

BE Services	Request	Responses
Qr Scan	<pre> "/scan/{product_id}": { "get": { "tags": ["qrscan"], "summary": "Scan the meat and give more details", "parameters": [{ "name": "product_id", "in": "path", "description": "Id of the product to be scanned obtained by the scan", "required": true, "schema": { "type": integer } }], }, }</pre>	<pre> "responses": { "200": { "description": "Successful scan the product", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/Product" } } }, }, "400": { "description": "Invalid product id", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/Error" } } }, }, "401": { "description": "User isn't authenticated", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/ErrorUnauthorized" } } }, }, "404": { "description": "Product not found", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/NotFound" } } }, }, }</pre>
News	<pre> "/news": { "get": { "tags": ["news"], "summary": "Retrieves a list of the news", }, }</pre>	<pre> "responses": { "200": { "description": "List of news was retrieved successfully", "content": { "application/json": { "schema": { </pre>

	<pre> } }, </pre>	<pre> "\$ref": "#/components/schemas/News" } } }, "401": { "description": "User isn't authenticated", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/ErrorUnautho- riz-ed" } }, }, "404": { "description": "No news found", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/ErrorNot- Found" } }, }, }, </pre>
	<pre> "/news/{news_id}": { "get": { "tags": ["news"], "summary": "Retrieve a specific new", "parameters": [{ "name": "news_id", "in": "path", "description": "Id of the new to be retrieved", "required": true, "schema": { "type": integer } },], } } </pre>	<pre> "responses": { "200": { "description": "Specified new was retrieved successfully", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/New" } } }, }, "400": { "description": "Invalid news id", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/Error" } } }, }, "401": { "description": "User isn't authenticated", "content": { "application/json": { "schema": { </pre>

		<pre> "\$ref": "#/components/schemas/ErrorUnauthoriz ed" } }, "404": { "description": "No news found", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/ErrorNotFound" } } } } </pre>
Login	<pre> "/login": { "post": { "tags": ["authentication"], "summary": "Add new user", "parameters": [{ "name": "body", "in": "query", "description": "User object to be added", "required": true, "style": "form", "explode": true, "schema": { "\$ref": "#/components/schemas/User" } }], } } </pre>	<pre> "responses": { "200": { "description": "Login was successful, token returned", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/ AuthToken " } } } }, "400": { "description": "Invalid request data for login", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/Error" } } } }, "401": { "description": "Invalid credentials, user is not authenticated", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/ErrorUnauthoriz ed" } } } } } </pre>

		<pre>} } }, "403": { "description": "Access forbidden, account may be blocked or restricted", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/ErrorForbidden" } } } },</pre>
--	--	--

	<pre> "/signup": { "post": { "tags": ["authentication"], "summary": "Register a new user", "requestBody": { "description": "User object to be added", "required": true, "content": { "application/json": { "schema": { "\$ref": "/components/schemas/User" } } } }, }, } </pre>	<pre> "responses": { "200": { "description": "User was successfully registered", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/AuthToken" } } }, }, "400": { "description": "Invalid user data", "content": { "application/json": { "schema": { "\$ref": "/components/schemas/Error" } } }, }, "409": { "description": "User already exists", "content": { "application/json": { "schema": { "\$ref": "/components/schemas/ErrorConflict" } } }, }, } </pre>
--	---	--

productCategories	<pre> "/product-categories": { "get": { "tags": ["Product Categories"], "summary": "Retrieve product categories based on location", "parameters": [{ "name": "location", "in": "query", "description": "Location to filter product categories", "required": true, "schema": { "type": "string" } }], } } </pre>	<pre> "responses": { "200": { "description": "Product categories retrieved successfully", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/ProductCategori es" } } }, "400": { "description": "Invalid location parameter", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/Error" } } }, "401": { "description": "User isn't authenticated", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/ErrorUnauthoriz ed" } } }, "404": { "description": "No product categories found for the specified location", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/ErrorNotFound" } } } } } } } } </pre>
-------------------	---	---

		<pre> } }, } } } </pre>
	<pre> "/product-categories/{category_id}": { "get": { "tags": ["Product Categories"], "summary": "Retrieve a product category by ID", "parameters": [{ "name": "category_id", "in": "path", "description": "ID of the product category to retrieve", "required": true, "schema": { "type": "string" } }], } } </pre>	<pre> "responses": { "200": { "description": "Product category retrieved successfully", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/ProductCategory" } } }, }, "400": { "description": "Invalid category ID", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/Error" } } }, }, "401": { "description": "User isn't authenticated", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/ErrorUnauthorized" } } }, }, "404": { "description": "Product category not found", "content": { "application/json": { "schema": { </pre>

		<pre> "\$ref": "#/components/schemas/ErrorNotFound" } } }, } </pre>
Product	<pre> "/product- categories/product/{product_id}": { "get": { "tags": ["Product Categories"], "summary": "Retrieve product category by product ID", "parameters": [{ "name": "product_id", "in": "path", "description": "ID of the product to retrieve the category for", "required": true, "schema": { "type": "string" } }], } } </pre>	<pre> "responses": { "200": { "description": "Product retrieved successfully", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/Product" } } }, }, "400": { "description": "Invalid product ID", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/Error" } } }, }, "401": { "description": "User isn't authenticated", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/ErrorUnauthoriz ed" } } }, }, "404": { "description": "Product id not found", </pre>

		<pre> "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/ErrorNotFound" } } }, </pre>
User	<pre> "/user/{userId}": { "get": { "tags": ["User"], "summary": "Retrieve user by ID", "parameters": [{ "name": "userId", "in": "path", "description": "ID of the user to retrieve", "required": true, "schema": { "type": "string" } }], } } </pre>	<pre> "responses": { "200": { "description": "User retrieved successfully", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/User" } } }, }, "400": { "description": "Invalid user ID", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/Error" } } }, }, "401": { "description": "User isn't authenticated", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/ErrorUnauthoriz ed" } } }, }, "404": { "description": "User not found", "content": { </pre>

		<pre> "application/json": { "schema": { "\$ref": "#/components/schemas/ErrorNotFound" } }, </pre>
Favorites	<pre> "/favorites": { "get": { "tags": ["Favorites"], "summary": "Retrieve user's favorite items", "security": [{ "BearerAuth": [] }], } } </pre>	<pre> "responses": { "200": { "description": "Favorites retrieved successfully", "content": { "application/json": { "schema": { "type": "array", "items": { "\$ref": "#/components/schemas/Product" } } } }, "401": { "description": "User isn't authenticated", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/ErrorUnauthoriz ed" } } } }, }, } </pre>
	<pre> "/favorites/{product_id}": { "post": { "tags": ["Favorites"], "summary": "Add a product to user's favorites", "security": [{ "BearerAuth": [] }], } } </pre>	<pre> "responses": { "200": { "description": "Product added to favorites successfully", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/SuccessMessa ge" } } }, }, } </pre>

	<pre> }], "parameters": [{ "name": "product_id", "in": "path", "description": "ID of the product to add to favorites", "required": true, "schema": { "type": "string" } }],</pre>	<pre> } } } }, "400": { "description": "Invalid request data", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/Error" } } } }, "401": { "description": "User isn't authenticated", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/ErrorUnauthoriz ed" } } } }, "404": { "description": "Product not found", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/NotFound" } } } }, "409": { "description": "Product already in favorites", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/ErrorConflict" } } } }</pre>
--	---	---

		<pre> }, } </pre>
	<pre> "/favorites/{product_id}": { "delete": { "tags": ["Favorites"], "summary": "Remove a product from user's favorites", "security": [{ "BearerAuth": [] }], "parameters": [{ "name": "product_id", "in": "path", "description": "ID of the product to remove from favorites", "required": true, "schema": { "type": "string" } }], } } </pre>	<pre> "responses": { "200": { "description": "Product removed from favorites successfully", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/SuccessMessa ge" } } }, }, "400": { "description": "Invalid request data", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/Error" } } }, }, "401": { "description": "User isn't authenticated", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/ErrorUnauthoriz ed" } } }, }, "404": { "description": "Product not found in favorites", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/ErrorNotFound" } } }, }, } </pre>

		<pre> } } }, } </pre>
Feedback	<pre> "/product- categories/product/{product_id}/feedb ack": { "get": { "tags": ["Product Feedback"], "summary": "Retrieve feedback for a specific product", "parameters": [{ "name": "product_id", "in": "path", "description": "ID of the product to retrieve feedback for", "required": true, "schema": { "type": "string" } }], } } </pre>	<pre> "responses": { "200": { "description": "Feedback retrieved successfully", "content": { "application/json": { "schema": { "type": "array", "items": { "\$ref": "#/components/schemas/Feedback" } } } }, }, "400": { "description": "Invalid product ID", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/Error" } } }, }, "404": { "description": "Product not found", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/ErrorNotFound" } } }, }, } </pre>
	<pre> "/product- categories/product/{product_id}/feedb ack": { "post": { "tags": [</pre>	<pre> "responses": { "201": { "description": "Feedback added successfully", "content": { </pre>

	<pre> "Product Feedback"], "summary": "Add feedback for a specific product", "parameters": [{ "name": "product_id", "in": "path", "description": "ID of the product to add feedback for", "required": true, "schema": { "type": "string" } }], "requestBody": { "required": true, "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/Feedback" } } } }, </pre>	<pre> "application/json": { "schema": { "\$ref": "#/components/schemas/Feedback" } } }, "400": { "description": "Invalid request data", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/Error" } } } }, "404": { "description": "Product not found", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/ErrorNotFound" } } } }, } </pre>
Comuns a todas as resources		<pre> "500": { "description": "Internal Server Error", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/Error" } } } }, "503": { "description": "Service unavailable", "content": { "application/json": { "schema": { "\$ref": "#/components/schemas/Error" } } } } </pre>

		<pre> } } } "Error": { "type": "object", "properties": { "code": { "type": "integer", "description": "Error code" }, "message": { "type": "string", "description": "Error message" } } }, </pre>
--	--	---

A Secção seguinte é referente a todos os requests.

```

"components": {
  "schemas": {
    "New": {
      "type": "object",
      "required": [
        "titulo",
        "data",
        "descricao"
      ],
      "properties": {
        "titulo": {
          "type": "string",
          "description": "Title of the news article"
        },
        "data": {
          "type": "string",
          "description": "Date of the news article",
          "format": "date-time"
        },
        "descricao": {
          "type": "string",
          "description": "Description of the news article"
        }
      }
    }
  },
},

"News": {
  "type": "object",

```

```

    "properties": {
      "newsList": {
        "type": "array",
        "items": {
          "type": "object",
          "required": [
            "titulo",
            "data",
            "descricao"
          ],
          "properties": {
            "titulo": {
              "type": "string",
              "description": "Title of the news article"
            },
            "data": {
              "type": "string",
              "description": "Date of the news article",
              "format": "date-time"
            },
            "descricao": {
              "type": "string",
              "description": "Description of the news article"
            }
          }
        }
      }
    },
    "User": {
      "type": "object",
      "properties": {
        "username": {
          "type": "string",
          "description": "Username of the user"
        },
        "password": {
          "type": "string",
          "description": "Password of the user"
        },
        "email": {
          "type": "string",
          "description": "Email address of the user"
        }
      },
      "required": ["username", "password", "email"]
    },
    "AuthToken": {
      "type": "object",
      "properties": {
        "access_token": {
          "type": "string",
          "description": "JWT access token"
        }
      }
    }
  }

```



```
"categories": {
  "type": "array",
  "items": {
    "type": "object",
    "properties": {
      "id": {
        "type": "string",
        "description": "Category id"
      },
      "name": {
        "type": "string",
        "description": "Category name"
      },
      "description": {
        "type": "string",
        "description": "Category description"
      },
      "image": {
        "type": "binary",
        "description": "Category image"
      }
    }
  }
},
"ProductCategory": {
  "type": "object",
  "properties": {
    "id": {
      "type": "string",
      "description": "Category ID"
    },
    "name": {
      "type": "string",
      "description": "Category name"
    },
    "description": {
      "type": "string",
      "description": "Category description"
    },
    "image": {
      "type": "binary",
      "description": "Category image"
    }
  }
},
"SuccessMessage": {
  "type": "object",
  "properties": {
    "message": {
      "type": "string",
```

```
        "description": "Success message"
      }
    },
    "Feedback": {
      "type": "object",
      "properties": {
        "rating": {
          "type": "integer",
          "description": "Rating given by the user (1-5)"
        },
        "comment": {
          "type": "string",
          "description": "Comment provided by the user"
        },
        "userId": {
          "type": "string",
          "description": "ID of the user who provided the feedback"
        }
      }
    }
  }
}
```

```
"ErrorNotFound": {
  "type": "object",
  "properties": {
    "code": {
      "type": "integer",
      "description": "Error code"
    },
    "message": {
      "type": "string",
      "description": "Not found error message"
    }
  }
},
```

```
"Error": {
  "type": "object",
  "properties": {
    "code": {
      "type": "integer",
      "description": "Error code"
    },
    "message": {
      "type": "string",
      "description": "Error message"
    }
  }
},
```

```
"ErrorUnauthorized": {
  "type": "object",
```

```
"properties": {
  "code": {
    "type": "integer",
    "description": "Error code"
  },
  "message": {
    "type": "string",
    "description": "Unauthorized access message"
  }
},
"ErrorForbidden": {
  "type": "object",
  "properties": {
    "code": {
      "type": "integer",
      "description": "Error code"
    },
    "message": {
      "type": "string",
      "description": "Forbidden access message"
    }
  }
},
}
```