

### User database

Description: Table contains basic details of the user.

Card		
	type	string
FK!	user_id	guid
	expiry_date	string
	account_no	string

User		
PK	Id	guid
	first_name	string
	last_name	string
	password	string
	email_addeess	string
	gender	string
	Phone_number	long
	user_type	string

address		
PK	id	guid
FK1	user_id	guid
	line1	string
	line2	string
	city	string
	state_name	string
	country	string
	zipcode	integer
	type	string
	delivery_phone	long

### Product database

Common_model		
PK	id	guid
	date_created	datetime
	date_updated	datetime
	Is_Active	book

product		
PK	id	guid
	name	string
	description	string
	image	byte[]
	category	string
	price	float
	visiblity	bool
	user_id	guid

cart		
FK1	user_id	guid
FK2	product_id	guid
	quantity	int

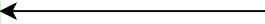
WishlistItems		
FK1	product_id	guid
FK2	WishListId	guid

Wishlist		
PK	id	guid
	user_id	guid
	name	string

order database

order		
PK	id	guid
	user_id	guid
	address_id	guid
	payment_id	guid
	total_amount	int
	payment_type	string
	Created_at	datetime

order_List		
FK2	order_id	guid
	status	string
	cart_id	guid

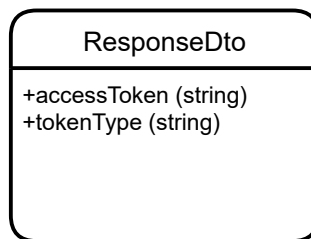
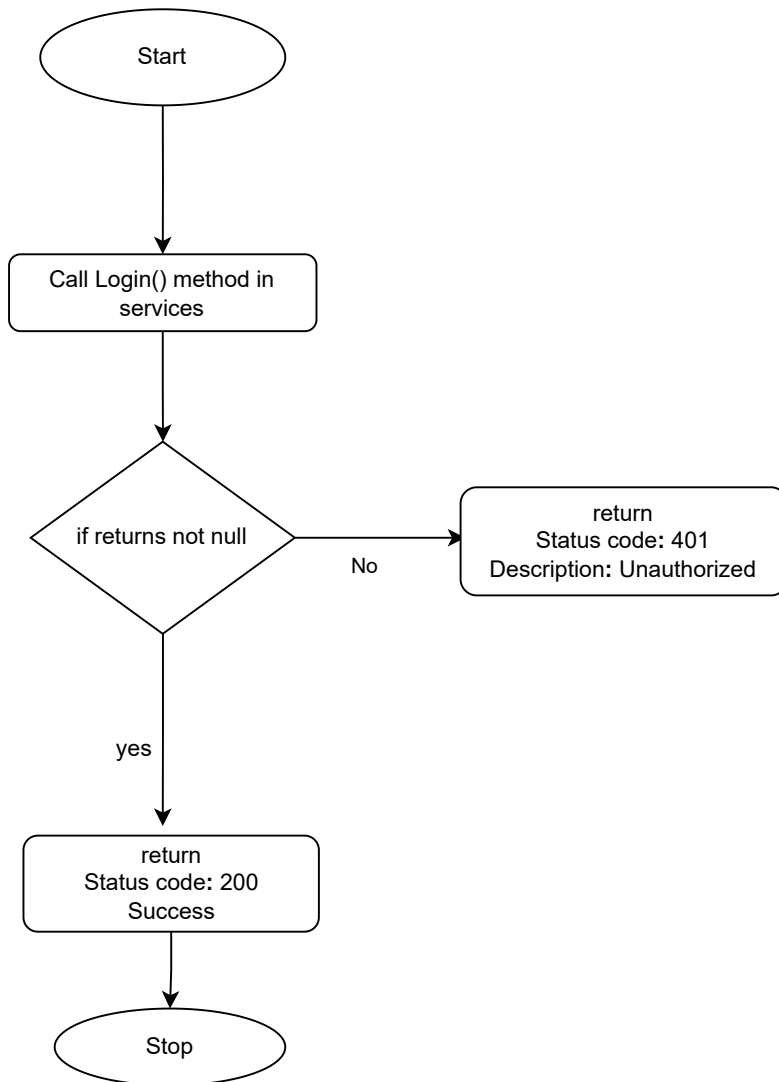


INPUT  
+String e-mail  
+String password

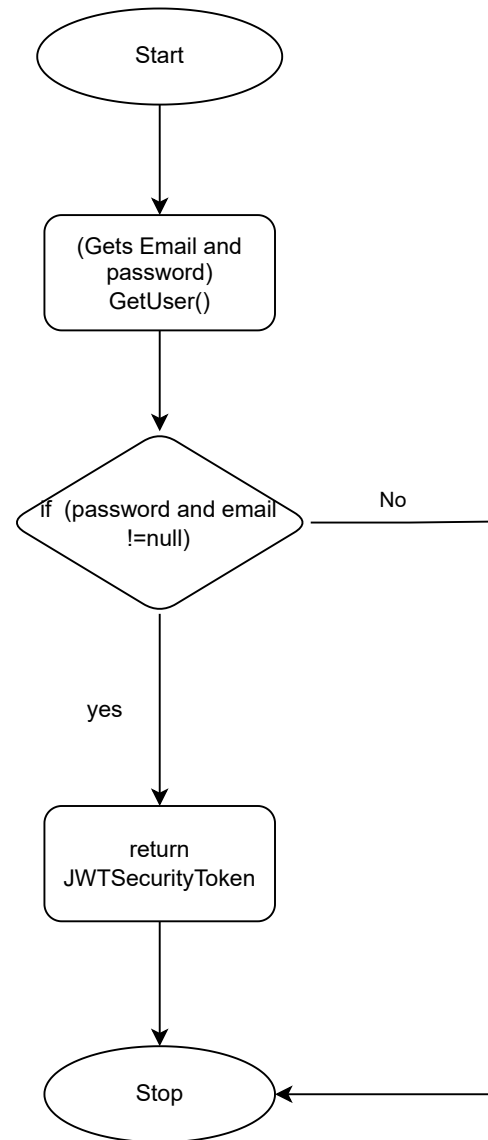
URL:  
</api/user/login>

Method :  
Post

### LoginFunction in Controllers



### LoginFunction in Services





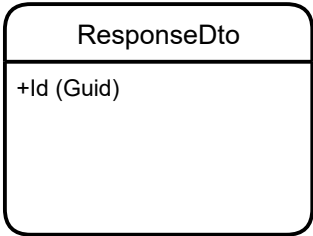
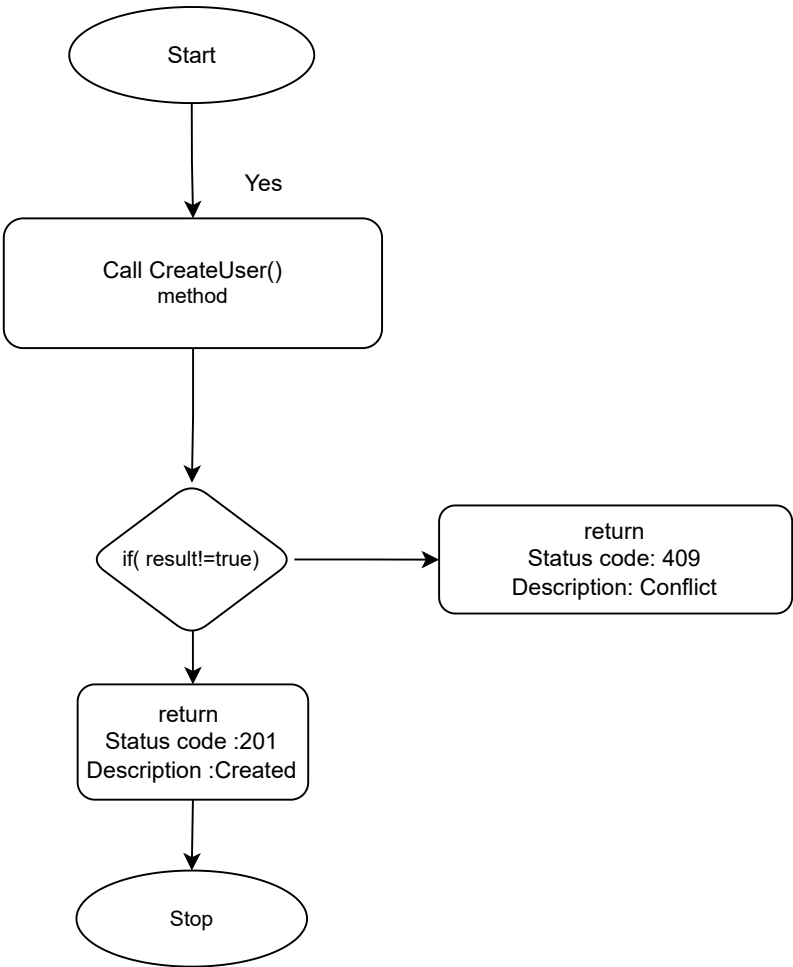
Text

Create User API

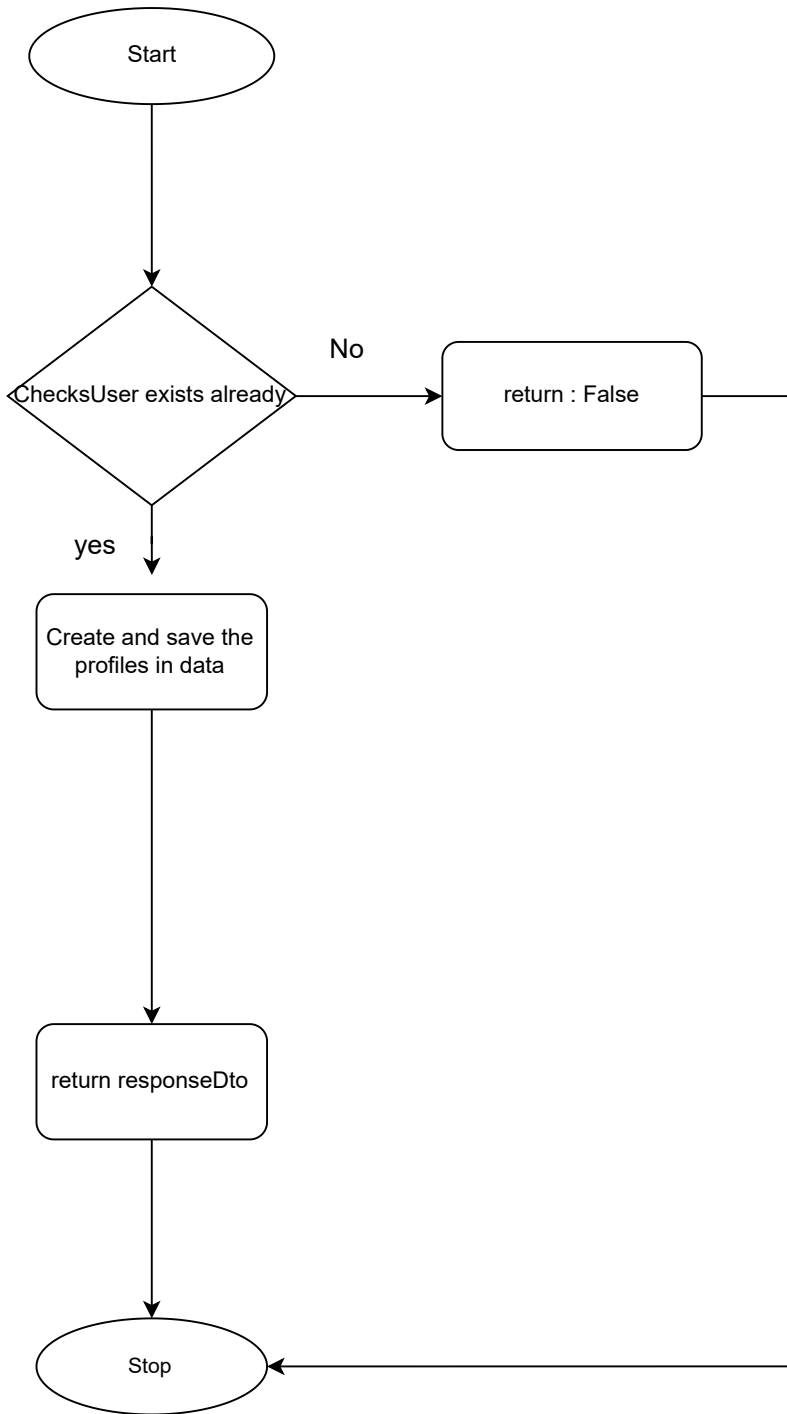
INPUT

+first\_name : string  
+last\_name: string  
+password:string  
+email\_address:string  
+phone\_number : string  
+user\_type:string  
+gender : string

URL  
[/api/user](#)  
Method :  
Post



## AddUser





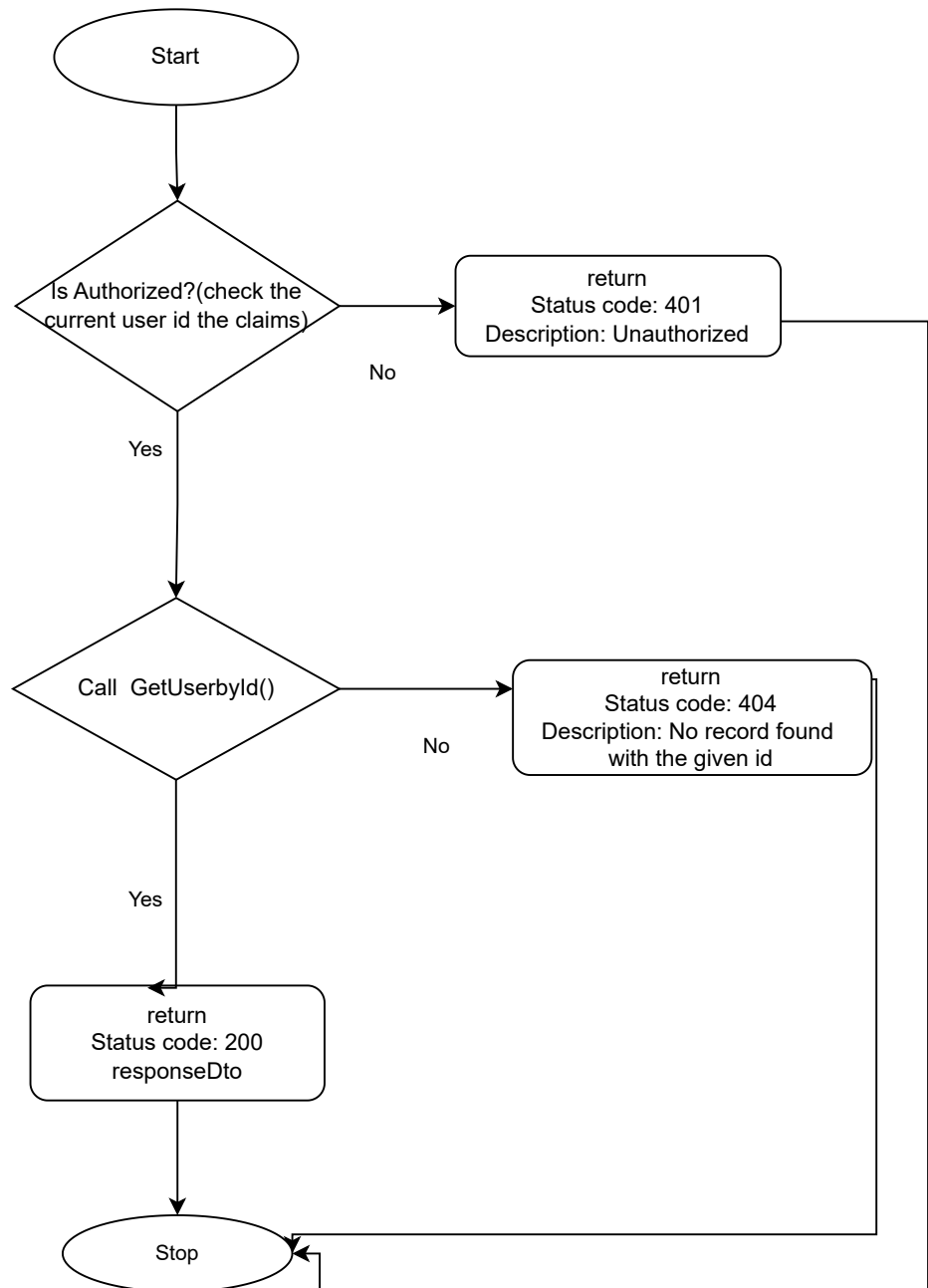


## GetUserById

Input:  
+id: guid

URL  
[/api/user/{id}](#)

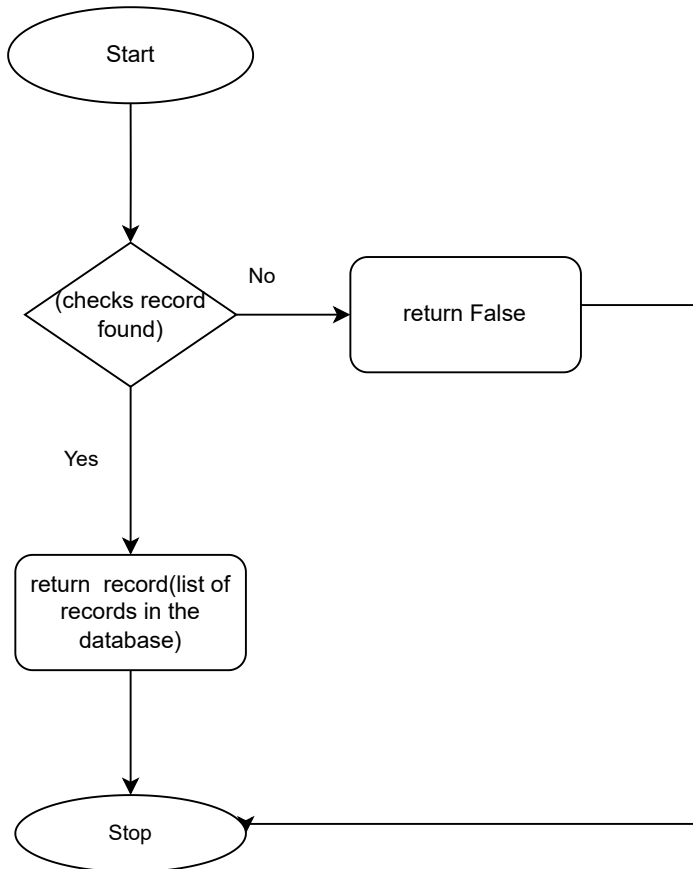
Method :  
Get



R

- +first\_na
- +last\_na
- +passwo
- +email\_
- +phone
- +user\_ty
- +gender
- +List<A
- +List<pa

### getUserByld in Services

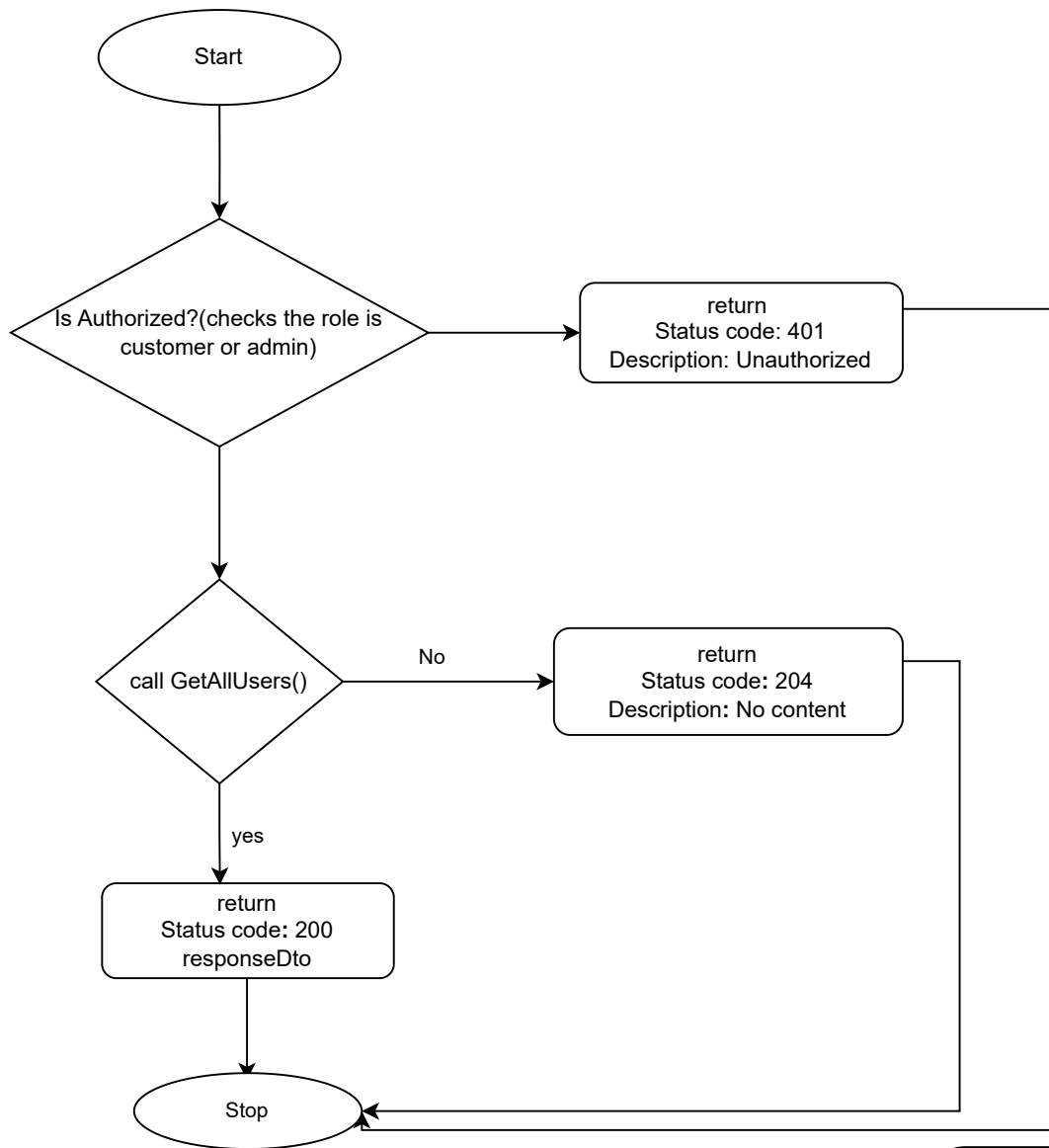


ResponseDto
ame : string
ame: string
ord:string
address:string
_number : string
ype:string
: string
ddress>
ayment>

## Get All user

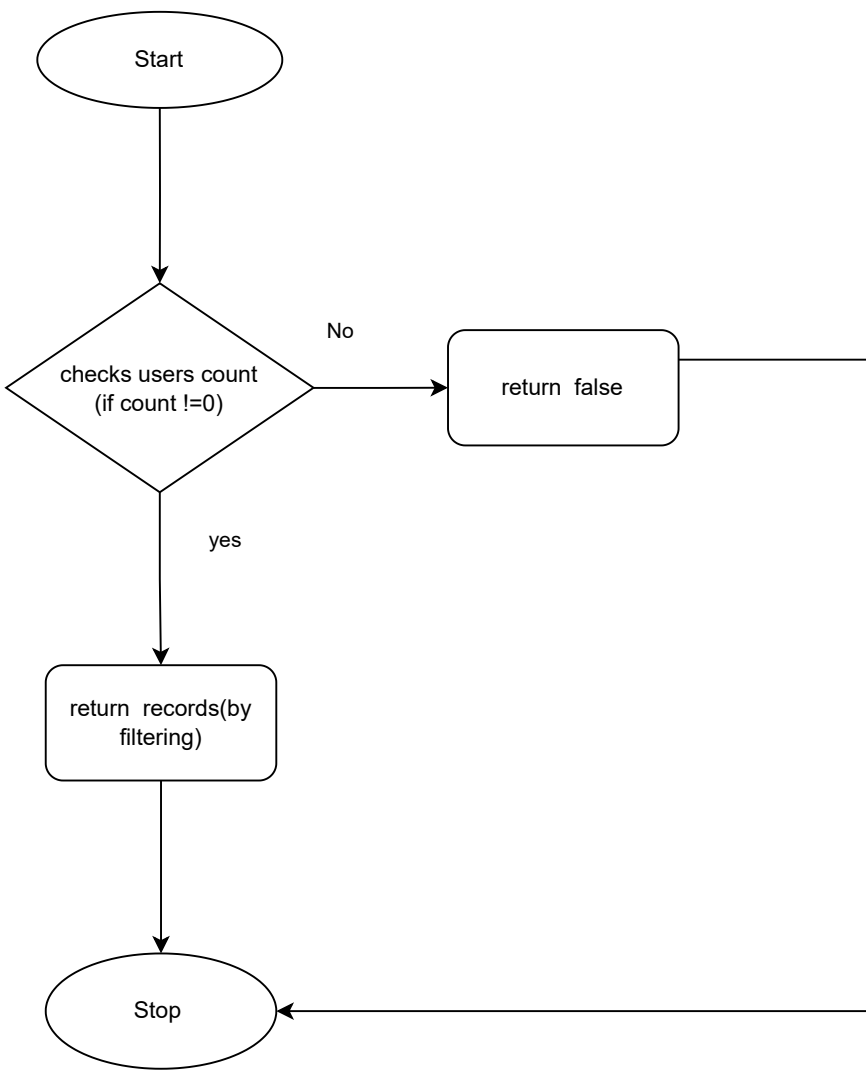
URL  
[/api/user](#)

Method :  
get



### ResponseDto

```
+first_name : string
+last_name: string
+password:string
+email_address:string
+phone_number : string
+user_type:string
+gender : string
+List<Address>
+List<payment>
```



## Update User

### INPUT

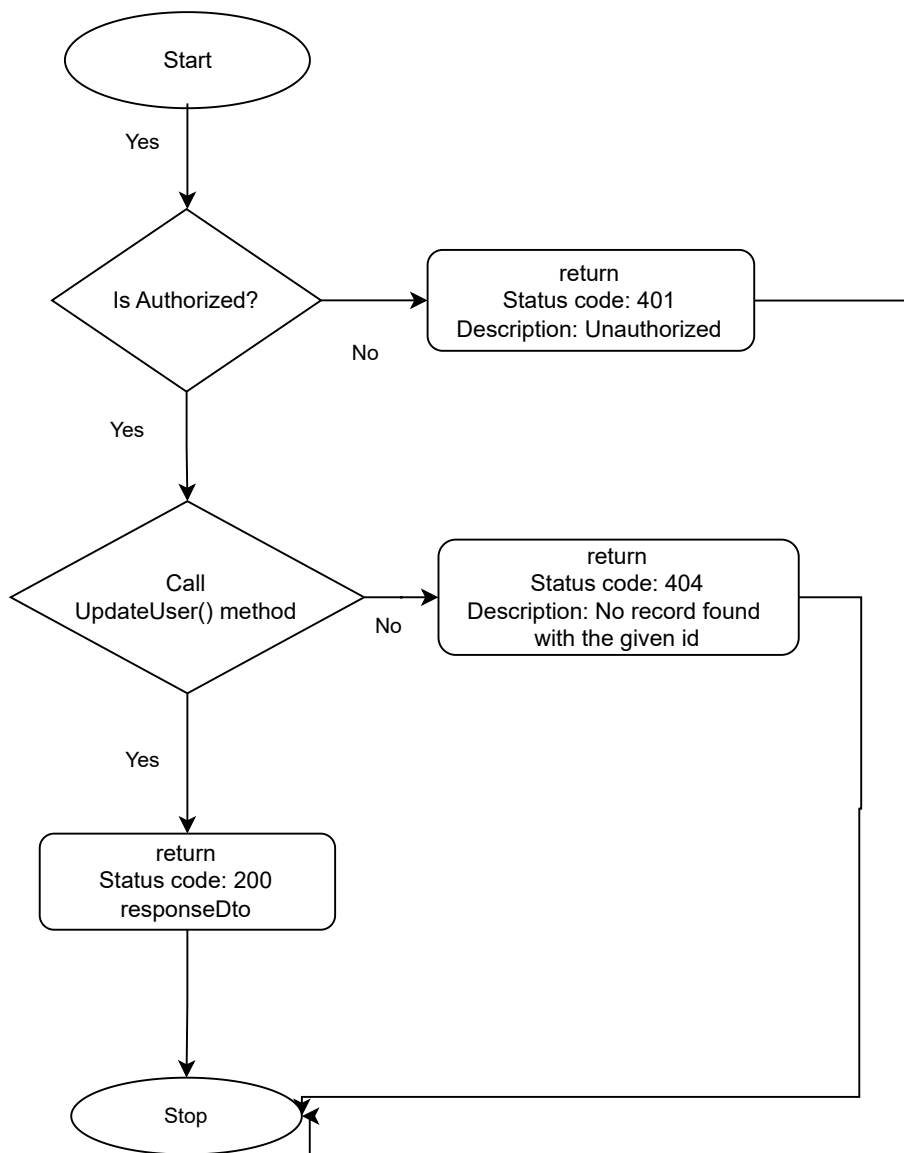
+first\_name : string  
+last\_name: string  
+password:string  
+email\_address:string  
+phone\_number : string  
+user\_type:string  
+gender : string

### URL

/api/user/{id}

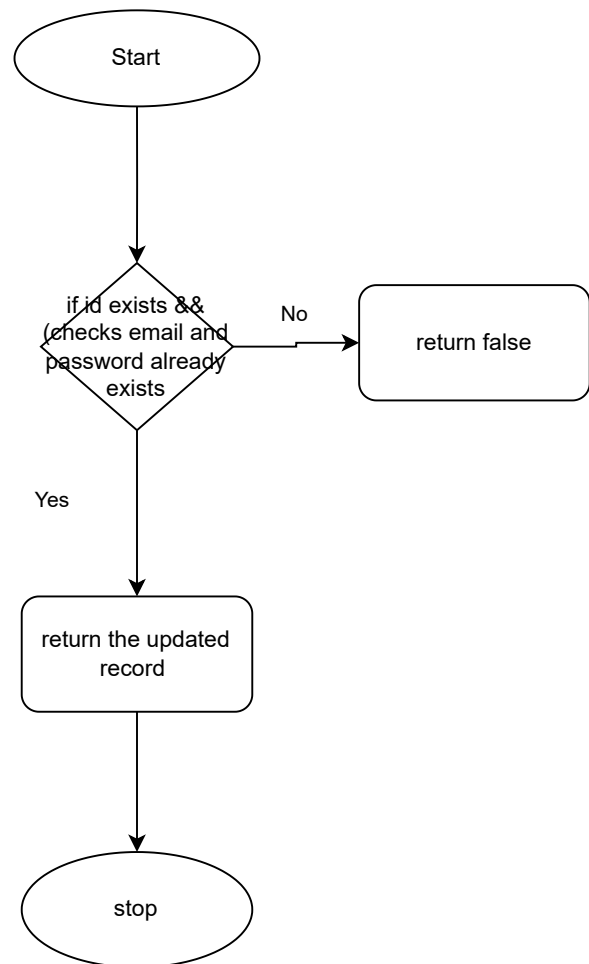
### Method :

Put



### ResponseDto

+first\_name : string  
+last\_name: string  
+password:string  
+email\_address:string  
+phone\_number : string  
+user\_type:string  
+gender : string  
+List<Address>  
+List<payment>

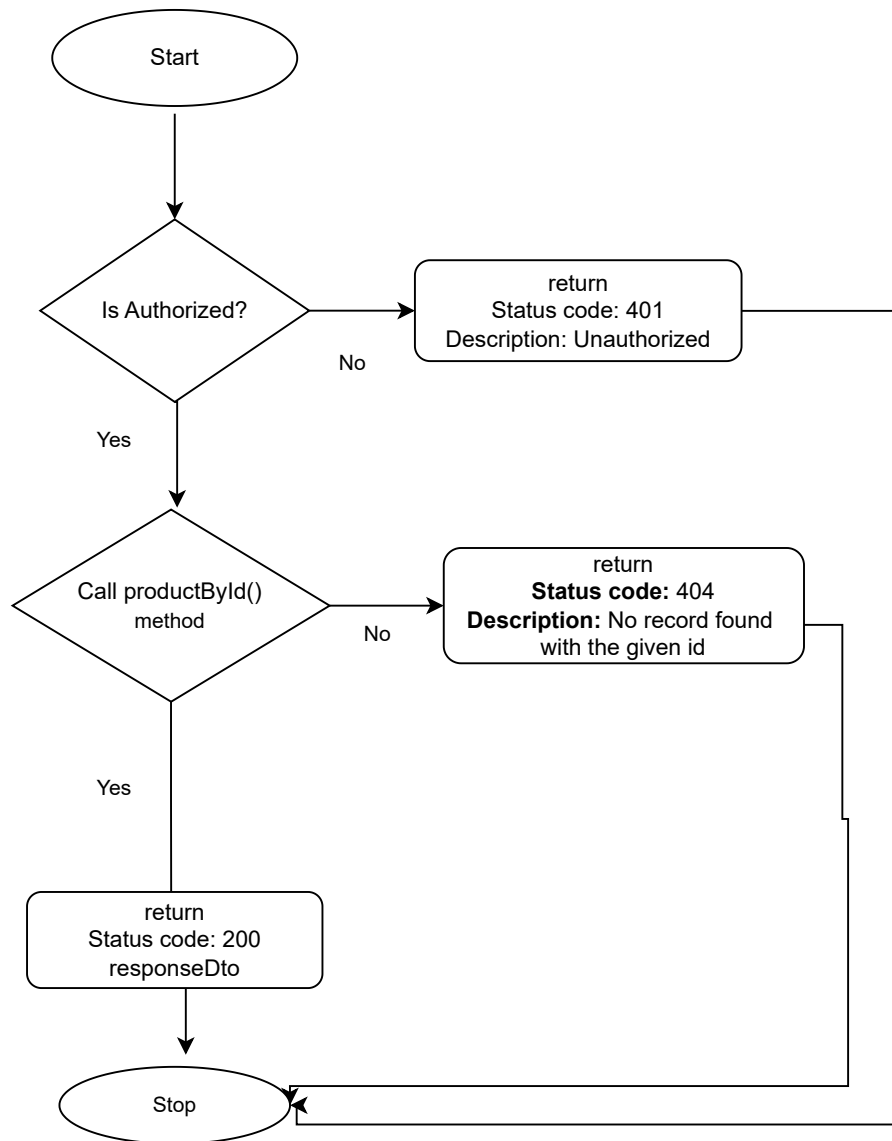


## Delete UserById

**Input:**  
**+id (integer)**

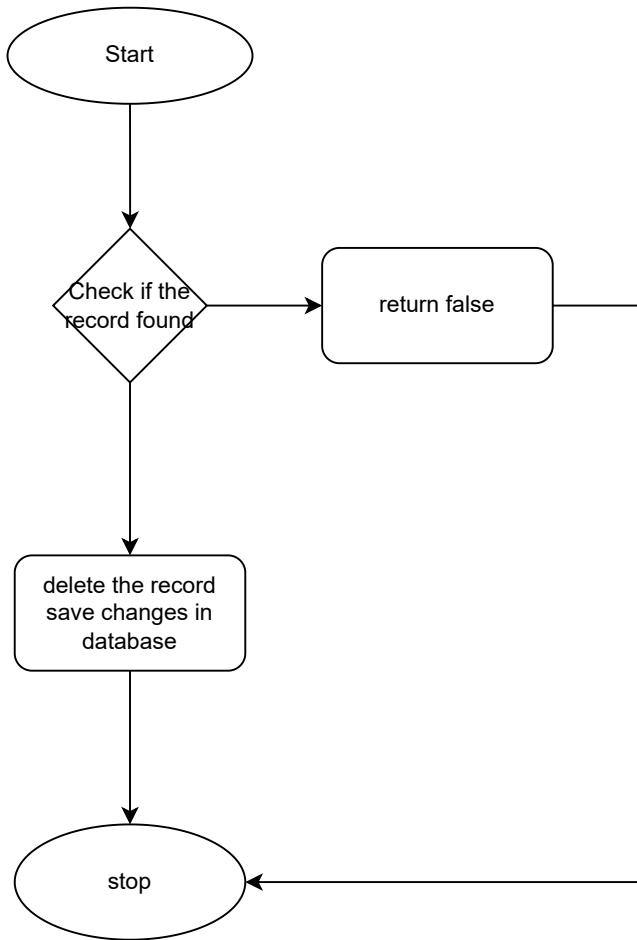
URL  
[/api/user.{id}](#)

**Method :**  
Delete





## DeleteproductbyId



Create products API

INPUT

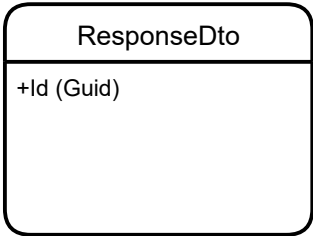
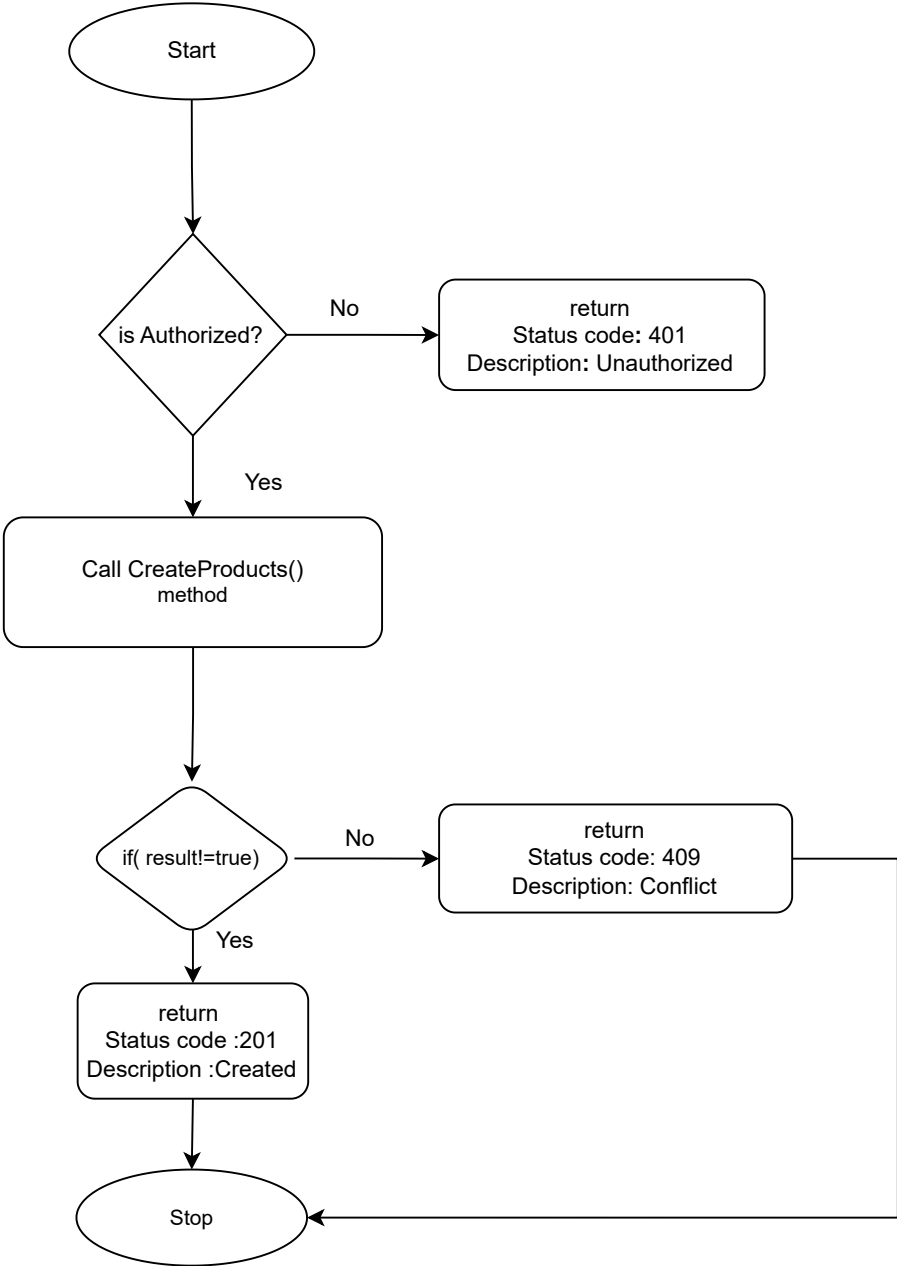
- +String product\_name
- +String product\_description
- +byte[] image
- +string availability\_count
- +float price
- +Category

URL

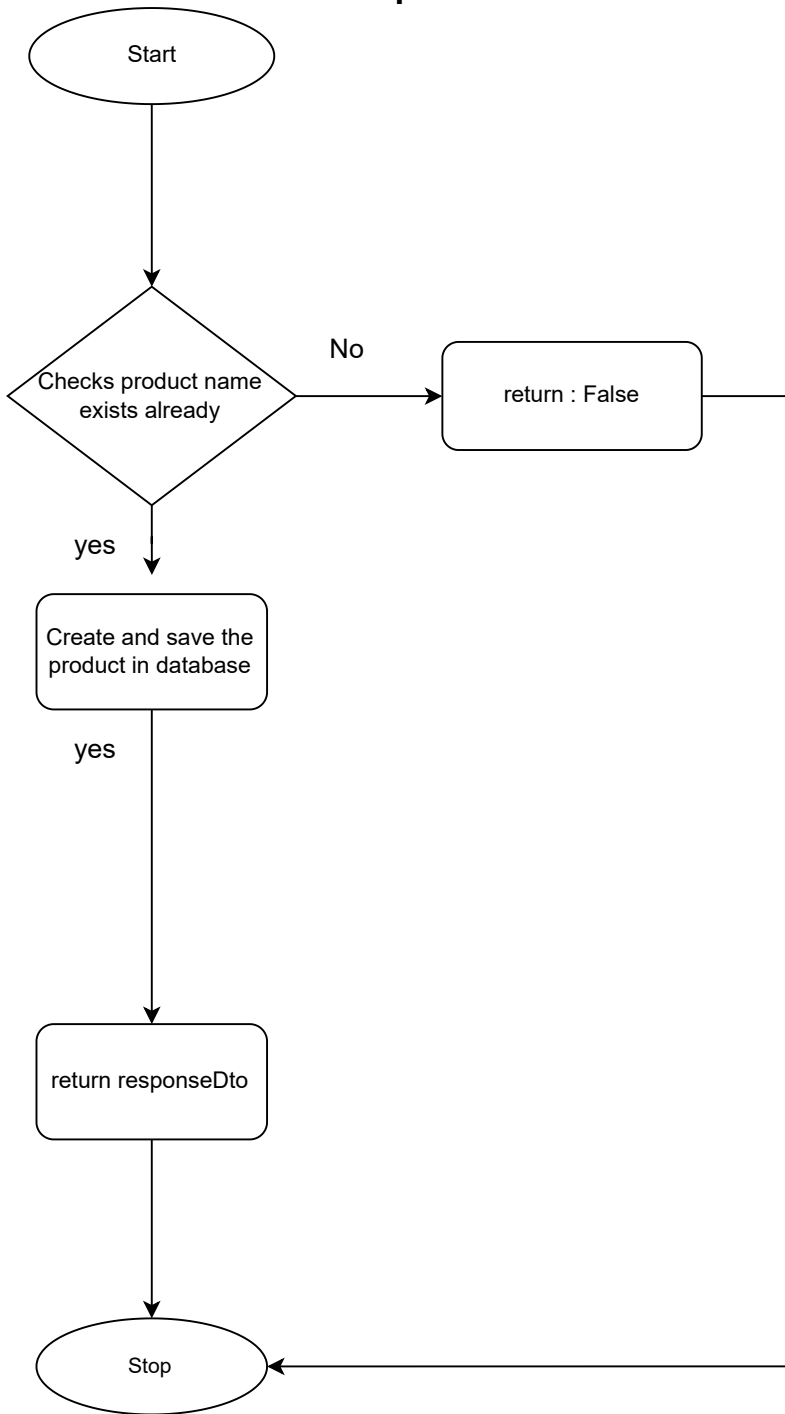
/api/product

Method :

Post



## Addproducts

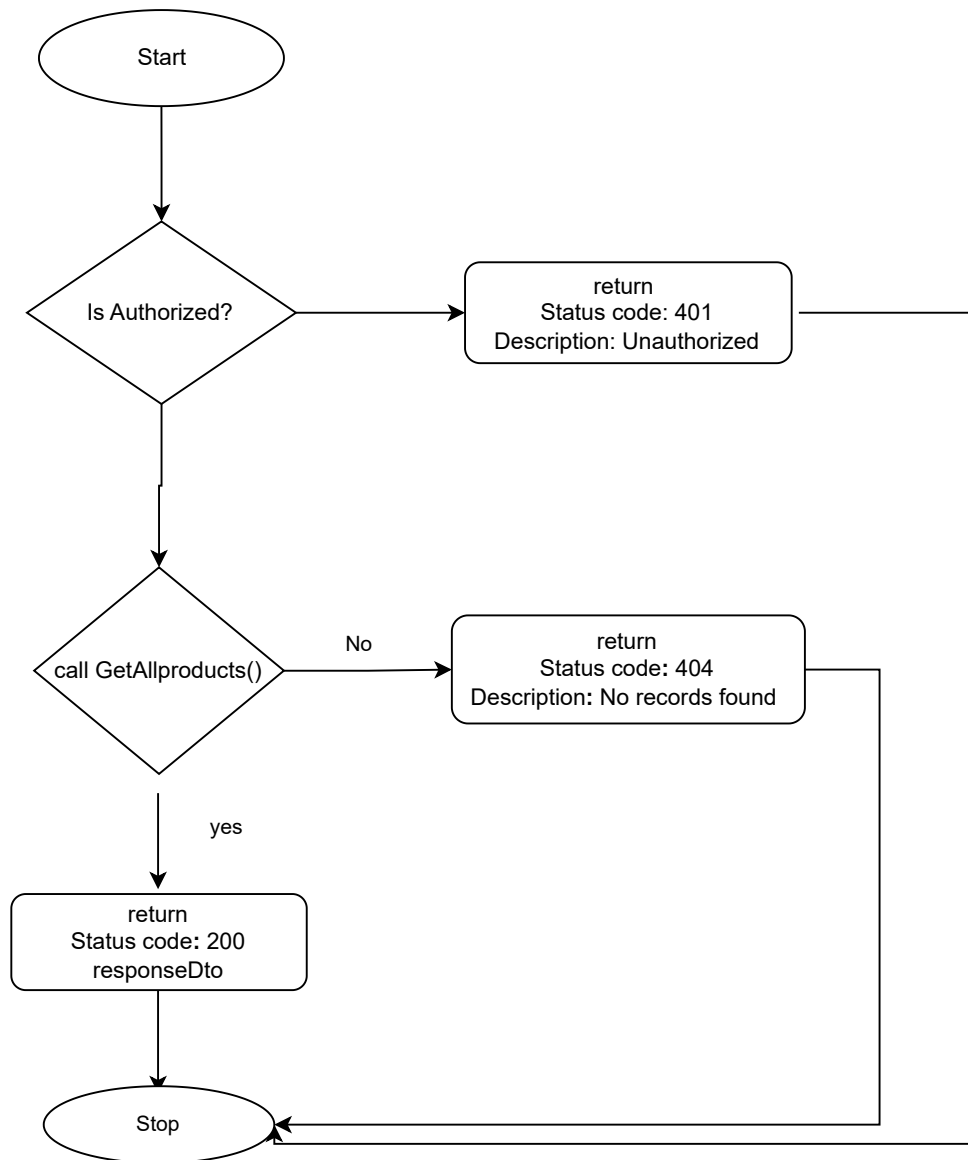




## Get All products

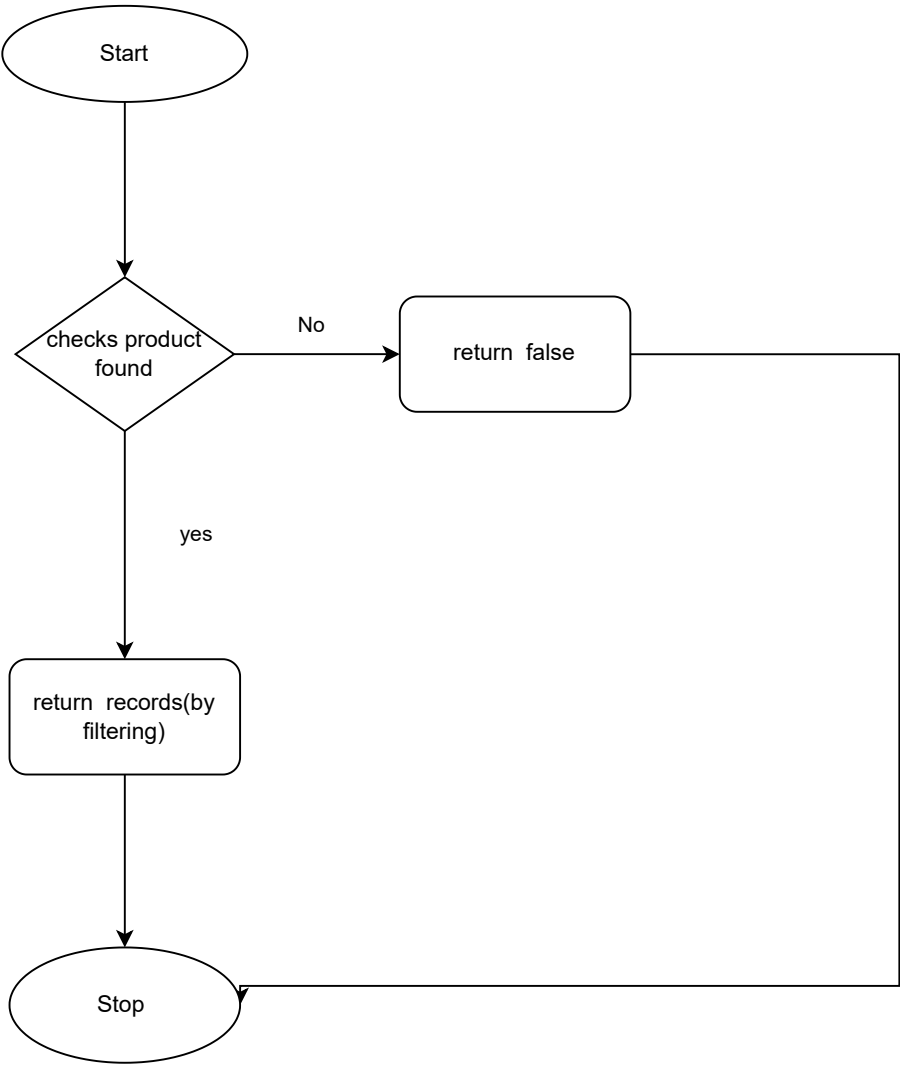
URL  
</api/product>

Method :  
Get



### ResponseDto

```
+String product_name
+String product_description
+byte[] image
+string availability_count
+float price
```



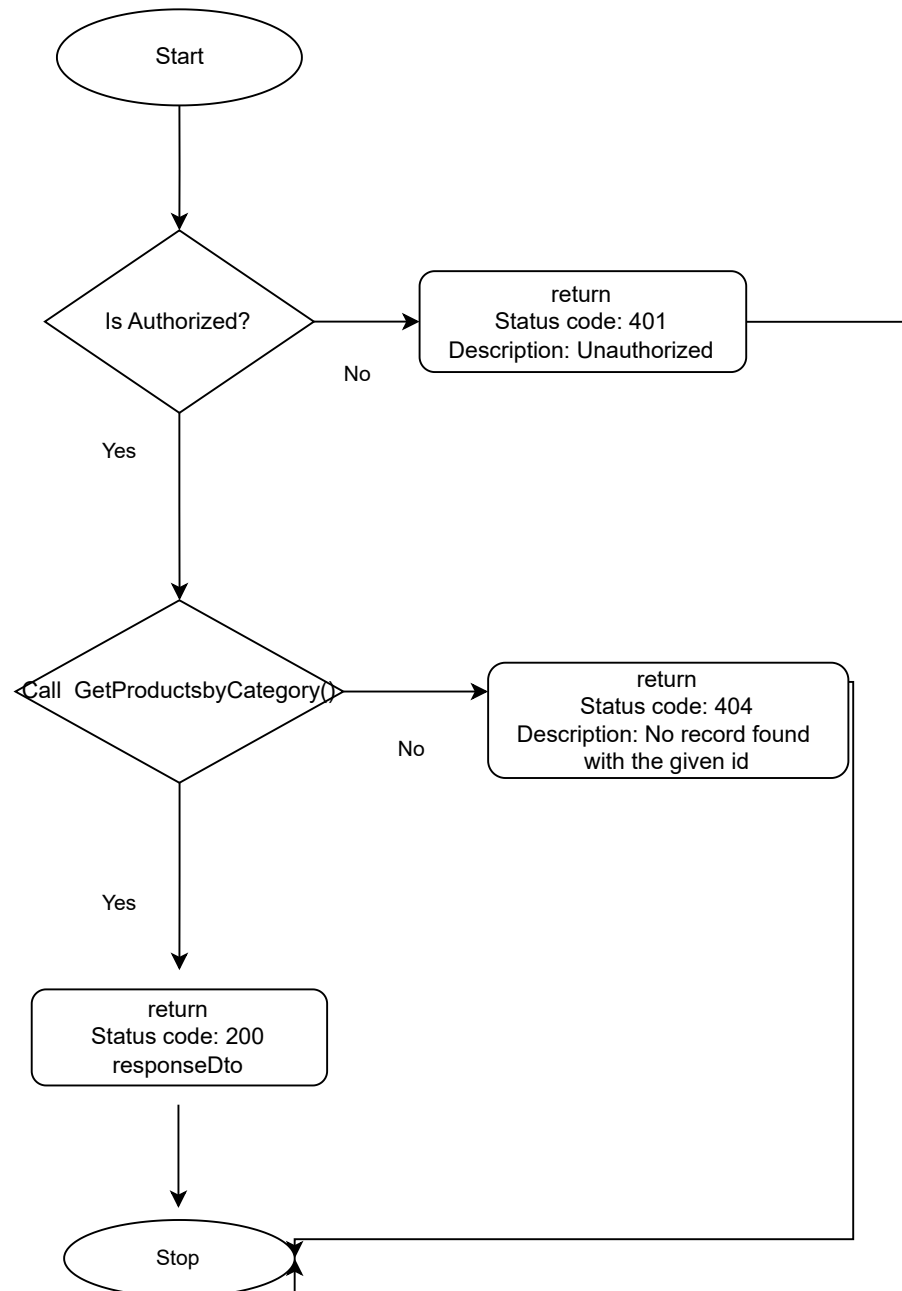
## Get products by category

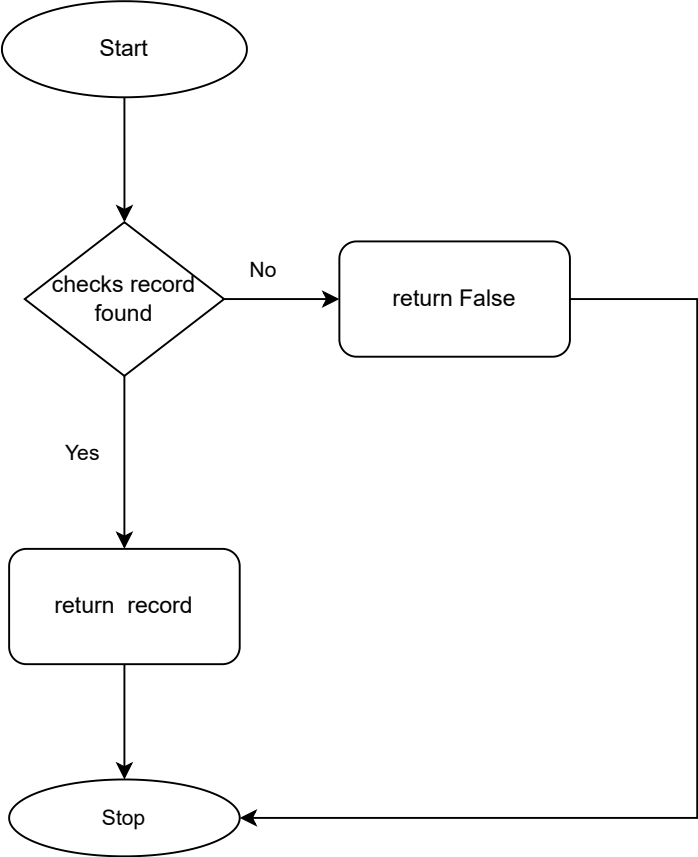
**Input:**  
**+category (string)**

**URL**  
</api/products/{category}>

**Method :**  
Get

ResponseDto
+String product_name +String product_description +byte[] image +string availability_count +float price







## Update Address Book API

### INPUT

+String product\_name  
+String product\_description  
+byte[] image  
+string availability\_count  
+float price  
+Category

### URL

/api/product/{id}

### Method :

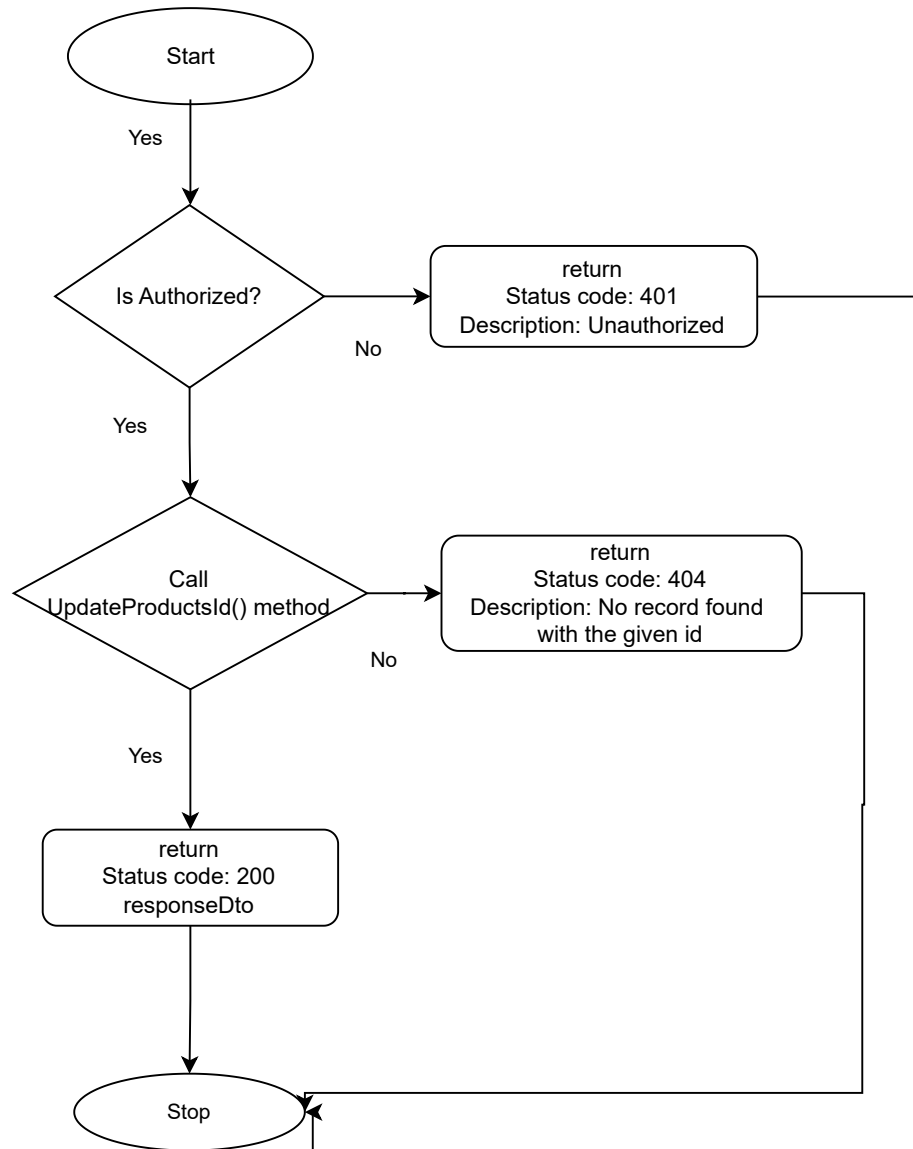
Put

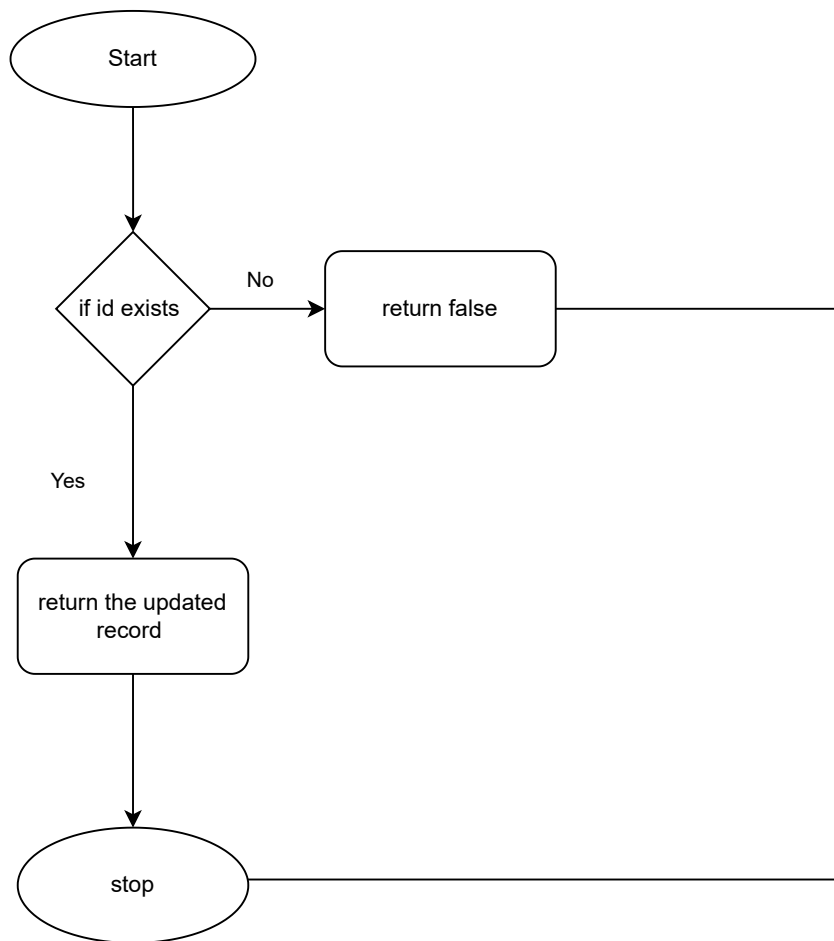
### Input:

+id (integer)

### ResponseDto

+String product\_name  
+String product\_description  
+byte[] image  
+string availability\_count  
+float price



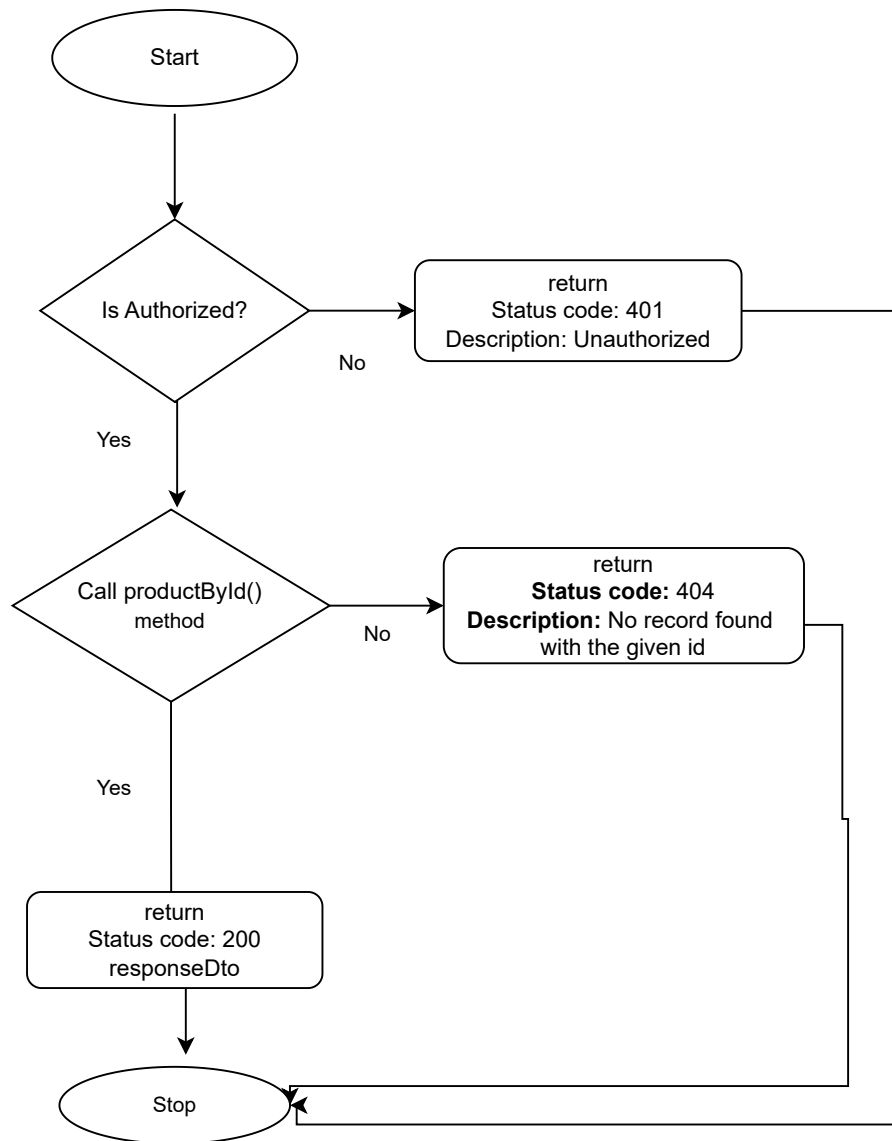


## Delete products by Id

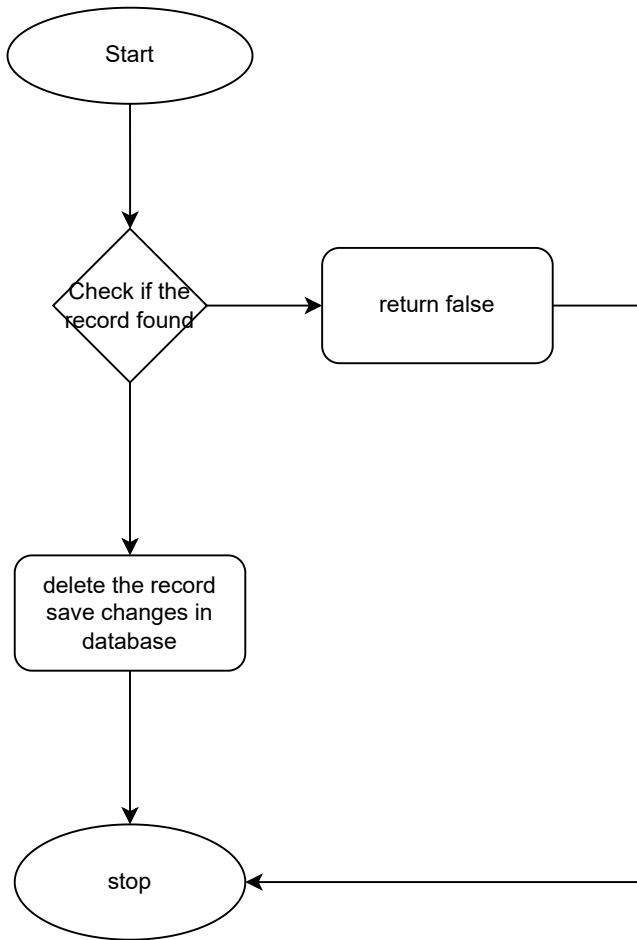
**Input:**  
+id (integer)

URL  
[/api/acoount.{id}](#)

**Method :**  
Delete



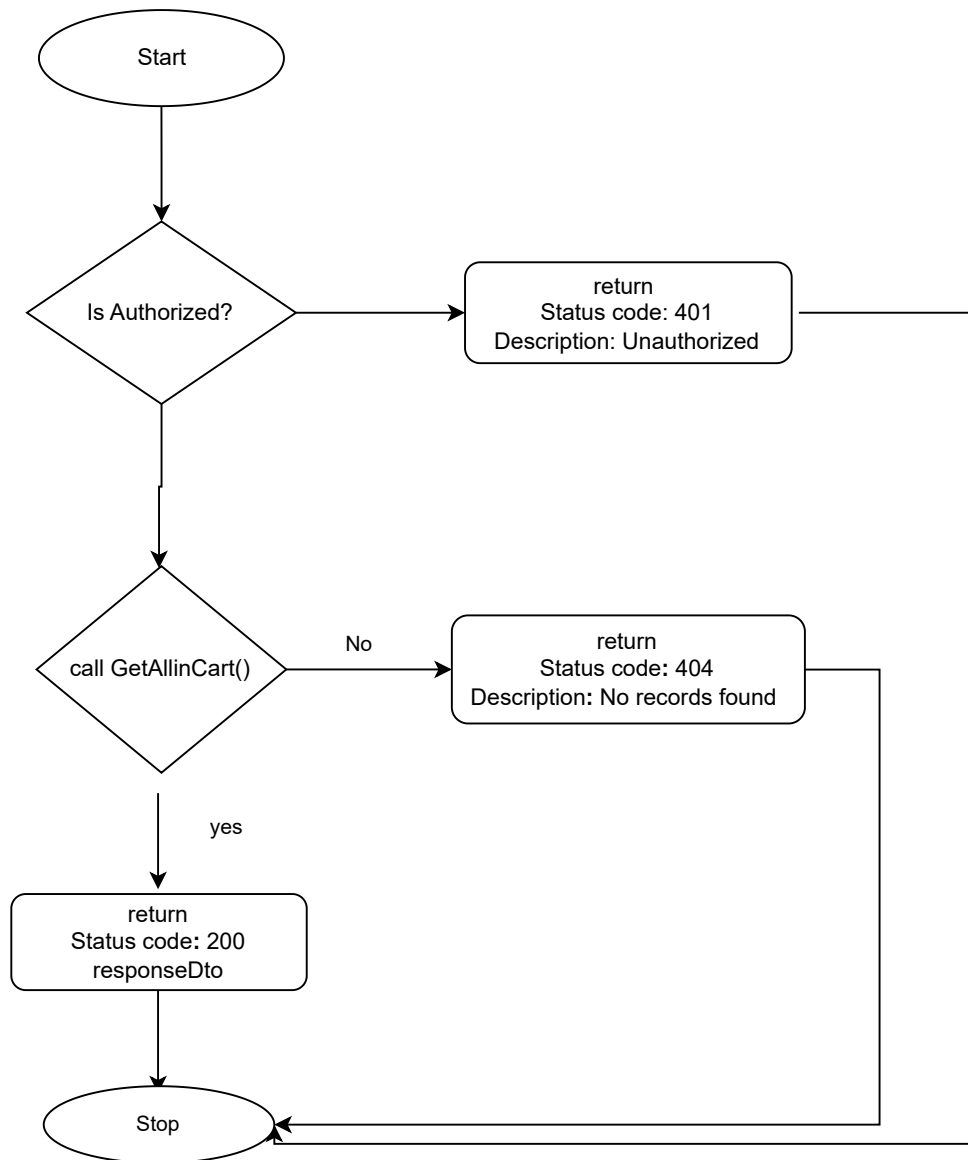
## DeleteproductbyId



## Get All products in cart

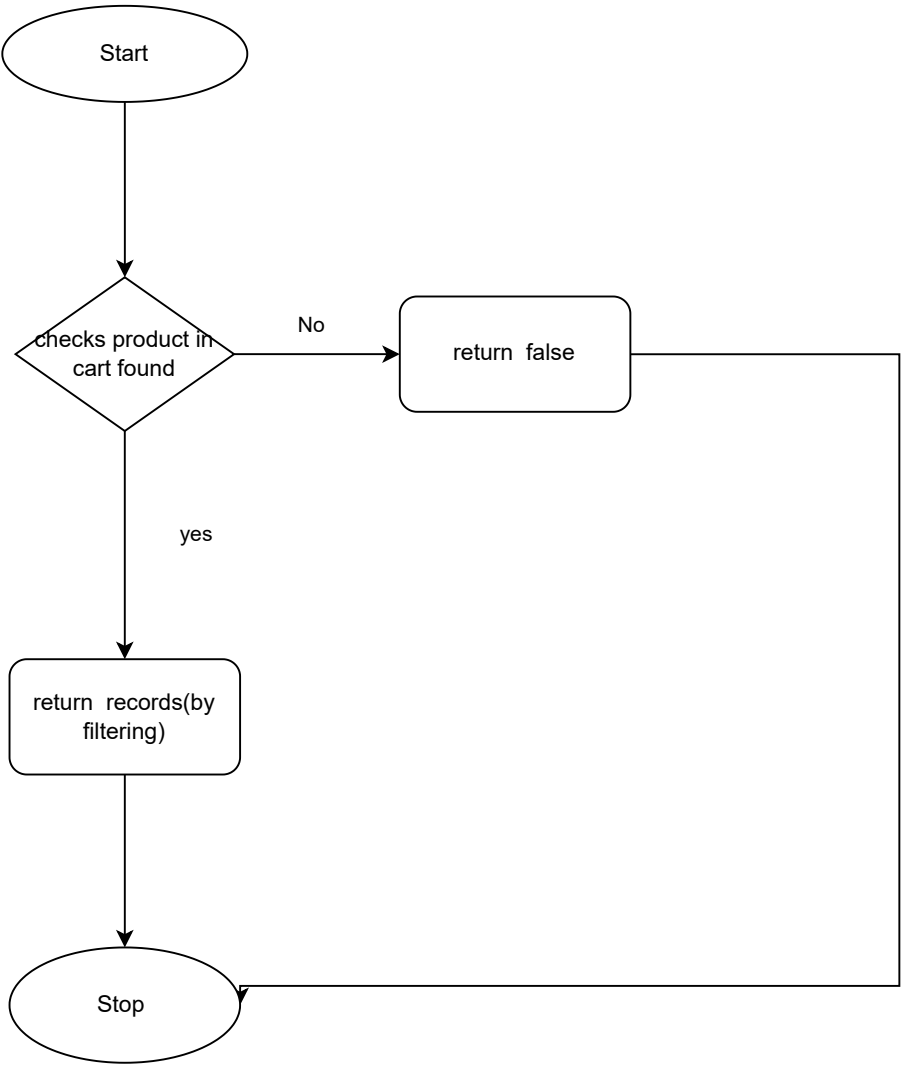
URL  
</api/product/cart>

Method :  
Get



### ResponseDto

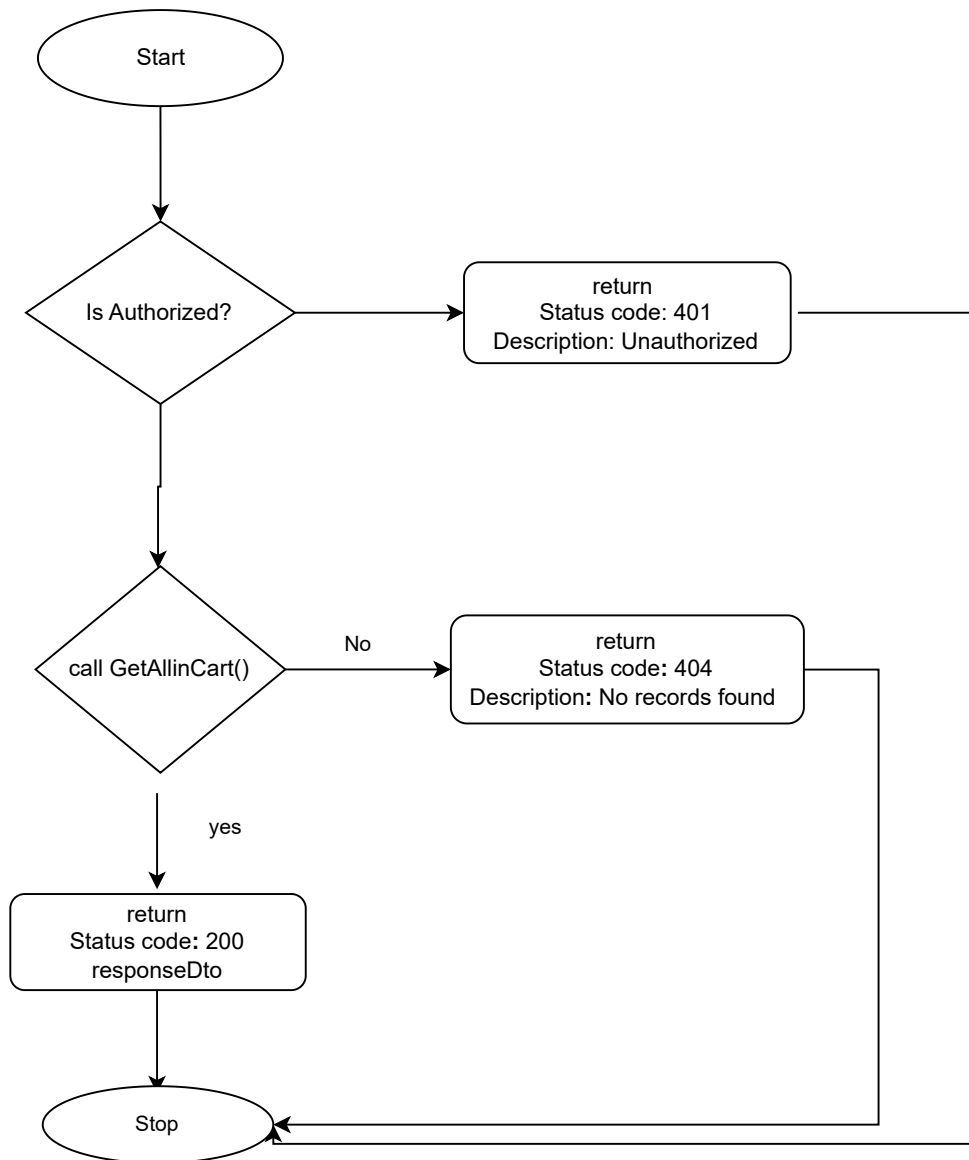
```
+String product_name
+String product_description
+byte[] image
+string availability_count
+float price
```



## Get All products

URL  
</api/product/cart>

Method :  
Get



ResponseDto
+String product_name
+String product_description
+byte[] image
+string availability_count
+float price

