USERS:

Description	Method + Path	Request / Response
First step of registration	POST, /firstregister	**REQ: { userName, idNumber, password}** **RES: { messege: 'first Step Registration Successful', firstStepRegistrationInfo }**
Second step of registration	POST, /secondregister	**REQ: { name, lastName, city, street, idNumber, password, userName, role }** **RES: { messege: 'Registration Successful', token, newUser }**
User login	POST, /login	**REQ: { userName, pass: password }** **RES: { success: true, token, currentUser: currentUser }**
Get user Info	GET, /userinfo	**REQ: { id: userID }** **RES: userInfo**

Get user's last order by date (descending)	GET, /lastorder	**REQ: { id: userID }** **RES: userLastOrder**

STORE:

Description	**Method + Path**	**Request / Response**
Get homepage info	GET, /homepage	**RES: { ordersSummery, productsSummery }**
Get store info (products, categories)	GET, /mainstorepage	RES: { categories, products }

PRODUCT:

Description	**Method + Path**	**Request / Response**
Add new product to store (productList)	POST, /addproduct	**REQ: {** **name,** **category,** **price,** **image,** **amount** **}** **RES: 'New category created' || 'Product added successfully' || 'Product already exists'**
Update existing product	PUT, /updateproduct	**REQ: {** **name,**

		category, **price,** **image,** **amount,** **_id** **}** **RES: { msg: 'Product updated' }** }
Filter products by category	GET, /oncategory	**REQ: { id: userID }** **RES: { sortedProducts }**
Filter products by search pattern	GET, /search	**REQ: { id: userID }** **REQ.QUERY: searchString** **RES: { sortedProducts }**

ORDER:

Description	**Method + Path**	**Request / Response**
Create new order	POST, /placeorder	**REQ.USER: { id: userID }** **REQ>BODY: {** **city,** **street,** **shippingDate,** **creditCard,** **creationDate,** **ttlPrice** **}** **RES: { msg: "Your order has been sent! thank you for choosing Super Nir!" }**

Validate if requested date isnt full	POST, /datevalidator	**Response:** {
		theOrderObj
		}

Create order	Post/	**REQ: { shippingDate }** **RES: { isValid: false, msg: 'Delivery date is full, please choose A different date' }**

		|| **{ isValid: true, msg: 'Your order has been sent! thank you for choosing Super Nir!' }** success: true, result: newOrder }

CART:

Description	**Method + Path**	**Request / Response**
Get user's cart products	GET, /getcart	**REQ: { id: userID }** **RES: { userCart } || { msg: "User doeasnt have a cart" }**
Add product to cart	POST, /addproducttocart	**REQ.USER: { id: userID }** **REQ.BODY: { productId, amount }** **RES: { msg: "Cart created & product added" }** **||** **{ msg: "Cart product updated", updatedCart: updatedCart}** **||** **{ msg: "Cart updated", cart: cart }**
Delete user's cart if logout action triggered while user's cart is empty	DELETE, /deleteemptycart	**REQ: { id: userID }** **RES: { msg: "Cart Deleted" }**