

API Documentation

Generated by GenDoc on 2026-02-15 00:44

Executive Summary

Endpoint	Method	Request	Response
/api/	POST	RegisterUserSerializer	400: RegisterUserSerializer
/api/	POST	CreateOrderSerializer	102: Object {message}, 200: Object {message}, 400: Object...
/api/me/	GET	NoBody	200: DisplayUserSerializer
/api/me/	DELETE	NoBody	204: NoContent
/api/me/	PUT	UpdateUserSerializer	200: UpdateUserSerializer, 400: UpdateUserSerializer
/api/me/	PATCH	PasswordSerializers	200: PasswordSerializers, 204: NoContent, 401: Object {r}
/api/me/address/	GET	NoBody	200: AddressSerializers
/api/me/address/	POST	AddressSerializers	201: Object {message}, 400: AddressSerializers
/api/me/address//	PATCH	AddressSerializers	200: Object {message}, 400: AddressSerializers
/api/me/address//	DELETE	NoBody	204: NoContent
/api/me/orders/	GET	NoBody	200: OrderSerializer

Endpoint	Method	Request	Response
/api/me/orders//	GET	NoBody	200: OrderSerializer, 400: Object {message}
/api/me/orders//	PATCH	UpdateOrderSerializer	200: Object {message}, 400: UpdateOrderSerializer
/api/me/shoppingcart/	GET	NoBody	200: ShoppingCartSerializer
/api/me/shoppingcart//	PATCH	NoBody	200: Unknown
/api/me/shoppingcart//	PUT	NoBody	200: Object {r}
/api/me/wishlist/	GET	NoBody	200: WishlistSerializer
/api/me/wishlist//	PATCH	NoBody	204: NoContent
/api/me/wishlist//	PUT	NoBody	200: Object {r}
/api/pay/	POST	RazorPayStartPaymentSe rializer	200: RazorPayStartPaymentSeri alizer
/api/payment/success/	POST	RazorPayFinishPayments erializer	200: RazorPayFinishPaymentSer ializer
/api/product/	GET	NoBody	200: ProductSerializer
/api/product/	POST	CreateProductSerialize r	200: CreateProductSerializer, 400: CreateProductSerializer
/api/product//	GET	NoBody	200: ProductSerializer
/api/product//	PUT	RawBody	200: Object {r}
/api/product//	POST	RawBody	201: Unknown
/api/product//	DELETE	NoBody	204: NoContent

Endpoint Details

`/api/`

View Class: `RegisterUserProfileView`

Class representing an API view for registering a user profile. Methods: - `get_permissions`: Get the permissions required for different HTTP methods. - `post`: Register a new user profile.

Methods

POST

Input Parameters: Schema: `RegisterUserSerializer`

Responses:

400 (`RegisterUserSerializer`)

`/api/`

View Class: `CreateOrderApiView`

No description.

Methods

POST

Input Parameters: Schema: `CreateOrderSerializer`

Responses:

102 (`Object {message}`)

200 (`Object {message}`)

400 (`Object {message}`)

`/api/me/`

View Class: `GetUserProfileView`

Class representing an API view for retrieving, updating, and deleting a user profile. Methods: - `get_user_instance`: Get the user instance of the authenticated user. - `get`: Retrieve the user profile. - `delete`: Delete the user profile. - `put`: Update the user profile. - `patch`: Change the user password.

Methods

GET

Input Parameters: *No input required.*

Responses:

200 (`DisplayUserSerializer`)

Field	Type	Properties	Nested Schema
address	<code>List[AddressSerializers]</code>	<code>ReadOnly</code>	-

DELETE

Input Parameters: *No input required.*

Responses:

204 (`NoContent`)

PUT

Input Parameters: Schema: `UpdateUserSerializer`

Responses:

200 (`UpdateUserSerializer`)

400 (`UpdateUserSerializer`)

PATCH

Input Parameters: Schema: `PasswordSerializers`

Field	Type	Properties	Nested Schema
old_password	CharField	-	-
new_password	CharField	-	-

Responses:

200 (PasswordSerializers)

Field	Type	Properties	Nested Schema
old_password	CharField	-	-
new_password	CharField	-	-

204 (NoContent)

401 (Object {r})

/api/me/address/

View Class: AddressAPIView

Class representing an API view for creating and retrieving user addresses. Methods: - *get_user_instance*: Get the user instance of the authenticated user. - *get*: Retrieve the addresses associated with the authenticated user. - *post*: Create a new address for the authenticated user.

Methods

GET

Input Parameters: No input required.

Responses:

200 (AddressSerializers)

POST

Input Parameters: Schema: AddressSerializers

Responses:

201 (Object {message})

400 (AddressSerializers)

/api/me/address/<address_id>/

View Class: AddressUpdateView

Class representing an API view for updating and deleting user addresses. Methods: - delete: Delete a specific address. - put: Update a specific address.

Methods

PATCH

Input Parameters: Schema: AddressSerializers

Responses:

200 (Object {message})

400 (AddressSerializers)

DELETE

Input Parameters: No input required.

Responses:

204 (NoContent)

/api/me/orders/

View Class: OrderAPIView

Class representing an API view for orders. Methods: - get: Retrieve all orders. - post: Create a new order. - patch: Update an existing order.

Methods

GET

Input Parameters: No input required.

Responses:

200 (OrderSerializer)

Field	Type	Properties	Nested Schema
order_id	UUIDField	-	-
order_payment_id	CharField	Nullable	-
order_price	DecimalField	-	-
status	CharField	-	-
MOP	CharField	-	-
date_of_order	DateTimeField	-	-
date_of_payment	DateTimeField	Nullable	-

/api/me/orders/<order_id>/

View Class: `GetUpdateOrderAPIView`

No description.

Methods

GET

Input Parameters: No input required.

Responses:

200 (`OrderSerializer`)

Field	Type	Properties	Nested Schema
order_id	UUIDField	-	-
order_payment_id	CharField	Nullable	-
order_price	DecimalField	-	-
status	CharField	-	-
MOP	CharField	-	-
date_of_order	DateTimeField	-	-

Field	Type	Properties	Nested Schema
date_of_payment	DateTimeField	Nullable	-

400 (Object {message})

PATCH

Input Parameters: Schema: UpdateOrderSerializer

Field	Type	Properties	Nested Schema
status	ChoiceField	-	-

Responses:

200 (Object {message})

400 (UpdateOrderSerializer)

Field	Type	Properties	Nested Schema
status	ChoiceField	-	-

/api/me/shoppingcart/

View Class: ShoppingcartAPIView

Class representing an API view for managing a user's shopping cart. Methods: - get_queryset: Get the shopping cart queryset for a specific user. - get: Retrieve the shopping cart items for the authenticated user. - patch: Remove a specific product from the shopping cart. - put: Add a specific product to the shopping cart.

Methods

GET

Input Parameters: No input required.

Responses:

200 (ShoppingcartSerializer)

Field	Type	Properties	Nested Schema
shopping_item	List[W_S_ProductSerializer]	-	-

`/api/me/shoppingcart/<product_id>/`

View Class: ShoppingCartUpdateView

No description.

Methods

PATCH

Input Parameters: *No input required.*

Responses:

200 (Unknown)

PUT

Input Parameters: *No input required.*

Responses:

200 (Object {r})

`/api/me/wishlist/`

View Class: WishlistAPIView

Class representing an API view for managing a user's wishlist. Methods: - get_queryset: Get the wishlist queryset for a specific user. - get: Retrieve the wishlist items for the authenticated user. - patch: Remove a specific product from the wishlist. - put: Add a specific product to the wishlist.

Methods

GET

Input Parameters: *No input required.*

Responses:

200 (WishlistSerializer)

Field	Type	Properties	Nested Schema
wishlist_item	List[W_S_ProductSerializer]	-	-

/api/me/wishlist/<product_id>/

View Class: WishlistUpdateView

No description.

Methods

PATCH

Input Parameters: No input required.

Responses:

204 (NoContent)

PUT

Input Parameters: No input required.

Responses:

200 (Object {r})

/api/pay/

View Class: RazorpayStartPayment

Class representing an API view for starting a payment with Razorpay. Methods: - post: Start a payment with Razorpay.

Methods

POST

Input Parameters: Schema: RazorPayStartPaymentSerializer

Field	Type	Properties	Nested Schema
order_item	UUIDField	-	-

Field	Type	Properties	Nested Schema
order_price	DecimalField	-	-

Responses:

200 (RazorPayStartPaymentSerializer)

Field	Type	Properties	Nested Schema
order_item	UUIDField	-	-
order_price	DecimalField	-	-

/api/payment/success/

View Class: RazorpayPaymentSuccess

Class representing an API view for handling successful payments with Razorpay. Methods: -
post: Handle a successful payment with Razorpay.

Methods

POST

Input Parameters: Schema: RazorPayFinishPaymentSerializer

Field	Type	Properties	Nested Schema
razorpay_order_id	CharField	-	-
razorpay_payment_id	CharField	-	-
razorpay_signature	CharField	-	-

Responses:

200 (RazorPayFinishPaymentSerializer)

Field	Type	Properties	Nested Schema
razorpay_order_id	CharField	-	-
razorpay_payment_id	CharField	-	-

Field	Type	Properties	Nested Schema
razorpay_signature	CharField	-	-

/api/product/

View Class: `List_products`

Class representing an API view for listing products. Methods: - get_permissions: Get the permissions required for different HTTP methods. - get: Retrieve all products. - post: Create a new product.

Methods

GET

Input Parameters: *No input required.*

Responses:

200 (`ProductSerializer`)

POST

Input Parameters: Schema: `CreateProductSerializer`

Responses:

200 (`CreateProductSerializer`)

400 (`CreateProductSerializer`)

/api/product/<id>/

View Class: `Get_product`

Class representing an API view for retrieving, updating, creating, and deleting a product. Methods: - get_permissions: Get the permissions required for different HTTP methods. - get: Retrieve a specific product. - put: Update a specific product. - post: Create an image for a specific product. - delete: Delete a specific product.

Methods

GET

Input Parameters: *No input required.*

Responses:

200 (`ProductSerializer`)

PUT

Input Parameters: Schema: `RawBody`

Responses:

200 (`Object {r}`)

POST

Input Parameters: Schema: `RawBody`

Responses:

201 (`Unknown`)

DELETE

Input Parameters: *No input required.*

Responses:

204 (`NoContent`)
