

Swagger url: [bupazar API](#)

## 1 Authentication

Functionality	URL	Method	Parameters	Responses
Sign up	/api/auth/register	POST	email*, username*, first_name*, last_name*, password*, address is_customer, is_vendor	success
Sign Up Activate	/api/auth/register_activate	POST	email*, number*	email, username, first_name, last_name, is_customer, is_vendor, is_active, is_staff, address, auth_token
Login	/api/auth/login	POST	email*, password*	id, email, username, first_name, last_name, is_customer, is_vendor, is_active, is_staff, address, auth_token
Logout	/api/auth/logout	POST	auth_token (inside headers)	success
Password Change	/api/auth/password_change	POST	auth_token (inside headers) current_password*, new_password*	success
User Information	/api/auth/user_info	POST	auth_token (inside headers)	id, email, username, first_name, last_name, is_customer, is_vendor, is_active, is_staff, address
User Profile Update	/api/auth/profile_update	POST	auth_token (inside headers) email, username, first_name, last_name, address	success
Google Login	/api/auth/google_login	POST	auth_token (that is taken from Google API)	id, email, username, first_name, last_name, is_customer, is_vendor, is_active, is_staff, address, auth_token
Facebook Login	/api/auth/facebook_login	POST	auth_token (that is taken from Facebook API)	id, email, username, first_name, last_name, is_customer, is_vendor, is_active, is_staff, address, auth_token

Table 1: Authentication Endpoints

## 2 Alarms

Functionality	URL	Method	Parameters	Responses
Deleting price alarm of a product for authenticated customers	api/alarms/delete-price-alarm/	POST	product_id*	success*
Setting price alarm with giving price for a product to notify requested customer when price of product goes down below giving price	api/alarms/set-price-alarm/	POST	product_id*, price*	success*
List all price alarms of requested customer	api/alarms/my-price-alarms/	GET	-	customer (username), product: Product, price*

Table 2: Alarms Endpoints

### 3 Users

Functionality	URL	Method	Parameters	Responses
List all users	/api/users	GET	-	id, email, username, first_name, last_name, is_vendor, is_customer, is_active, is_staff, address
Retrieve user details by user id	/api/users/{id}	GET	-	id, email, username, first_name, last_name, is_vendor, is_customer, is_active, is_staff, address
Retrieve vendor details by username	api/users/vendor/details	POST	vendor_username*	email*, username*, first_name*, last_name*, address*, rating*, products*: [Products]

Table 3: User Endpoints

### 4 Products

Functionality	URL	Method	Parameters	Responses
Retrieve product details by id	api/products/{id}/	GET	id* (path)	id, name, price, stock, description, date_added, number_of_sales, image_url, category, subcategory, vendor, total_rating_score, rating_count, brand, discount, rating
List all products	api/products/	GET	-	[Products]
List all products which are in same subcategory	api/products/subcategory/	POST	subcategory_name*	[Products]
List all products which are in same category	api/products/category/	POST	category_name*	[Products]
List all products of vendor which sends request	api/products/vendor-products/	GET	-	[Products]
List home page products	api/products/homepage/	POST	number_of_products*	'newest_arrivals': [Products], 'best_sellers': [Products], 'trends': [Products]
Add product for vendor	api/products/opts/add/	POST	name*, price*, stock*, description*, image_file*, subcategory_name*, brand*, discount*	success
Delete product for vendor	api/products/opts/delete/	POST	product_id*	success
Update product for vendor	api/products/opts/update_product/	POST	product_id*, name, price, stock, description, discount	success
Add comments and give rating	api/products/opts/add_comment/	POST	product_id*, comment_text*, is_anonymous*, rating_score*	success
Retrieve all comments of a given product	api/products/opts/get_all_comments/	POST	product_id*	[Comments]
Retrieve all comments of a customer	api/products/opts/get_user_comments/	POST		[Comments]
Search products	api/products/search/	POST	query*	[Products]
Sort products	api/products/sort/	POST	[product_ids*], sort_by*, order*	[Products]
Filter products	api/products/filter/	POST	[product_ids*], [filter_data*]	[Products]
Recommends best sellers for guests and similar products via checking search history, cart, favorites, lists and purchases for authenticated customers	api/products/recommend/	GET	-	[Products]

Table 4: Product Endpoints

## 5 Comments

Functionality	URL	Method	Parameters	Responses
List all comments	/api/comments	GET	-	id, customer, product* comment_text*, rating_score*, is_anonymous*
Retrieve comment details by comment id	/api/comments/{id}	GET	-	id, customer, product* comment_text*, rating_score*, is_anonymous*

Table 5: Comment Endpoints

## 6 Credit Cards

Functionality	URL	Method	Parameters	Responses
List all credit cards	/api/credit-cards	GET	-	id, name, customer*, card_owner* card_number*, expiration_date*, cvc_security_number*
Retrieve credit card details by comment id	/api/credit-cards/{id}	GET	-	id, name, customer*, card_owner* card_number*, expiration_date*, cvc_security_number*
Add new credit card	/api/credit-cards/opts/add	POST	name, card_owner* card_number* expiration_date*, cvc_security_number*	success
Delete a credit card	/api/credit-cards/opts/delete	POST	creditcard_id*	success
Retreive all credit cards of a user	/api/credit-cards/opts/get_all_credit_cards	POST	-	[Credit Cards]

Table 6: Credit Card Endpoints

## 7 Purchases

Functionality	URL	Method	Parameters	Responses
List all purchases	/api/purchases	GET	-	id, customer*, vendor*, product amount*, unit_price*, order* status*
Retrieve purchase details by purchase id	/api/purchases/{id}	GET	-	id, customer*, vendor*, product amount*, unit_price*, order* status*

Table 7: Purchase Endpoints

## 8 Orders

Functionality	URL	Method	Parameters	Responses
Make purchase	/api/orders/make_purchase	POST	-	success
Get a customer's orders	/api/orders/customer-orders	GET	-	order_id*, [Purchases]
Look whether a product is purchased by customer	/api/orders/customer-purchased	POST	product_id*	true/false
Vendor update status of an order	/api/orders/update-status	POST	order_id*, status*	success
Retrieve all purchases which contain requesting vendor' products	api/orders/vendor-orders/	GET	-	[Purchases]
Cancel purchase for vendor which sends request	api/orders/vendor-cancel/	POST	purchase_id	success
Cancel order for customer which sends request	api/orders/customer-cancel/	POST	order_id*	success
Add shipment for a purchase in an order	api/orders/add-shipment/	POST	purchase_id*, cargo_company*	cargo_no*
Get shipment information for a purchase in an order	api/orders/get-shipment/	POST	purchase_id*	id, purchase, date, cargo_no*, cargo_company*
Add a rating to a vendor	api/orders/add-vendor-rating/	POST	purchase_id*, rating_score*	success
Calculate average vendor rating in a product page	api/orders/avg-rating-product-page/	POST	product_id*	score*
Calculate average vendor rating in a vendor profile page	api/orders/avg-rating-profile-page/	POST	-	score*

Table 8: Order Endpoints

## 9 Chats

Functionality	URL	Method	Parameters	Responses
Create Chat	/api/chats/create_chat	POST	vendor_username*, product_id*	success, [Chat]
Get all chats	/api/chats/get_all_chats	GET	-	success, [Chat, [Message]]
Get chat history	/api/chats/get_chat_history/	POST	chat_id*	success, [Message]
Get last message	/api/chats/get_last_message/	POST	chat_id*	success, Message
Send message	/api/chats/send_message/	POST	chat_id*, content*	success, [Message]
Get the number of unread messages	/api/chats/get_unread_messages_number/	GET	-	number
Delete message	/api/chats/delete_message/	DELETE	message_id	success
Get the number of unread messages	/api/chats/delete_chat/	DELETE	chat_id, message_id	success

Table 9: Chat Endpoints

## 10 Cart

Functionality	URL	Method	Parameters	Responses
Get Cart	/api/cart/get	GET	None	Cart
Edit cart	/api/cart/edit	POST	product_id, count	CartResponse
Clear Cart	/api/cart/clear	DELETE	None	CartResponse

Table 10: Cart Endpoints

## 11 Favorites

Functionality	URL	Method	Parameters	Responses
Get Favorites	/api/favorites/get	GET	None	FavoriteList
Add product to favorites	/api/favorites/add	POST	product_id	FavoritesResponse
Remove product from favorites	/api/favorites/add	POST	product_id	FavoritesResponse

Table 11: Favorites Endpoints

## 12 Product Lists

Functionality	URL	Method	Parameters	Responses
Get my product lists	/api/product-lists/opts/my	GET	None	[ProductList]
Create A product list	/api/product-lists/opts/add	POST	name	ProductListResponse
Delete one of my lists	/api/product-lists/opts/delete	POST	list_id	ProductListResponse
Add a product to one of my lists	/api/product-lists/opts/add_product	POST	list_id , product_id	ProductListResponse
Remove a product from one of my lists	/api/product-lists/opts/remove_product	POST	list_id , product_id	ProductListResponse

Table 12: Product Lists Endpoints

## 13 Notifications

Functionality	URL	Method	Parameters	Responses
List all notifications of requested user	api/notifications/my/	GET	-	[id, text*, notificationType, user, createdAt, product, order*, isSeen]
Setting notification as seen	api/notifications/set_seen/	POST	notification_id*	success*
List all notifications in database	api/notifications/all/	GET	-	[Notifications]

Table 13: Notification Endpoints

## 14 Admins

Functionality	URL	Method	Parameters	Responses
Is Admin	/api/admin/is_admin	GET	None	true/false
Assign Admin	/api/admin/assign_admin	POST	username	success
Delete Comment	/api/admin/delete_comment	POST	comment_id	success
Delete User By Comment Id	/api/admin/delete_user_by_comment_id	POST	comment_id	success
Delete User	/api/admin/delete_user_by_username	POST	username	success

Table 14: Admin Endpoints