**Super Mario Documentation**

Endpoint: <http://localhost:5000/>api

Main types display in the request/response column:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Category | Item | Order | Product | User |
| }  categoryId: number,  categoryName: string  { | {  productId:number,  quantity:number  totalPerProduct: number,  cartId: number,  itemId?: number  price:number,  productName:string  } | {  userId:number  cartId:number,  orderSum:number,,  shipCity:string,  shipStreet:string,  shipDate:Date,  orderDate:Date  creditCard:number,  payLastDigits:number,  orderId?:number  } | {  productId?:number,  productName:string  categoryId: number,  categoryName: string,  price: number  image:string,  imageFile: FileList  } | {  firstName:string,  lastName:string,  userId: number,  username: string,  password: string  city:string,  street:string  role?: number  } |

API

|  |  |  |
| --- | --- | --- |
| Request / Response | Method + Path | Description |
| Request: {  lastUserToken: string  }  Response: User | POST  users/public/autoLogin | Login with valid token/after register |
| Request: {  username: string,  password: string  }  Response: User+token | POST  users/public/login | Login without token |
| request.params.id  Response: User[ ] | GET  users/public/idValidation/:id | Check if user ID already exist.  (during registration) |
| request.params.username;  Response: User[ ] | GET  users/public/usernameValidation/:username | Check if username already exist .  (during Registration) |
| Request: User  Response: {  success/error message  } | POST  users/public/register | Register |
| Request: { }  Response: {  quantity: number  } | GET  products/public | Get amount of products |
| request.params.imageName;  Response: Image directory | GET  products/images/:imageName | Get Product image for cards. |
| request.params.categoryId  Response: Product[ ] | GET  products/productsByCategory/:categoryId | Get products by category |
| request.params.categoryId  Response: Product[ ] | GET  products/productsBySearch/:productName | Get products by search |
| Request: Product  Response: Product | POST  products/postNewProduct | Add new product |
| Request: Product  Response: success/error message | PUT  product/editProduct | Edit product |
| Request: { }  Response: {  quantity: number  } | GET  orders/public/ordersAmount | Get number of orders. |
| Request: Order  Response: Order | POST  orders/newOrder | Add new order |
| Request: { }  Response: Date[ ] | GET  orders/ busyDates | Get days with 3 or more orders. |
| Request: Item  Response: success/error message | POST  Items/postItem | Add new item |
| Request:{  params:{  cartId:number,  itemId:number  }  Response: success/error message | DELETE  items/deleteItems/:cartId/:itemId | Delete item |
| Request: { }  Response: categories[ ] | GET  categories/ | Get categories |
| Request{  newName:string  }  Response: success/error message | POST  categories/ | Add new category |
| request.params.userId  Response:{  resType: number,  data: string | object  } | GET  carts/orderStatus/:userId | Get information about open cart/last order. |
| Request:{  userId:number  }  Response:{  cartId:number  } | POST  carts/ | Open new cart |
| request.params.cartId  Response:item[ ] | GET  carts/cartContent/:cartId | Get cart content |