|  |  |  |  |
| --- | --- | --- | --- |
| **Description** | **Permission** | **Method + Path** | **Request/response** |
| Creates a new product. | Admin | **POST**  /admin  /product | Request: FormData{    product\_name: string,    category\_id: number,    product\_price: number,    product\_image: File,  }  Response: SuccessObject |
| Edit an already existing product | Admin | **PUT**  /admin  /product/:id | Request: FormData{    product\_name: string,    category\_id: number,    product\_price: number,  keepImage?:Boolean,    product\_image?: File,  }  Response: SuccessObject |

API doc

Endpoint: [**http://localhost:6500/api/**](http://localhost:6500/api/)

**SuccessObject : {** affectedRows: number,

  changedRows: number,

  fieldCount: number,

  insertId: number,

  message: string,

  protocol41: Boolean,

  serverStatus: number,

  warningCount: number,

**}**

|  |  |  |  |
| --- | --- | --- | --- |
| Creates a new order | User (token) | **POST**  /medium  /order | Request:{    city\_id: number,  street\_name: string,  arrival\_date: Date,  credit\_card\_digits:number,  cart\_id:number,  order\_price:number,  }  Response:SuccessObject |
| Deletes a cart product by its ID | User (token) | **DELETE**  /medium  /cart-product/  :id | Response: SuccessObject |
| Empties a certain cart by its ID | User (token) | **DELETE**  /medium  /cart-products  /cart-id/:cartId | Response: SuccessObject |
| Gets the dates that are fully occupied | User (token) | **GET**  /medium  /orders  /takeDates | Response: { takenDates : string }[] |
| Creates a new shopping cart | User (token) | **GET**  /medium  /new/shopIng-cart | Response: SuccessObject |

|  |  |  |  |
| --- | --- | --- | --- |
| Creates a new cart product | User (token) | **POST**  /medium  /cart-product | Request: {    product\_id: number,    amount: number,    total\_price: number,    cart\_id: number,  }  Response: SuccessObject |
| Returns the category list | User (token) | **GET**  /medium  /product-  categories | Response:{    category\_id: number,    category\_name: string,  }[] |
| Returns the products in a specific category | User (token) | **GET**  /medium  /products  /category-id  /:categoryId | Response: {    product\_id: number,    product\_name: string,    category\_id: number,    product\_price: number,    image\_name: string,  }[] |
| Returns the products that contain in the their name the parameter | User (token) | **GET**  /medium  /products  /product-name  /:productName | Response: {    product\_id: number,    product\_name: string,    category\_id: number,    product\_price: number,    image\_name: string,  }[] |
| Returns all of the user’s carts with their fitting orders | User (token) | **GET**  /medium  /cart-orders | Response: {    cart\_id: number,    order\_id: number | null,    creation\_date: string,    order\_date: string,    order\_price: number,  }[ |

|  |  |  |  |
| --- | --- | --- | --- |
| Returns the cart products of a certain cart by cart ID | User  (token) | **GET**  /medium  /cart-products  /:cartId | Response: {    cart\_product\_id: number,    product\_id: number,    amount: number,    total\_price: number,    cart\_id: number,  }[] |
| Returns an images by image name, if it doesn’t find returns a not found image | Guest | **GET**  /public  /images  /:imageName | Response  :File<image> |
| Returns the amount of orders that were made | Guest | **GET**  /public  /orders  /amount | Response:{    number\_of\_  orders:  number  } |
| Returns the amount of products that there are in the store | Guest | **GET**  /public  /products  /amount | Response:{    number\_of\_  products:  number  } |

|  |  |  |  |
| --- | --- | --- | --- |
| Returns the cities from the cities to select from | Guest | **GET**  /public/cities | Response: {    city\_id:  number,    city\_name:  string,  }[] |
| Validates the part 1 of the registration process | Guest | **POST**  /public  /register  /validate  /part1 | Request: {    id\_card: number,    user\_email: string,    password: string,    passwordConfirmation ?: string,  }  Response: {    valid:boolean  } |
| Creating a user for a guest. | Guest | **POST**  /public  /register | Request: {    first\_name: string,    last\_name: string,    city\_id: number,    street\_name:  string,  }  Response: SuccessObject |
| Logins a guest if his information matches the ones in the database. | Guest | **POST**  /public  /login | Request:{    user\_email:  string,    password:  string,  }  Response: { token: string  } |