11/01/21

API DOC – Project "Hayarkan" Greengrocer By Shlomy Gerstner

Admin user-

Id:10101010

password:123

| **Description** | **Method + Path** | **Request** | **Response Success** |
| --- | --- | --- | --- |
| Check Token of user | GET  "/user" | headers :  { Authorization :  access token  } | Response : res.send(req.user) |
| Check User Id not taken for register | Post  'user/register\_first\_step' | Req.body : { userId } | Response : res.status(201).json({ error: false, msg: "user id does not exist in the system, you can move on to the next step" })  error {  res.status(201).json({ error: true, msg: "An existing user id in the system" }) |
| Register | Post 'user/register' | Req.body : { userId, fname, lname, email, password, city, address } | Response : res.status(201).json({ error: false, msg: "User Register successfully",access\_token }) |
| Login | Post "user/login" | Req.body : { userId , password} | Response : res.status(200).json({ error: false, msg: "User Login successfully",access\_token })  Errors of wrong password/missing info |
| Open new cart in the first shop | Get 'user/new\_cart/  :userId' | Req is the access token | Response : res.send(200).json(saveCart)  Errors for cart already or user id wrong |
| Add product to cart | Post 'user/add\_product\_to\_cart/  :cartId' | Req : { product\_id, quantity, price, sub\_total\_price } + User access token | Response : { res.status(201).json(result) }  Check if is need to add quantity or new product  Update the total price  Error: missing info |
| Remove product from cart | Post '/remove\_product\_from\_cart/:cartId' | Req : { product\_id } + user access token | Response res.status(201).json(result)  Error{ res.status(400).json({ error: true, msg: "The item not in the cart" }) |
| Clear all products in cart | Delete 'user/clear\_cart/  :cartId' | access token | Response : res.status(201).json(result)  error{  res.status(400).json({ error: true, msg: "Cart not found" }) |
| Get all cart items by user | Get '/getCartByUser/:userId' | Req : access token | Response : res.json(result) |
| Get all products | Get '/products' | Req : access token | Response : res.json(result)  Error {  res.status(500).json(err) |
| Get product by ID | Get '/products/:id' | Req : access token | Response : : res.json(result)  Error {  res.status(500).json(err) |
| Get products by category | Get '/products/  getByCategory/:id' | Req : access token | Response : res.json( products\_list) |
| Get products by search word | Get 'products/search\_product/:product' | Req : access token | Response : res.json(result) |
| Place order | Post 'orders/add' | Req body : { user\_id,cart\_id,total\_price,city,street,delivery\_date,credit\_card } + access token | Response res.status(201).json(orderDetails) |
| Get delivery available dates | Get  'orders/  check\_delivery\_date' | Req body : { customer\_id, cart\_id } + access token | Response : res.json({ data: [date, count(of deliveries)] } |
| Get all orders (to homepage) | Get  "/orders/ " |  | Response : all orders details |
| Add a product (Admin) | Post 'admin/add' | Req body : { product\_name, category\_id, price, image\_url } + access token Admin | Response : res.json({ error: false, msg: "product added successfully" }) |
| Edit a product | Post  'admin/edit/:productId' | Req body : { product\_name, category\_id, price, image\_url } + access token Admin | Response : res.json({ error: false, msg: "product edited successfully" }) |

**Languages**

* mongoDB
* nodeJS
* angular
* typescript
* javascript