

GLAMORE APPLICATION

Architecture and Documentation

version 1.0.0

by Lac-doh-Wilson Tchofor

BASE_URL:DEV <http://localhost:900>

BASE_URL:PROD <https://www.backend.glamorefashions.com>

SN	Module	Feature	Endpoints	Fields	Description	Method
1	Authentication	Sign up	api/auth/signup	firstname: required lastname: required email: required phone: required password: required passwordConfirm: required	Register or sign up a new user	POST
		Login	api/auth/login	email: required password: required	Login an already existing user	POST
		Forgot password	api/auth/forgot-password			POST
		Reset password	api/auth/reset-password		Reset user password	POST
2	User	Search	api/users/search		Search for an existing user by firstname, email or lastname	GET
		Count	api/users/count		Count number of users	GET
		Register a delivery agent	api/users/register		Register a new delivery agent	POST
		Profile	api/users/me		Get user Profile	GET
		Update me	api/users/update-me		Update user profile	PATCH
		Delete me	api/users/delete-me		Delete my account	DELETE
		My Shops	api/users/my-shops		Get the shops user follows	GET
		My Favorites	api/users/my-favorites		Get user favorite products	GET
3	Category	My cart	api/users/my-cart		Get user carts	GET
		Get category Products	api/categories/:categoryId/products		Get al Products of a particular	GET

	Search	api/categories/search	category Search category by name	GET
	Count	api/categories/count	Count all categories	GET
	Create	api/categories	Create a new category	POST
	Get Categories	api/categories	Get all categories	GET
	Get Category	api/categories/:id	Get a category by id	GET
	Update Category	api/categories/:id	Update a category by it id	PATCH
	Delete Category	api/categories/:id	Delete a category by it id	DELETE
	4 Products	Get product reviews	Get all product reviews	GET
		Search	Search a product by it name	GET
	Count	api/products/count	Count all products	GET
	Create	api/products	Create a new product	POST
	Get All	api/products	Get all products	GET
	Get one	api/products/:id	Get/Read a single product by it id.	GET
	Update One	api/products/:id	Update a single product by it id	PATCH
	Delete One	api/products/:id	Delete a product by it id	DELETE
	5 Store	Store Products	Get all products belonging to store	GET
		Store followers	Get all users following store	GET
	Search	api/stores/search	Search a store by name	GET
	Count	api/stores/count	Count number of stores	GET
	My Store	api/stores/my-store	Get current user store	GET
	Create	api/stores	Create a new Store	POST
	Get All	api/stores	Get all Stores	GET
	Get one	api/stored/:id	Get a store by it id	GET
	Update one	api/stored/:id	Update a store	PATCH

6	Carts	Delete one	<i>api/stored/:id</i>	by it id Delete a store	DELETE
		Count	<i>api/carts/count</i>	by it id Count products in cart	GET
		Create	<i>api/carts</i>	Create a new cart	POST
		Get all	<i>api/carts</i>	Get all products from cart	GET
		Get one	<i>api/carts/:id</i>	Get a single product from cart	GET
		Update one	<i>api/carts/:id</i>	Update an item from carts	UPDATE
7	Review	Delete one	<i>api/carts/:id</i>	Delete an item from carts	DELETE
		Count	<i>api/reviews/count</i>	Count number of reviews	GET
		Create	<i>api/reviews</i>	Create a new review	POST
		Get all	<i>api/reviews/</i>	Get all reviews	GET
		Get one	<i>api/reviews/:id</i>	Get a single review	GET
		Update one	<i>api/reviews/:id</i>	Update a review	PATCH
8	Follow	Delete one	<i>api/reviews/:id</i>	Delete a review by id	DELETE
		Toggle follow	<i>api/follows/toggle-follow</i>	Add or remove a store from user follows. If the store exist, it is removed else added.	PATCH
		Is Following	<i>api/follows/is-following</i>	Check if user if following a store	GET
9	Favorites	Toggle Favorite	<i>api/favorites/toggle-favorite</i>	Add or remove a product from user favorites. If the product exist, it is removed else added	PATCH
		Is Favorite	<i>api/favorites/is-favorite</i>	Check if a product is user favorite	GET
		Get all users favorites	<i>api/favorites</i>	Get list of user and products	GET

favorites.

- 10 **Orders**
- 11 **Deliveries**
- 12 **Notification**

Note:

- ✓ We are using jwt token
- ✓ When a user login or register, a token will be returned which gives user access to the application according to its expiration date and the role of the user.
- ✓ Each user has a role (ADMIN, CUSTOMER, STORE, DELIVERY).
- ✓ Users have abilities to on each resource these abilities are (CREATE, READ, UPDATE, DELETE, RESTORE, MANAGE).
- ✓ A resource is an entity (Ex User, Product, Category...)
- ✓ By default a user is registered as a CUSTOMER

Resource	ADMIN	DELIVERY	STORE	CUSTOMER
User	ALL	CRUD	CRUD	CRUD
Store	CRUD	R	CRUD	R
Category	CRUD	R	R	R
Product	RD	R	CRUD	R
Review	R	R	R	CRUD
Cart	RUD	R	R	CRUD
Follow	R	R	RU	CRUD
Favorite	R	R	RU	CRUD
Delivery	ALL	RU	RU	CRUD
Order	ALL	RU	RU	CRU
Notification	ALL	CRU	CRU	R