|  |  |  |
| --- | --- | --- |
| Description | Method + Path | Request / Response |
| Validate first step of signup | POST http://localhost:4000/api/users/signup1 | Request: {  “userId”: 123, “email”: “test@gmail.com”,  “password”: “123”, “passwordAuthentication”: “123”  }  Response: OK |
| Create new User through signup | PUT http://localhost:4000/api/users/signup2 | Request: {  “userId”: 123, “firstName”: “Niv”, “lastName”: “Gr”,  “email”: “test@gmail.com”, “password”: “123”,  “passwordAuthentication”: “123”, “city”: “Tel Aviv”,  “street”: “abc 8”, “isAdmin”: 0  (automatically inserts 0, unless admin wants to signup  through POSTMAN, then he sends “1”)  }  Response: OK |
| User Login through email and password | POST http://localhost:4000/api/users/login | Request:{  “email”: “test@gmail.com”,  “password”: “123”  }  Response: {  "user": {  "isAdmin": false,  "\_id": "5d33188b4044da3ecc6e8d1d",  "userId": 2,  "firstName": "niv",  "lastName": "Gr",  "email": "test@gmail.com",  "password": "$2b$10$TROtl.35JqkCuQqfAUcezerlo3XO.bA31Grfzhkq54I/AQua366si",  "city": "Tel Aviv",  "street": "abc 8",  "\_\_v": 0  }  } |
| User Logout | GET http://localhost:4000/api/users/logout | Response: {  "status": "success"  } |
| Get All Categories in DB | GET http://localhost:4000/api/products/allCategories | Response: {  "categories": [  {  "products": [  "5d1f54325127ef290597120b",  "5d1f54435127ef290597120c",  "5d3afbe4d64cace9f8bdd4d3",  "5d3afc0de875411ae199f537",  "5d3afc25611b66f480e6f15b",  "5d3afc3d8c5ef4ec70d1cff3",  "5d3afc4b8c5ef4ec70d1cff4"  ],  "\_id":  "5d1f54325127ef290597120a",  "categoryName": "dairy",  "\_\_v": 0  },  {  "products": [  "5d2b18765d279b1ced695cd0"  ],  "\_id":  "5d2b18765d279b1ced695ccf",  "categoryName": "shit",  "\_\_v": 0  },  {  "products": [  "5d3d9e4c4e0e4402dc446bbb"  ],  "\_id":  "5d3d9e4c4e0e4402dc446bba",  "categoryName": "niv",  "\_\_v": 0  },  {  "products": [  "5d4d34383bddc36b2885f85c"  ],  "\_id":  "5d4d34383bddc36b2885f85b",  "categoryName": "yes",  "\_\_v": 0  }  ]  } |
| Get products depending on selected category | GET http://localhost:4000/api/products/productsByCategory/dairy | Response: {  "productsByCategory": [  {  "products": [  {  "category": [  "5d1f54325127ef290597120a"  ],  "\_id":  "5d1f54325127ef290597120b",  "productName": "yogurt",  "price": 25,  "imgURL":  "מרץ אפריל 2019.PNG",  "\_\_v": 0  },  {  "category": [  "5d1f54325127ef290597120a"  ],  "\_id":  "5d1f54435127ef290597120c",  "productName": "cheese",  "price": 20,  "imgURL":  "מרץ אפריל 2019.PNG",  "\_\_v": 0  }  ],  "\_id":  "5d1f54325127ef290597120a",  "categoryName": "dairy",  "\_\_v": 0  }  ]  } |
| Get all products from DB | GET  http://localhost:4000/api/products/allProducts | Response: {  "allProducts": [  {  "category": [  "5d1f54325127ef290597120a"  ],  "\_id":  "5d1f54325127ef290597120b",  "productName": "yogurt",  "price": 25,  "imgURL": "מרץ אפריל 2019.PNG",  "\_\_v": 0  },  {  "category": [  "5d1f54325127ef290597120a"  ],  "\_id":  "5d1f54435127ef290597120c",  "productName": "cheese",  "price": 20,  "imgURL": "מרץ אפריל 2019.PNG",  "\_\_v": 0  },  {  "category": [  "5d2b18765d279b1ced695ccf"  ],  "\_id":  "5d2b18765d279b1ced695cd0",  "productName": "fuck",  "price": 12409,  "imgURL": "asfkjaslf",  "\_\_v": 0  },  {  "category": [  "5d4d34383bddc36b2885f85b"  ],  "\_id":  "5d4d34383bddc36b2885f85c",  "productName": "niv",  "price": 420,  "imgURL": "random-pic-01.jpg",  "\_\_v": 0  }  ]  } |
| Check if user exists in DB, and get his info for further use (such as Admin indication, name, address). Works through Passport.js – only works after login (if session is not destroyed yet) | GET  http://localhost:4000/api/users/checkUserExists | Response: {  "myUser": {  "isAdmin": false,  "\_id": "5d33188b4044da3ecc6e8d1d",  "userId": 2,  "firstName": "niv",  "lastName": "gr",  "email": "niv2@gmail.com",  "password": "$2b$10$TROtl.35JqkCuQqfAUcezerlo3XO.bA31Grfzhkq54I/AQua366si",  "city": "safasf",  "street": "wqrqwrwq",  "\_\_v": 0  }  } |
| Get your specific cart (if open, meaning – didn’t go through order approved phase) | GET  http://localhost:4000/api/orders/getUserCart | Response: {  "userCart": {  "user": [  "5d33188b4044da3ecc6e8d1d"  ],  "cartproducts": [  {  "product": [  {  "category": [  "5d1f54325127ef290597120a"  ],  "\_id":  "5d1f54325127ef290597120b",  "productName":  "yogurt",  "price": 25,  "imgURL":  "מרץ אפריל 2019.PNG",  "\_\_v": 0  }  ],  "\_id":  "5d515cdfeab66ddeec2d880e",  "quantity": 1,  "totalPrice": 25,  "\_\_v": 0  }  ],  "\_id": "5d515cdfeab66ddeec2d880d",  "createDate":  "2019-08-12T12:34:39.160Z",  "\_\_v": 0,  "isOrdered": false  }  } |
| Get user’s last closed cart (went through completed order phase) – for info in the main welcome menu | GET  http://localhost:4000/api/orders/getUserClosedCart | Response: {  "userCart": {  "user": [  "5d33188b4044da3ecc6e8d1d"  ],  "cartproducts": [  {  "product": [  {  "category": [  "5d1f54325127ef290597120a"  ],  "\_id":  "5d1f54435127ef290597120c",  "productName":  "cheese",  "price": 20,  "imgURL":  "מרץ אפריל 2019.PNG",  "\_\_v": 0  }  ],  "\_id":  "5d51464fd831d4d2b41ce115",  "quantity": 1,  "totalPrice": 20,  "\_\_v": 0  }  ],  "\_id": "5d51464fd831d4d2b41ce114",  "createDate":  "2019-08-12T10:58:23.528Z",  "\_\_v": 0,  "isOrdered": true  }  } |
| Adding product to user’s cart | PUT  http://localhost:4000/api/orders/addProductToCart | Request: {  “productId”: "5d1f54325127ef290597120b",  “quantity”: 1  }  Response: OK |
| Admin functionality to edit existing product | PATCH  http://localhost:4000/api/products/updateProduct/productId | Request: {  “productData”: new FormData()  }  Response: OK |
| Admin functionality to add new product to DB | PUT  http://localhost:4000/api/products/addProduct | Request: {  “productData”: new FormData()  }  Response: OK |
| User go through order approval after placing order (cart closes down and order is created) | PUT  http://localhost:4000/api/orders/placeOrder | Request: {  “cart”: {userCart}, “city”: “Tel Aviv”,  “finalPrice”: 500, “street”: “abc 8”,  “shipDate”: “2020-01-01”,  “creditCard”: 123456789,  “cCardMethod”: “Visa”  }  Response: “Order Done” |
| User functionality to delete specific product from his cart | DELETE  http://localhost:4000/api/orders/deleteCartProduct/cartProductId | Response: OK |
| Get number of completed orders for the INFO component (showing how many orders were completed in our site) | GET  http://localhost:4000/api/orders/numberOfOrders | Response: 14 |
| Clear all products from user’s cart | DELETE  http://localhost:4000/api/orders/deleteCart/cartId | Response: OK |
| Get user’s (through Passport.js) last completed order | GET  http://localhost:4000/api/orders/getUserOrder | Response:  {  "user": [  "5d33188b4044da3ecc6e8d1d"  ],  "cart": [  "5d5045bdea54c2aa1c1ce9b8"  ],  "\_id": "5d5045cfea54c2aa1c1ce9ba",  "finalPrice": 25,  "city": "Be'er Sheva",  "street": "wqrqwrwq",  "shipDate": "3000-02-01",  "lastFourDigits": "3121",  "orderDate": "2019-08-11T16:43:59.648Z",  "\_\_v": 0  } |