

URL	TYPE	inputs	output	DESCRIPTION
buyerAuth/signin	POST	phone, password	Token + details	buyer signin
buyerAuth/signup	POST	email, password, confirmPassword, fullname, phone, shopName, shopAddress	Token + details	buyer signup
sellerAuth/signin	POST	phone, password	Token + details	seller signin
sellerAuth/signup	POST	{ email, password, confirmPassword, fullname, phone, shopName, shopAddress, description } = req.body;	Token + details	seller signup
deliveryAuth/signin	POST	Phone, fullname	Token + details	deliverer signin
deliveryAuth/signup	POST	phone, fullname	Token + details	deliverer signup
buyer				
buyer/profile	GET		buyerProf or Buyer Not Found	buyer profile
buyer/sellers	GET		sellers	get all sellers
buyer/products	GET		products	all products
buyer/seller/products/:id	GET	sellerId = req.params.id;	No Products of Seller Found Or products	search all seller products
buyer/search/:name	GET	productName =	products	search product by name

		req.params.name;	or No Products found	
buyer/products/idSearch/:id	GET	productId = req.params.id	products or Product not Found	search product by id
buyer/order	GET		count: docs.length, orders: docs  doc = all orders	get buyer orders
buyer/cart	GET		buyerCart or err	get cart products
buyer/cart/add	POST	{ productId, quantity } = req.body	buyerCart or newCart	modify cart (add, modify, quantity change)
buyer/cart/remove/:productId	DELETE	productId = req.params.productId;	buyerCart or Cart is already empty!	delete cart Items
buyer/checkout	POST	(Address in future right now it's get)	id: response.id, currency: response.currency, amount: response.amount,	Online payment + order adding (need to add address future scope)
buyer /cart/remove	DELETE		res.status(500).send("Something went wrong");	Deleting the cart for a user
buyer/payment/verify	POST	"OrderId": req.body.razorpay_order_id,  "paymentId": req.body.razorpay_payment_id, "Signature": req.body.razorpay_signature	response	Online payment verify and order create

buyer/COD	GET		response	Order created for cash payment
Buyer/order/:id	GET	req.params.id	ORDER	Get order by Id
seller				
seller/profile	GET		sellerProf or Seller Not Found	seller profile
seller/products/add	POST	name: req.body.name, price: req.body.price, quantity: req.body.quantity , description: req.body.description, img: req.body.img,	message: 'Product added', createProduct: { _id: result._id, name: result.name, description: result.description, price: result.price, quantity: result.quantity, img: result.img }	adding products
seller/products	GET		products	all products
seller/deliverers	GET		deliverers	all deliverers
seller/deliverers/add	POST	phone, fullname	result: result, token: token or Deliverer already exist. Or "Unauthorised User. Only a seller is Allowed to signup delivery person."	add new delivery person
seller/products/delete/:id	DELETE	id = req.params.id;  product to be deleted ID	product: doc  deleted	delete product

seller/products/update/:id	PATCH	id = req.params.id; product to be updated	message: 'Product updated'	update product data
seller/products/:id	GET	id = req.params.id; product ID to search	product: doc	search product by id
seller/products/:name	GET	name = req.params.name; product to search it's name	product: doc	search product by name
seller/orders	GET		Orders	Get orders
seller/assignOrder	Patch	{ orderId, delivererId } = req.body;	assigned	Assigning delivery person
seller/buyer/:id	GET	buyerId = req.params.id;	BUYER PROFILE	Get BuyerProfile
seller/order/:id	GET	req.params.id	ORDER	Get order by Id
delivery				
delivery/profile	GET		Profile	Delivery person profile
delivery/orders	GET		All orders assigned	See delivery person orders
delivery/buyer/:id	GET	buyerId = req.params.id;	BUYER PROFILE	Get BuyerProfile
Delivery/orderStatus	PATCH	{ orderId, status } = req.b ody;	message: 'Status Updated', order: result	Update order status
delivery/order/:id	GET	req.params.id	ORDER	Get order by Id
Forget password				
passwordReset	Patch	{ phone, password, confirmPassword } = req.body;	Password Updated	Buyer password reset
passwordReset/delivery	Patch	{ phone, password, confirmPassword } =	Password Updated	Delivery person password reset

		req.body;		
passwordReset/seller	Patch	{ phone, password, confirmPassword } = req.body;	Password Updated	Seller password reset
Reviews				
Need x-access-token in headers				
review/	GET		allReviews or no reviews found!	Get all the reviews
review/:productId	GET	productId = req.params.productId;	allReviewsOfProduct or No reviews Found!	Get reviews of a given product
review/myReview/:productId	GET	productId = req.params.productId;	reviewForProduct or "You don't have review for this product!"	Get your reviews of a product
review/delete/:productId	PATCH	productId = req.params.productId;	"Deleted Review!" Or "Review already deleted!"	Delete your review of a product
review/new	PATCH	{ productId, review, rating , img } = req.body;	newReview or Updated review	Post a new review or update an old review modify or create review