

# "Online Super Market" API Doc

Hamutal Greenberg

<b>Endpoint:</b> <b>http://localhost:4200/register</b>		
Description	Method + Path	Request/ Response
Register user to the system	POST users/register	new Account({ username: String (email), ID: String, city: String, street: String, firstName: String, lastName: String, password: String (through passport) });  res.json

<b>Endpoint:</b> <b>http://localhost:4200/</b>		
Description	Method + Path	Request/ Response
Log in to the system	POST users/login	username: String (email), password: String  passport.authenticate res.json

<b>Endpoint:</b> <b>http://localhost:4200/</b>		
Description	Method + Path	Request/ Response
Check if session exists	GET users /checkSession	username: string (email) Passport  res.json

<b>Endpoint:</b> <b>http://localhost:4200/</b>		
Description	Method + Path	Request/ Response
Logout of system	GET users /logout	req.logout();  res.json

<b>Endpoint:</b> <b>http://localhost:4200/startShopping</b>		
Description	Method + Path	Request/ Response
Show Categories	GET store /categories	name:String  res.json

Users	Email	Password	Role
Manager	manager@gmail.com	123	Manager
Customer	customer@gmail.com	123	(none)

Validations	
Registration	Password and confirm password need to match
	Email verification ("@" and ".")
	All fields need to be filled in
Login	Email and password need to be correct (and match)
Order	All fields need to be filled in
	Cannot pick a date that already has 3 orders, cannot pick a date in the past, or more than a month from the order date
Users	Manager can add or edit a product in the system (customer cannot see this form)

<b>Endpoint:</b> <b>http://localhost:4200/startShopping</b>		
Description	Method + Path	Request/ Response
Show products by category	POST store /productsByCategory	Product { name:String, image:String, price:Number }  res.json

<b>Endpoint:</b> <b>http://localhost:4200/</b>		
Description	Method + Path	Request/ Response
Show how many products exist in database	GET store /ProductsNumber	res.json(products.length);

<b>Endpoint:</b> <b>http://localhost:4200/startShopping</b>		
Description	Method + Path	Request/ Response
Add a product to the system	POST store /addProduct	new Product { name:String, category:String, image:String, price:Number }  res.json

<b>Endpoint:</b> <b>http://localhost:4200/startShopping</b>		
Description	Method + Path	Request/ Response
Edit a product in the system	PUT store /editProduct	Product { _id: String, name:String, category:String, image:String, price:Number }  res.json

<b>Endpoint:</b> <b>http://localhost:4200/startShopping</b>		
Description	Method + Path	Request/ Response
Search for products in the system	GET store /search	Product  name:String,  by \$regex  res.json

---

<b>Endpoint:</b> <b>http://localhost:4200/startShopping</b>		
Description	Method + Path	Request/ Response
Show future shipping dates of orders	GET store /dates	futureDates[ ];  res.json

<b>Endpoint:</b> <b>http://localhost:4200/startShopping</b>		
Description	Method + Path	Request/ Response
Place an order	POST orders /newOrder	new Order { customerId: String, city: String, street: String, shippingDate: Date, updatedAt: Date, last4Digits: Number, total: Number, productsBought: Array }  res.json

<b>Endpoint:</b> <b>http://localhost:4200/</b>		
Description	Method + Path	Request/ Response
Show number of total orders	GET orders /OrdersNumber	res.json(orders.length);

<b>Endpoint:</b> <b>http://localhost:4200/</b>		
Description	Method + Path	Request/ Response
Show existing open cart or last order that was made	GET cart /userOrderDetails	Order Cart by customerId  res.json

<b>Endpoint:</b> <b>http://localhost:4200/startShopping</b>		
Description	Method + Path	Request/ Response
Add a new cart or check if customer has an open cart	POST cart /addCart	new Cart { customerID:String, products:Array, cartDate:String, totalPrice:Number, isClosed:Boolean }  Or check if existing cart by customerId  res.json

<b>Endpoint:</b> <b>http://localhost:4200/startShopping</b>		
Description	Method + Path	Request/ Response
Update an existing cart	PUT cart /updateCart	Cart by _id  res.json

<b>Endpoint:</b> <b>http://localhost:4200/startShopping</b>		
Description	Method + Path	Request/ Response
Show existing cart	GET cart /getCart	Cart by customerID  res.json