercial nature of the project.

1 GET parts/list

This endpoint allows users to retrieve a list of all available parts stored in the database.

1.1 Success Result

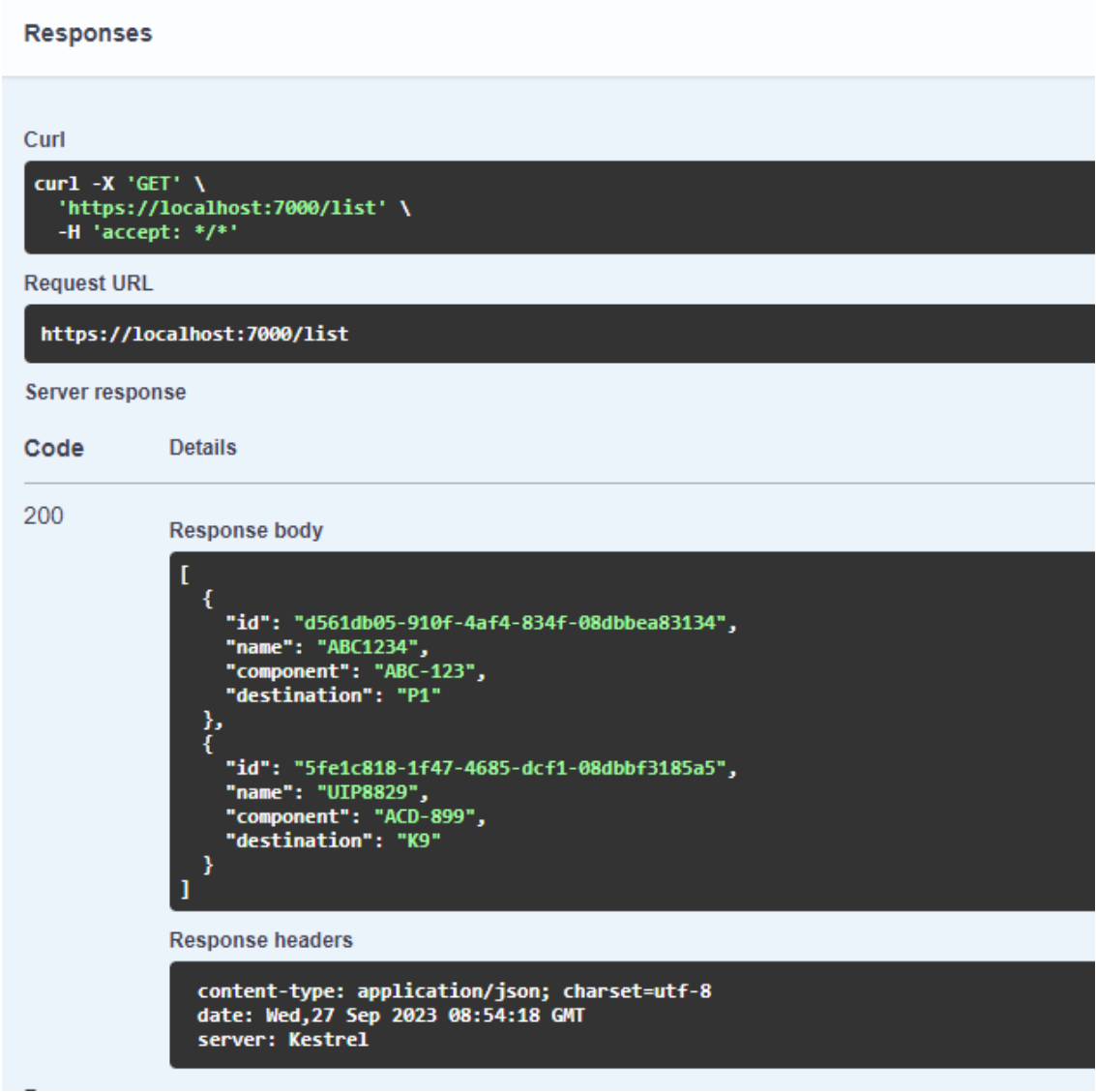


Figure 1: GET parts/list - result for successfully handled request

2 GET parts/destination

This endpoint allows users to retrieve a destination for a requested part.

2.1 Success Result

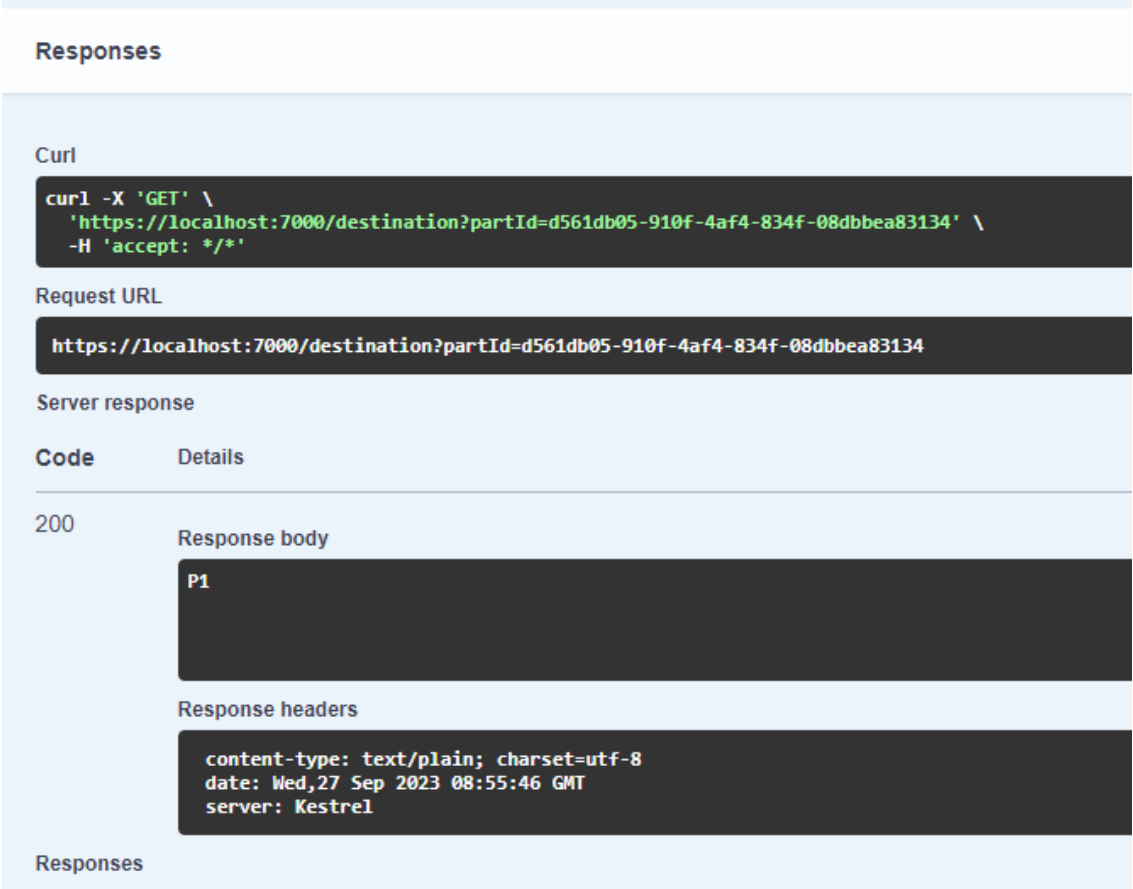


Figure 2: GET parts/add - result for successfully handled request

2.2 Error Result

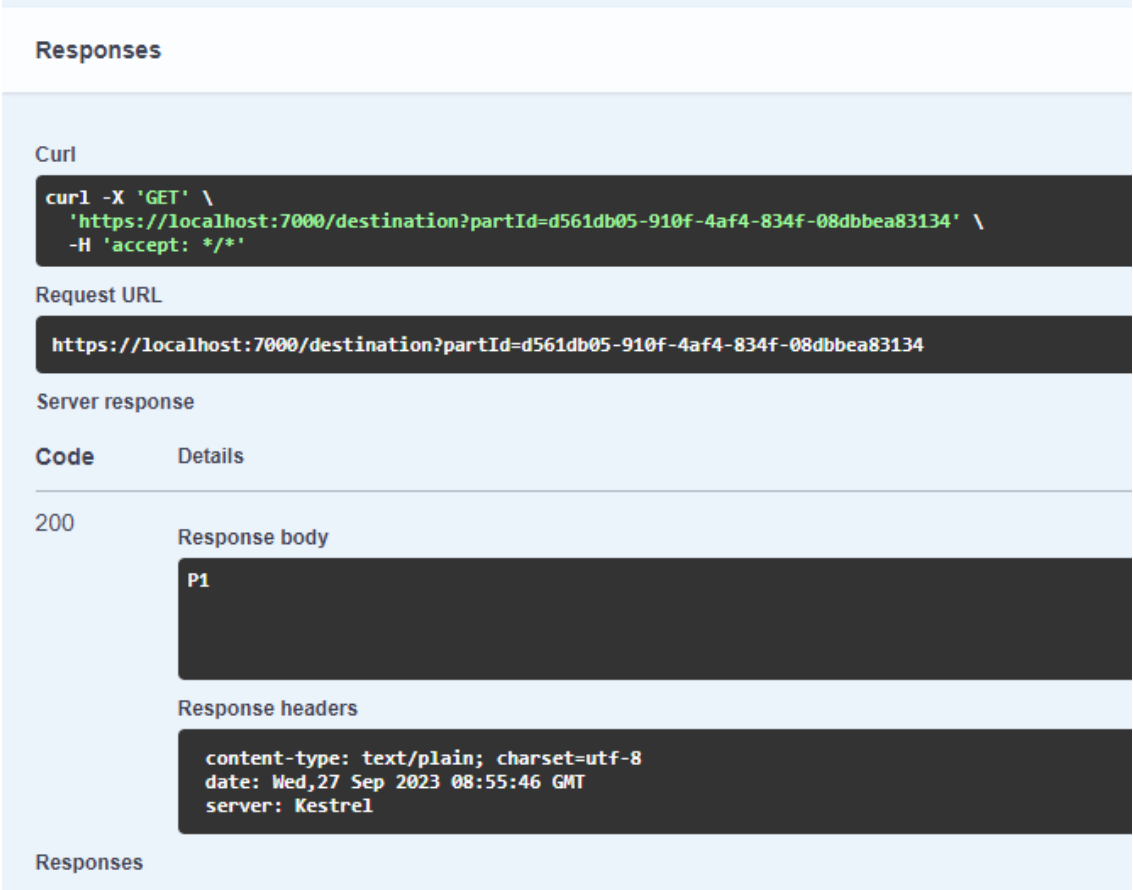


Figure 3: GET parts/add - result for successfully handled request

3 POST parts/add

This endpoint allows users to add a new Part record to the database. The provided record must fulfill validation criterion:

- Name property - string of exactly 7 characters: 3 letters followed by 4 numbers
- Component property - string of exactly 7 characters: 3 letters followed by '-', followed by 3 numbers
- Destination property - string of exactly 2 characters, its value must exists in table *DestinationType* with assigned to it Order value

Moreover, additional requirements are forced:

- The API should disallow to add the part that is already present in the system
- If there are already 20 parts in total, no more new parts can be added
- No more than 10 parts can be in "P1" state

In case any requirement is not met, the response result will include information indicating which specific condition or requirement is not satisfied.

3.1 Success Result

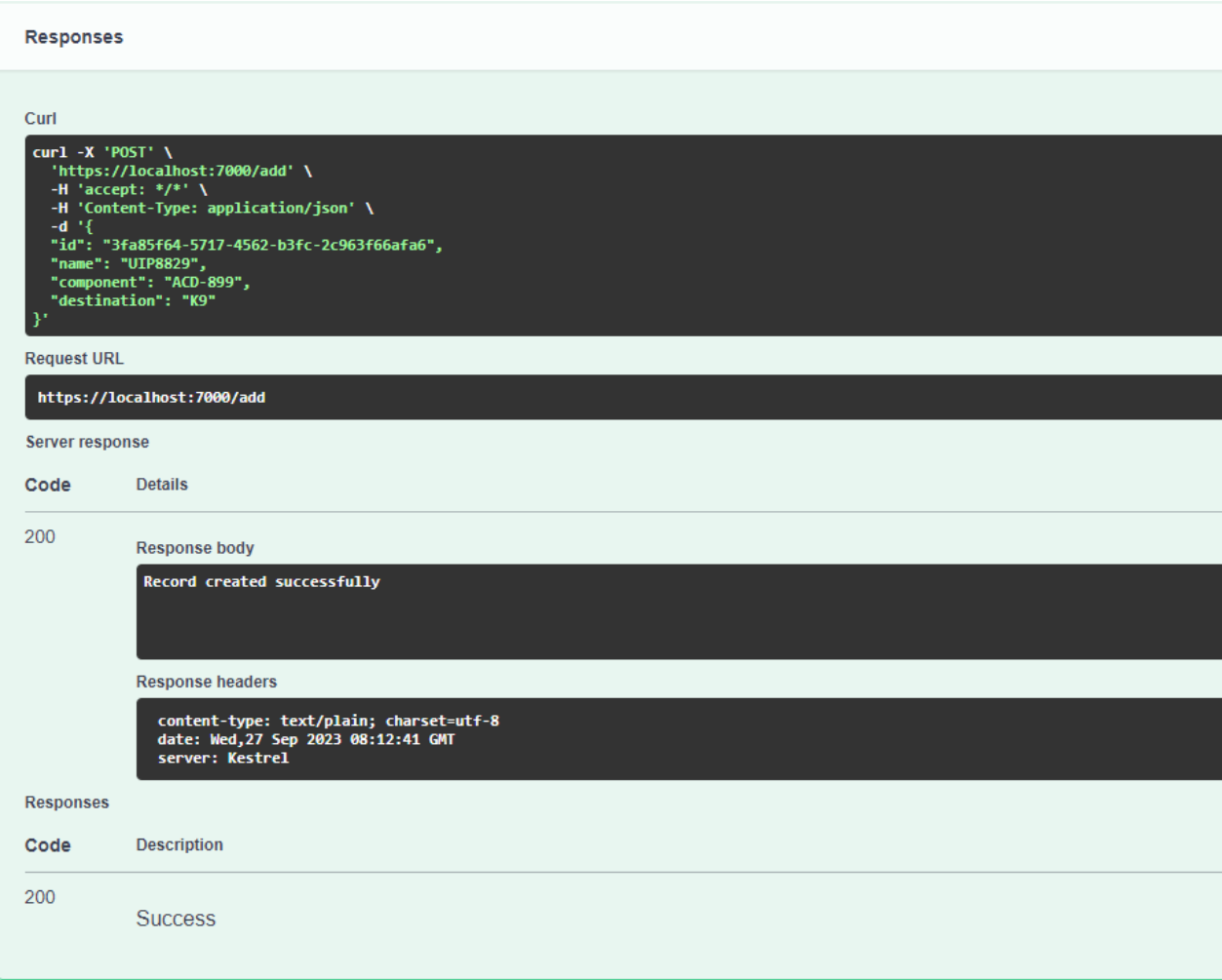


Figure 4: POST parts/add - result for correctly provided data

3.2 Error Result

Responses

Curl

```
curl -X 'POST' \
  'https://localhost:7000/add' \
  -H 'accept: */*' \
  -H 'Content-Type: application/json' \
  -d '{
    "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
    "name": "gQj3401",
    "component": "Yrh-397",
    "destination": "Y9"
  }'
```

Request URL

```
https://localhost:7000/add
```

Server response

Code	Details
400	Error: response status is 400

Undocumented

Response body

```
Provided Destination Code does not exist in database.
```

Response headers

```
content-type: text/plain; charset=utf-8
date: Wed, 27 Sep 2023 09:10:41 GMT
server: Kestrel
```

Responses

Figure 5: POST parts/add - provided destination does not exist in database

Responses

Curl

```
curl -X 'POST' \
  'https://localhost:7000/add' \
  -H 'accept: */*' \
  -H 'Content-Type: application/json' \
  -d '{
    "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
    "name": "eHGGu7050",
    "component": "ERJ-201",
    "destination": "B7"
  }'
```

Request URL

```
https://localhost:7000/add
```

Server response

Code

Details

400

Undocumented

Error: response status is 400

Response body

```
{
  "Name": [
    "The Name field must be in the format 'ABC1234'."
  ]
}
```

Response headers

```
content-type: application/json; charset=utf-8
date: Wed, 27 Sep 2023 10:28:58 GMT
server: Kestrel
```

Figure 6: POST parts/add - Part model validation not fulfilled

4 POST parts/update-destination

This endpoint allows users to update destination for a given part. Order of destinations is stored in DestinationTypes table. In case of parts that has already reached their maximum destination, operation results in ErrorMessage - *Operation failed. Given record has already its final destination.*

4.1 Success Result

Responses

Curl

```
curl -X 'POST' \
'https://localhost:7000/update-destination?partId=5FE1C818-1F47-4685-DCF1-08D8BF3185A5' \
-H 'accept: */*' \
-d ''
```

Request URL

https://localhost:7000/update-destination?partId=5FE1C818-1F47-4685-DCF1-08D8BF3185A5

Server response

Code	Details
200	<div><div>Response body</div><div>Part with Id 5fe1c818-1f47-4685-dcf1-08dbbf3185a5 has been updated.</div></div> <div><div>Response headers</div><div>content-type: text/plain; charset=utf-8 date: Wed,27 Sep 2023 10:53:05 GMT server: Kestrel</div></div>

Responses

Figure 7: POST parts/update-destination - result for successfully handled request

4.2 Error Result

Responses

Curl

```
curl -X 'POST' \
'https://localhost:7000/update-destination?partId=d561db05-910f-4af4-834f-08dbbea83134' \
-H 'accept: */*' \
-d ''
```

Request URL

https://localhost:7000/update-destination?partId=d561db05-910f-4af4-834f-08dbbea83134

Server response

Code	Details
400 <i>Undocumented</i>	<div>Error: response status is 400</div> <div><div>Response body</div><div>Operation failed. Given record has already its final destination.</div></div> <div><div>Response headers</div><div>content-type: text/plain; charset=utf-8 date: Wed,27 Sep 2023 08:56:52 GMT server: Kestrel</div></div>

Responses

Figure 8: POST parts/update-destination - result for part having its final destination

5 DELETE parts/remove

This endpoint allows users to remove concrete Part from database.

5.1 Success Result

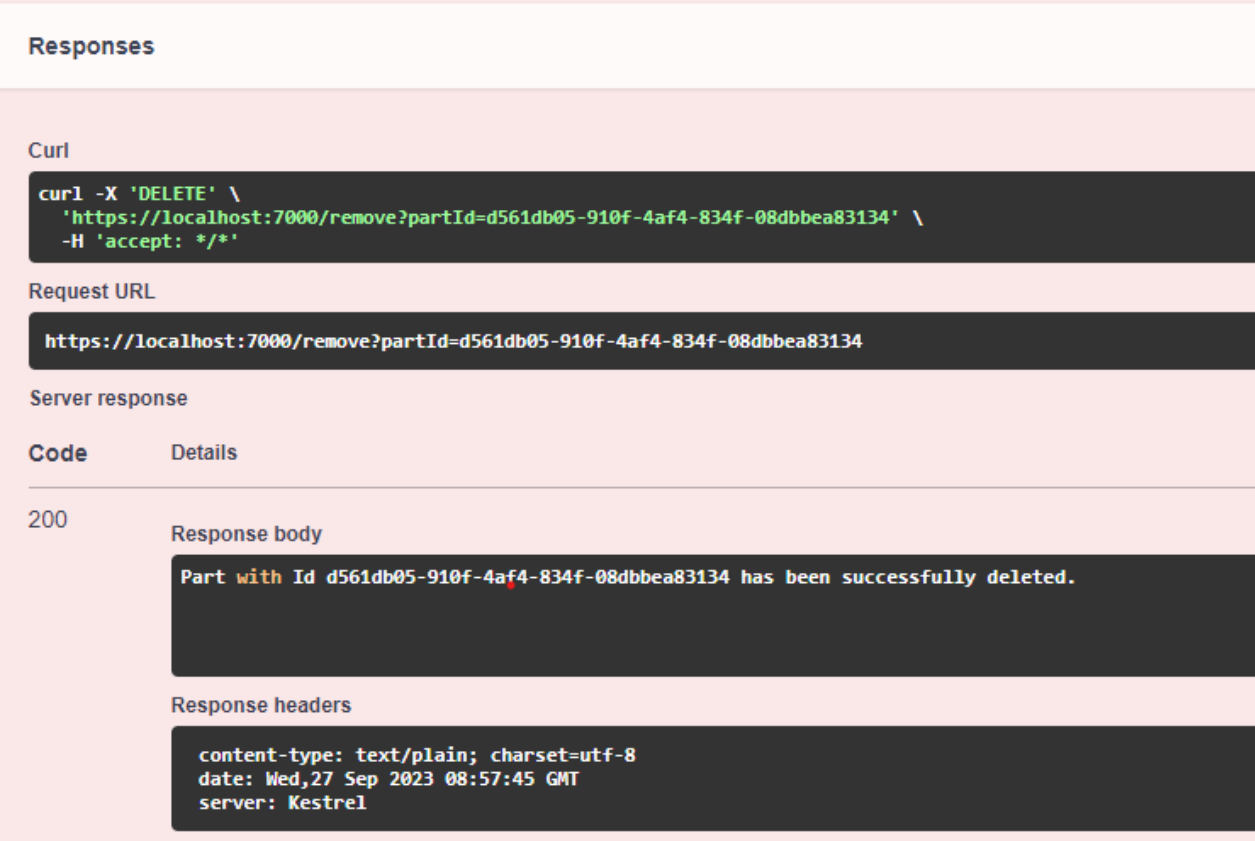


Figure 9: DELETE parts/remove - result for successfully handled request

5.2 Error Results

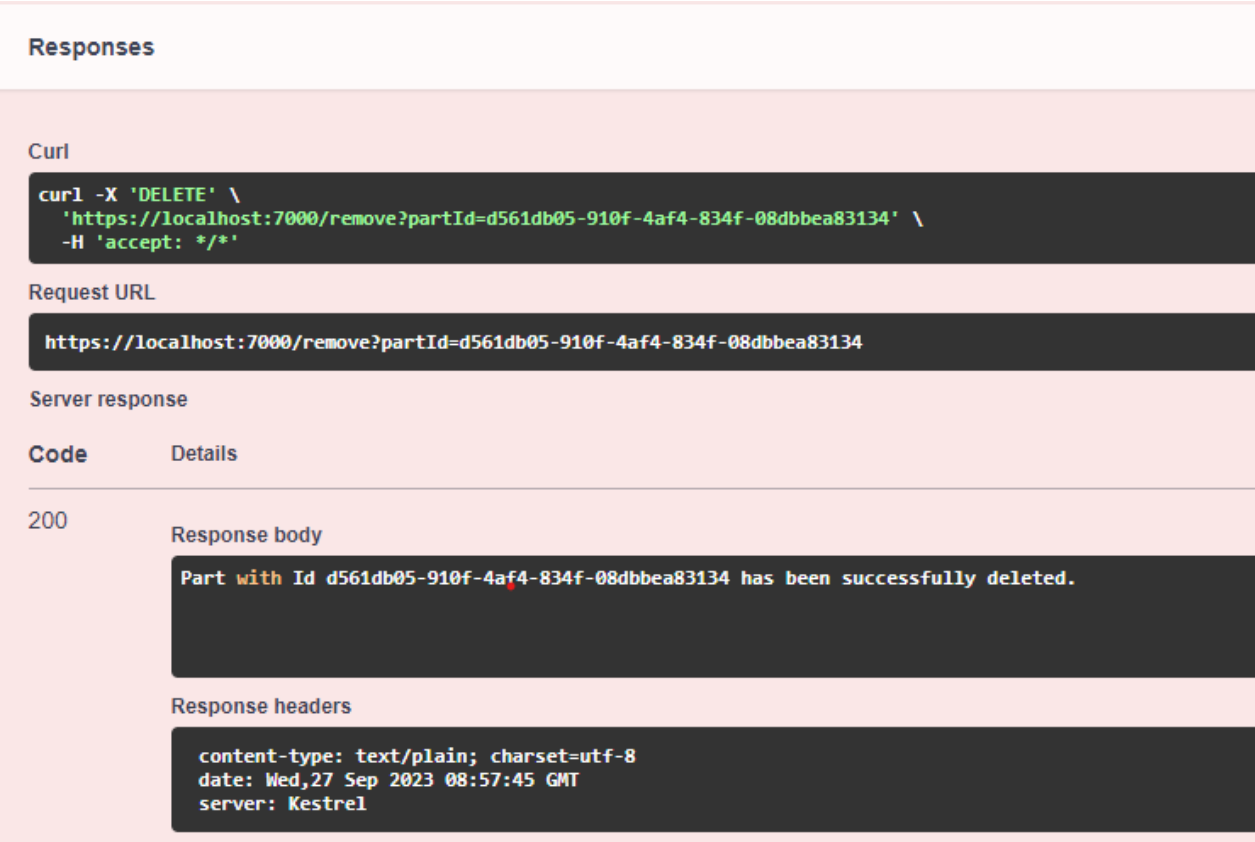


Figure 10: DELETE parts/remove - result for not found part in database