IT4020
Modern Topics in IT
4th Year, 1st Semester

Group Project 4- Group Report

Group ID – MTIT-028

Submitted By:

Registration Number	Name	Workload (Micro-Services)
IT19121802	Sharmilan S	Payment Management
IT19241760	Nilakshana R	Product Management
IT 19029832	Kithusshand R	Order Management
IT19119618	Sriram R	User Management

Introduction

ES is exactly analogous to a marketplace on the internet. ES (also referredto as EC, e-commerce) consist primarily of the distributing, buying, selling, marketing, and servicing of products or services over electronic systems such as the internet and other computer networks. E-commerce follows – that is, buyers and sellers exchange and transport goods from one place to another. But rather than conducting business in the tradition way-in stores and other “brick and mortar” buildings or through mail order catalogs and telephone operators- in e- commerce buyers and sellers transact business over networked computers.

E-commerce also has some disadvantages, however. Consumers are reluctant to buy someproducts online. Online furniture business, for example, have failed for the most part because customers want to test the comfort of an expensive item such as a sofa before they purchase it. Consumers also need to be reassured that credit card transaction are secure and that their privacy is respected.

‘

1. Members' Details and Workload Allocation

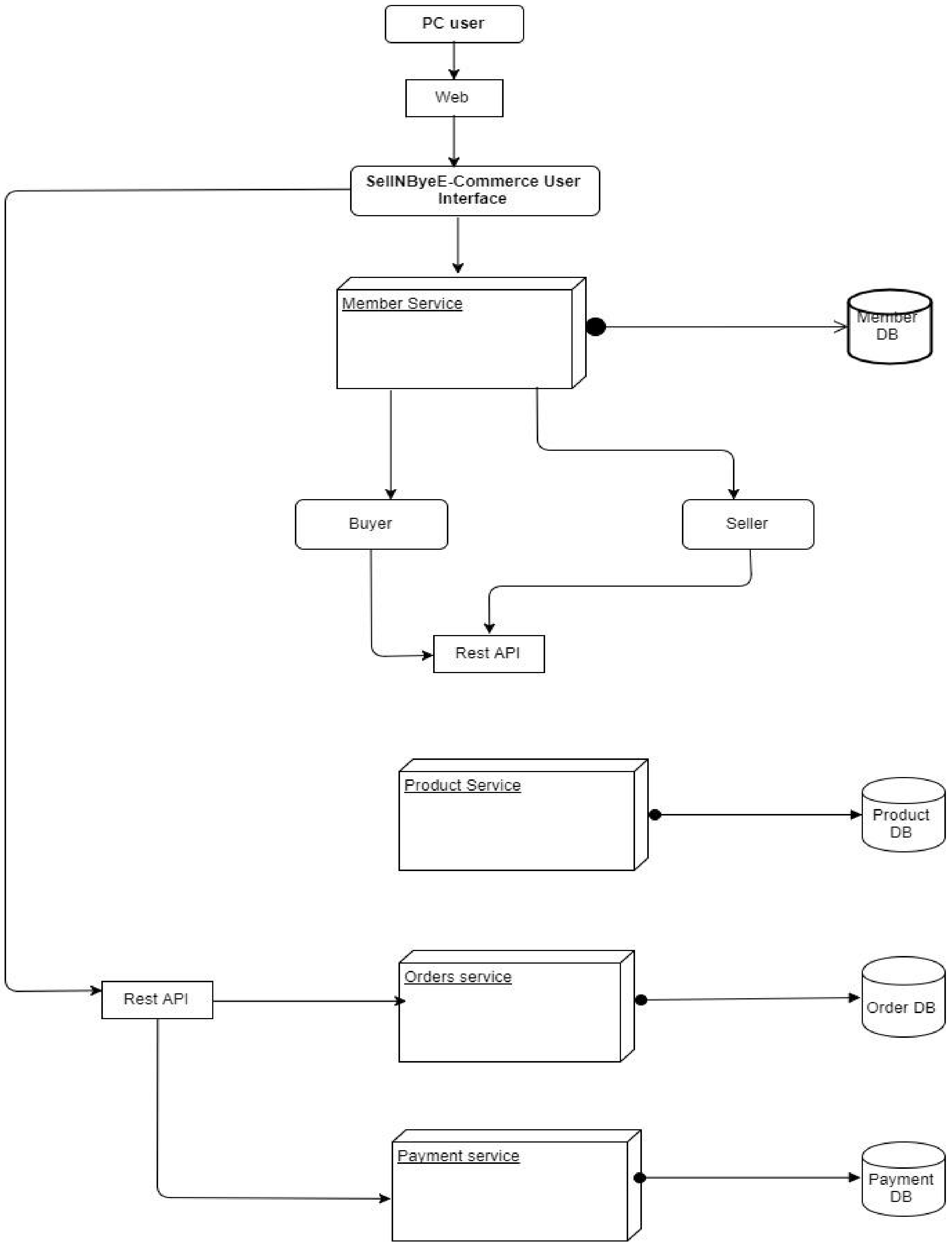
IT Number	Name	Web Service	Description of the Web Service
IT19121802	**Sharmilan S**	Payment Management	• Add Payment Details • Update Payment Details. • Delete Payment Details. • View Payment Details.
IT19241760	**Nilakshana.R**	Product Management	• Add a new product. • View product details • Update product details • Delete product fromsystem.
IT 19029832	**Kithusshand R**	Order Management	• Add Order • Update Order • Delete Order • View Order
IT19119618	**Sriram R**	User Management	• Add User • Update User • Delete User • View User

Clickable Link (VCS repo) – https://github.com/SharmBB/MTIT-4

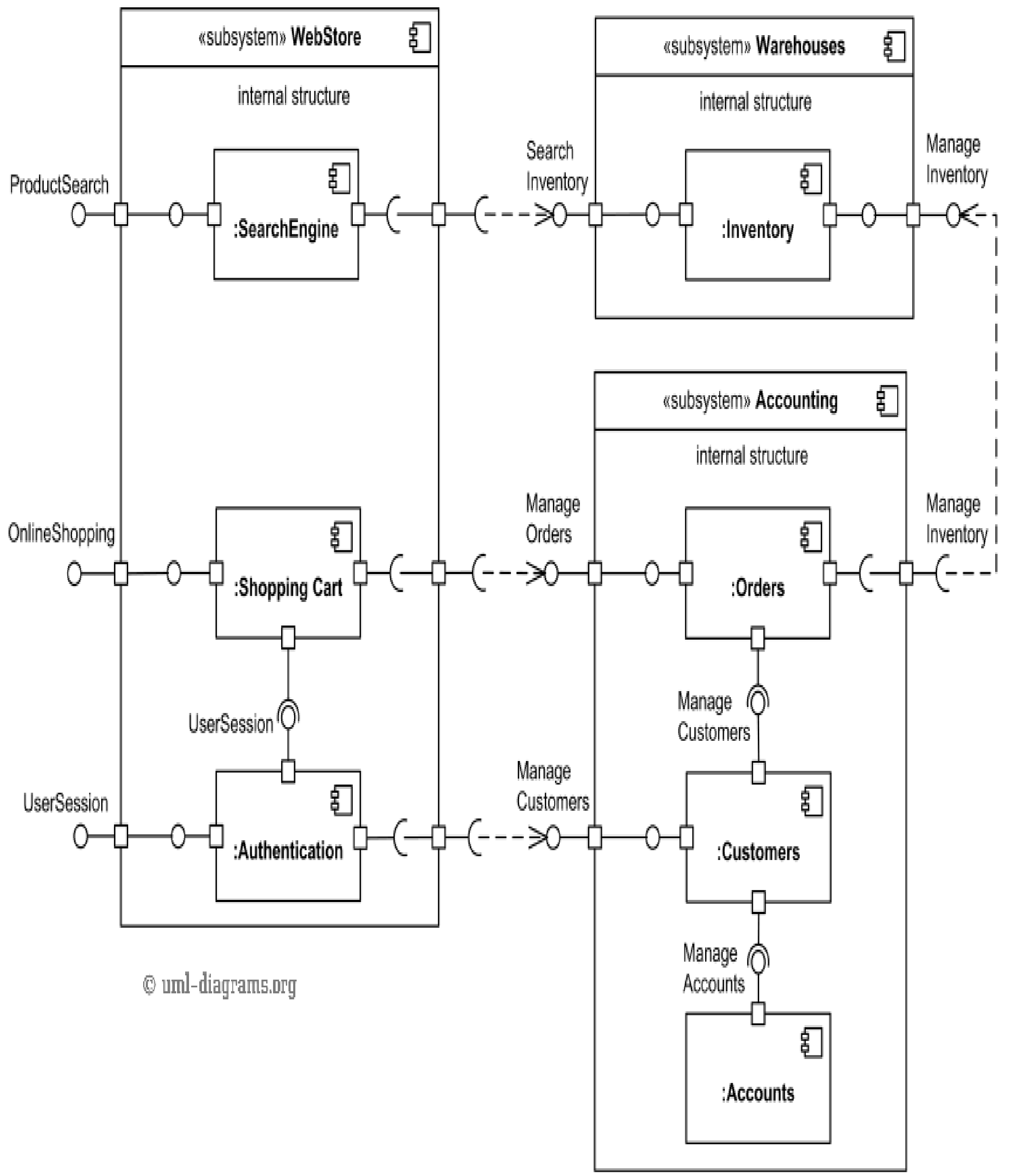
Tools and Technology used in this project.

Technology:	Laravel, PHP web framework
Database:	MySQL Database.

Overall architecture



Component Diagram



5. Individual Services

Student IT Number: IT19121802 **Web Service: Payment Management**

Service Development

- Technologies Used: Laravel, PHP web framework
- IDE : Visual Studio
- Database : MySQL
- Testing : Insomnia

//Read

API for get all the Payment details in the database (GET Request)

URL: http://127.0.0.1:8000/api/getPayment

Resource: PaymentService

Request: GET **getPayment**

Responce: benefactor,payer,amount,account_Number,bank

//Insert

API for register Payment to the database (POST Request)

URL: http://127.0.0.1:8000/api/add

Request: POST **add**

Media: Application form URL Encoded

```
{
        “benefactor”: “Sharmilan”,
        “payer”: “Nilu”,
        “amount”: “2000”,
        “accountNumber”: “200034678”,
```

“bank”: “Commercial Bank”

}

Response: {

“ errorMessage” = false,
message = “Payment Added Successfully”
}

//Delete

API for Delete Payment to the database (DELETE Request)

URL: **http://127.0.0.1:8000/api/deletePayment**

Request: DELETE **deletePayment**

Media: Application form URL Encoded

```
{
        “id":5
}
```

Response: {

“ errorMessage” = false,
message = “Payment details Deleted Successfully”
}

//Update

API for Update Payment to the database (PUT Request)

URL: **http://127.0.0.1:8000/api/updatePayment**

Request: PUT **updatePayment**

Media: Application JSON

```
 {
 “id”:”2”
 “benefactor”: “Sharmilan”,
          “payer”: “Nilakshana”,
          “amount”: “10000”,
          “accountNumber”:“2000346”,
          “bank”: “Commercial Bank”
  }
```

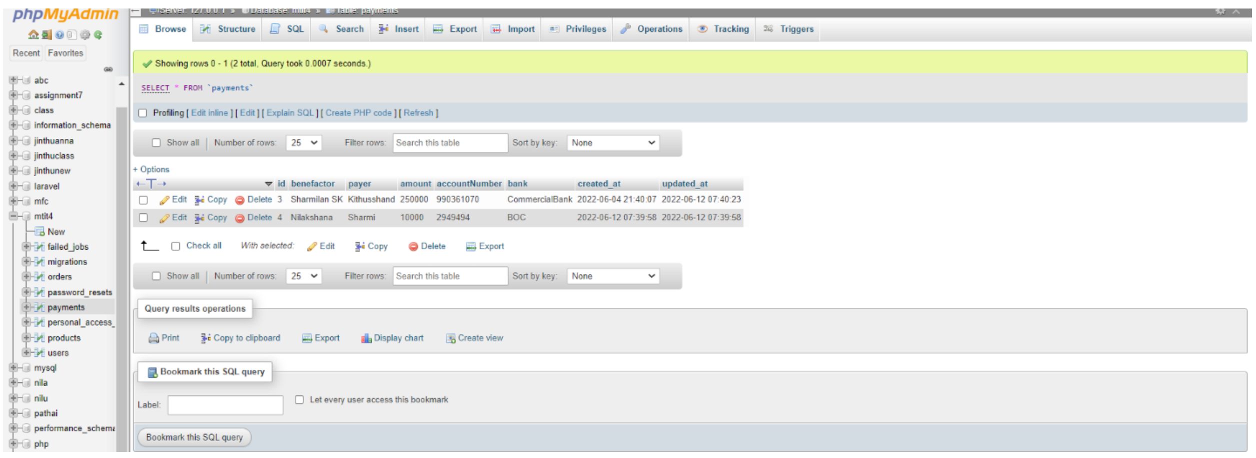
```
Response:  {
 “ errorMessage” = false,
    message =  “Payment update Successfully!!!”
}
```

Testing Methodology and Results

Test ID	Test Description	Test Inputs	Expected Output	Actual Output	Result (pass/fail)
01	Inserting Data	" benefactor ":" Sharmilan ", " payer ":" Nilu ", " amount ":"2000", "accountNumber":"20003468, " bank ":" Commercial Bank "	Inserted successfully	Inserted successfully	Pass
02	Updating Data	" id ":3, " benefactor ":" Sharmilan ", " payer ":" Nilu ", " amount ":"2000", "accountNumber":"20003468, " bank ":" Commercial Bank "	Updated successfully	Updated successfully	Pass
03	Deleting Data	id = 3	Deleted successfully	Deleted successfully	Pass

Screenshots

Database



Showing rows 0 - 1 (2 total, Query took 0.0007 seconds.)

```
SELECT * FROM `payments`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

id	benefactor	payer	amount	accountNumber	bank	created_at	updated_at
3	Sharmilan SK	Kithusshand	250000	990361070	CommercialBank	2022-06-04 21:40:07	2022-06-12 07:40:23
4	Nilakshana	Sharmi	10000	2949494	BOC	2022-06-12 07:39:58	2022-06-12 07:39:58

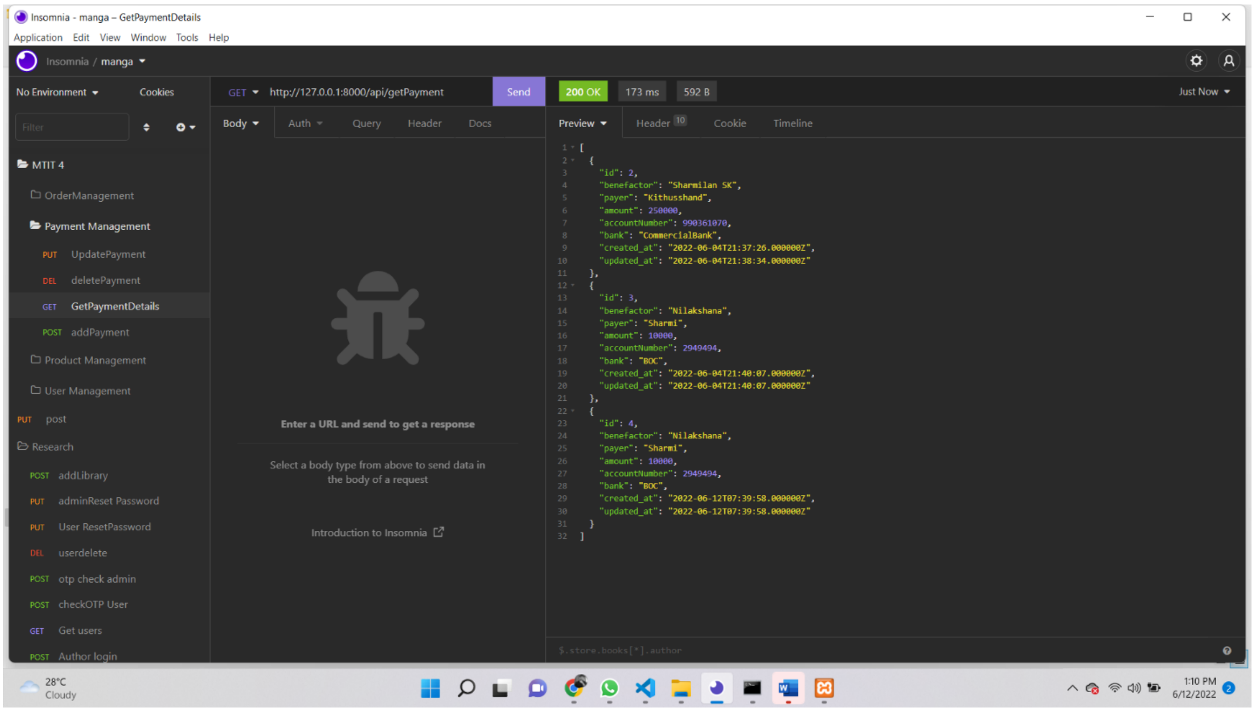
Check all With selected: Edit Copy Delete Export

Query results operations: Print, Copy to clipboard, Export, Display chart, Create view

Bookmark this SQL query

Label: Let every user access this bookmark

API for get all the Payment details in the database (GET Request)

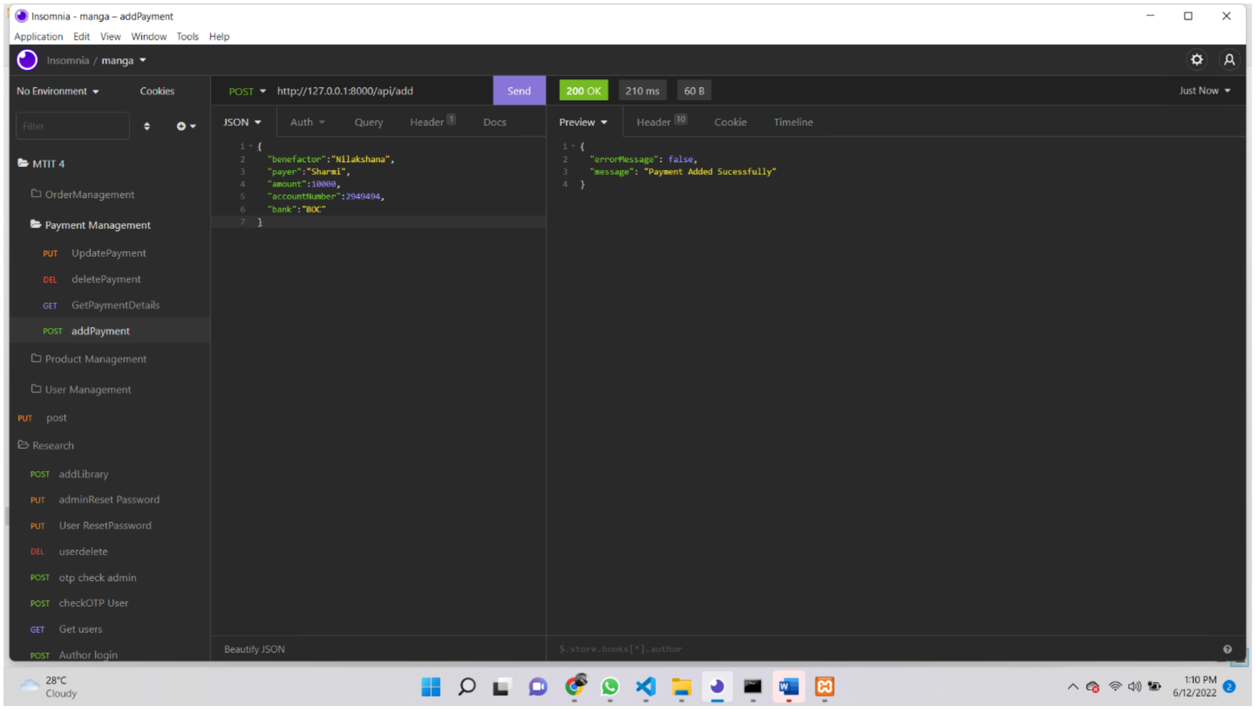


GET http://127.0.0.1:8000/api/getPayment

200 OK 173 ms 592 B

```
[
  {
    "id": 2,
    "benefactor": "Sharmilan SK",
    "payer": "Kithusshand",
    "amount": 250000,
    "accountNumber": 990361070,
    "bank": "CommercialBank",
    "created_at": "2022-06-04T21:37:26.000000Z",
    "updated_at": "2022-06-04T21:38:34.000000Z"
  },
  {
    "id": 3,
    "benefactor": "Nilakshana",
    "payer": "Sharmi",
    "amount": 10000,
    "accountNumber": 2949494,
    "bank": "BOC",
    "created_at": "2022-06-04T21:40:07.000000Z",
    "updated_at": "2022-06-04T21:40:07.000000Z"
  },
  {
    "id": 4,
    "benefactor": "Nilakshana",
    "payer": "Sharmi",
    "amount": 10000,
    "accountNumber": 2949494,
    "bank": "BOC",
    "created_at": "2022-06-12T07:39:58.000000Z",
    "updated_at": "2022-06-12T07:39:58.000000Z"
  }
]
```

API for register Payment to the database (POST Request)



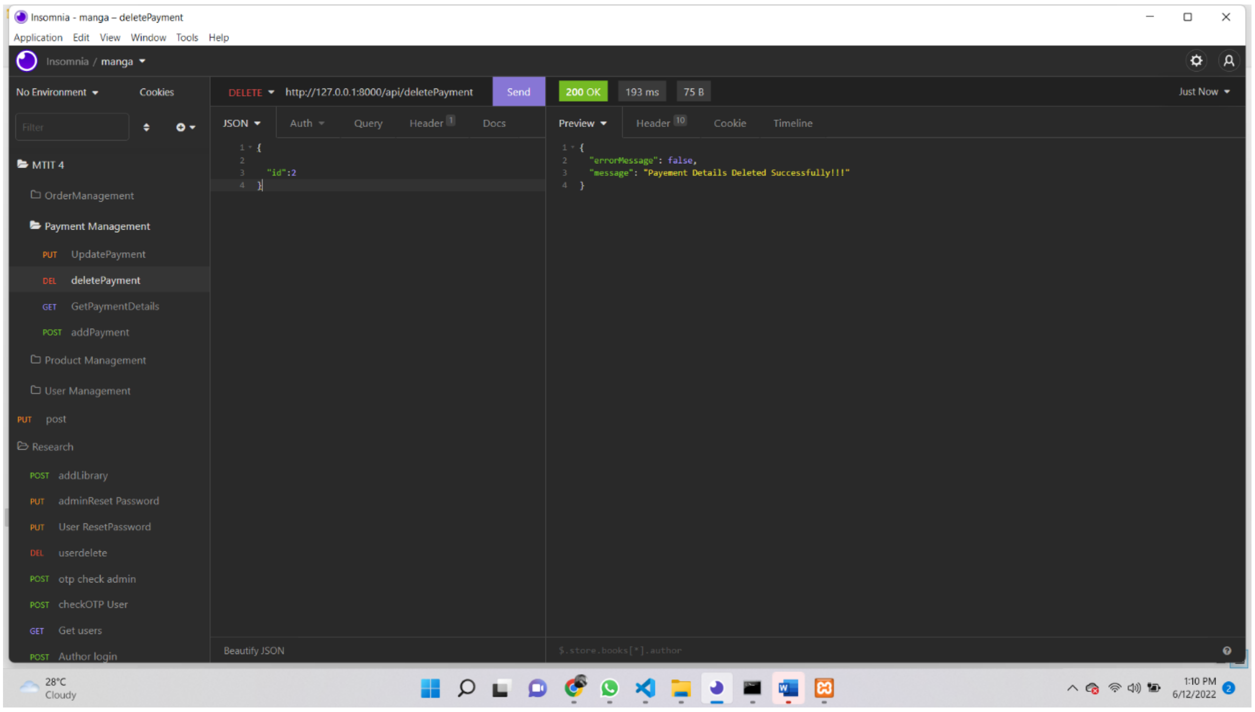
POST http://127.0.0.1:8000/api/add

```
{
  "benefactor":"Nilakshana",
  "payer":"Sharmi",
  "amount":10000,
  "accountNumber":2949494,
  "bank":"BOC"
}
```

200 OK 210 ms 60 B

```
{
  "errorMessage": false,
  "message": "Payment Added Sucessfully"
}
```

API for Delete Payment to the database (DELETE Request)



DELETE http://127.0.0.1:8000/api/deletePayment

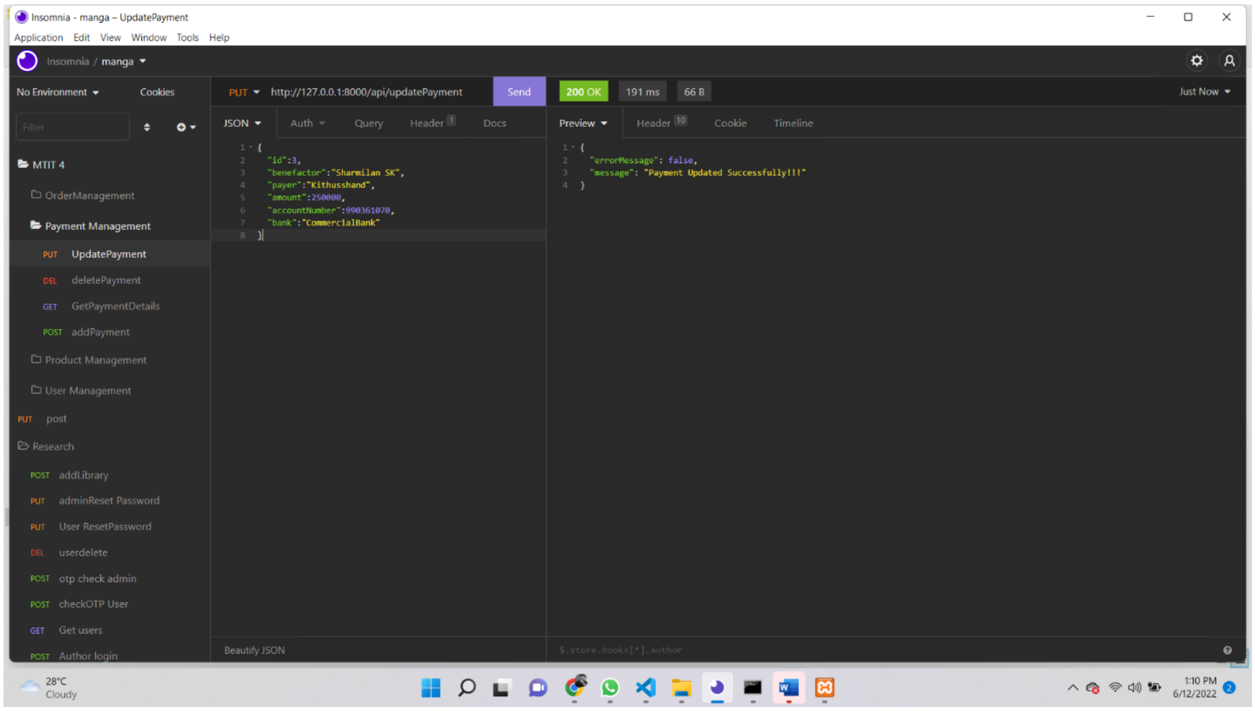
200 OK 193 ms 75 B

```
{

  "id":2
}
```

```
{
  "errorMessage": false,
  "message": "Payement Details Deleted Successfully!!!"
}
```

API for Update Payment to the database (PUT Request)



PUT http://127.0.0.1:8000/api/updatePayment

200 OK 191 ms 66 B

```
{
  "id":3,
  "benefactor":"Sharmilan SK",
  "payer":"Kithusshand",
  "amount":250000,
  "accountNumber":990361070,
  "bank":"CommercialBank"
}
```

```
{
  "errorMessage": false,
  "message": "Payment Updated Successfully!!!"
}
```

Student IT Number: IT19241760 **Web Service: Product Management**

//Read

API for get all the Product details in the database (GET Request)

URL: http://127.0.0.1:8000/api/getProduct

Request: getProduct

Responce: Product_ID, Product Name , Product Type, Price Quantity

//Insert

API for register Product to the database (POST Request)

URL: http://127.0.0.1:8000/api/addProduct

Request: POST **api/addProduct**

Media: JSON

```
{
        "productID":"P005",
        "productName":"running shoe",
        "productType":"Shoes",
        "price":3000,
        "Quantity":25
}
```

Response:

```
                {
                        "errorMessage": false,
                        "message": "Product Added Sucessfully"
                }
```

//Delete

API for Delete Product to the database (DELETE Request)

URL: http://127.0.0.1:8000/api/deleteProduct

Request: DELETE **api/deleteProduct**

Media: Application JSON

```
{
        "id":5
}
```

Response:

```
 {
        "errorMessage": false,
        "message": "Product Deleted Successfully!!!"
}
```

//Update

API for Update Product to the database (PUT Request)

URL: http://127.0.0.1:8000/api/update

Request: PUT a**pi/update**

Media: Application JSON

```
{
        "productID":"P005",
        "productName":"Hoodies full sleeve",
        "productType":"T shirt",
        "price":1250,
        "Quantity":75,
        "id":6
}
```

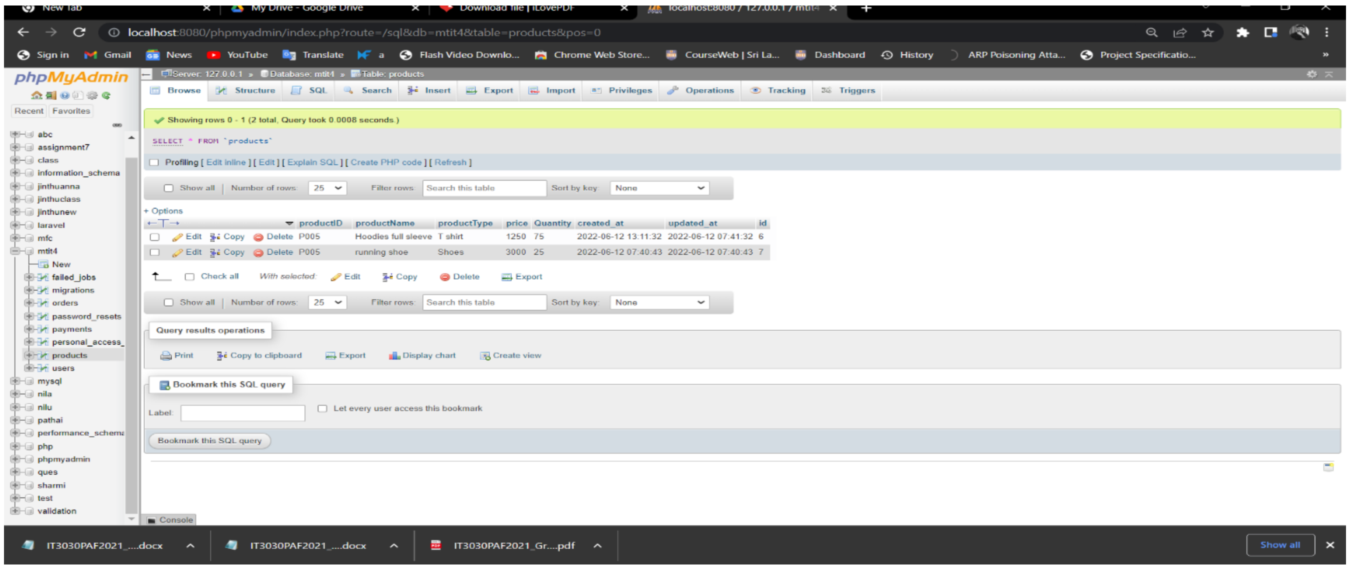
```
Response: {
        "errorMessage": false,
        "message": "Prtoduct Updated Successfully!!!"
}
```

Testing Methodology and Results

Test ID	Test Description	Test Inputs	Expected Output	Actual Output	Result (pass/fail)
01	Inserting Data	"productID":"P005", "productName":"runningshoe", "productType":"Shoes", "price":3000, "Quantity":25	Inserted successfully	Inserted successfully	Pass
02	Updating Data	"productID":"P005", "productName":"Hoodies fullsleeve", "productType":"T shirt", "price":1250, "Quantity":75, "id":6	Updated successfully	Updated successfully	Pass
03	Deleting Data	id= 1	Deleted successfully	Deleted successfully	Pass

Screenshots

Database



localhost:8080/phpmyadmin/index.php?route=/sql&db=mtit4&table=products&pos=0

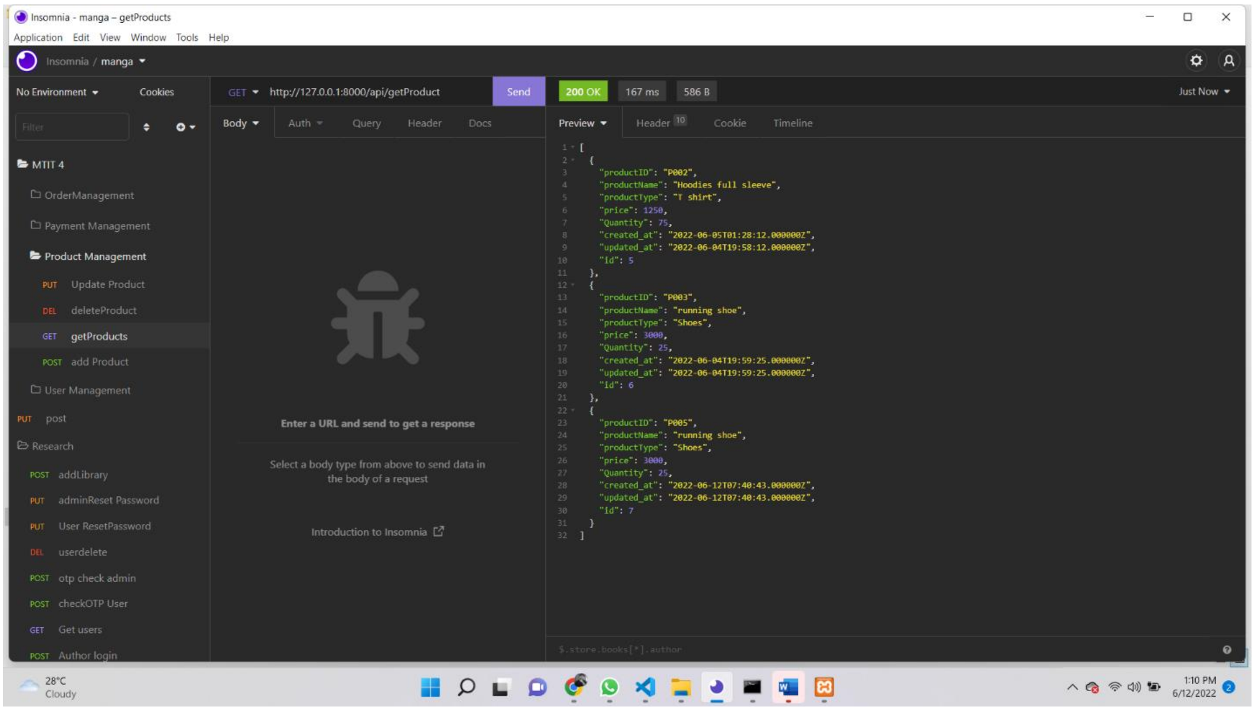
Server: 127.0.0.1 » Database: mtit4 » Table: products

Showing rows 0 - 1 (2 total, Query took 0.0008 seconds.)

```
SELECT * FROM `products`
```

productID	productName	productType	price	Quantity	created_at	updated_at	id
P005	Hoodies full sleeve	T shirt	1250	75	2022-06-12 13:11:32	2022-06-12 07:41:32	6
P005	running shoe	Shoes	3000	25	2022-06-12 07:40:43	2022-06-12 07:40:43	7

API for get all the Product details in the database (GET Request)

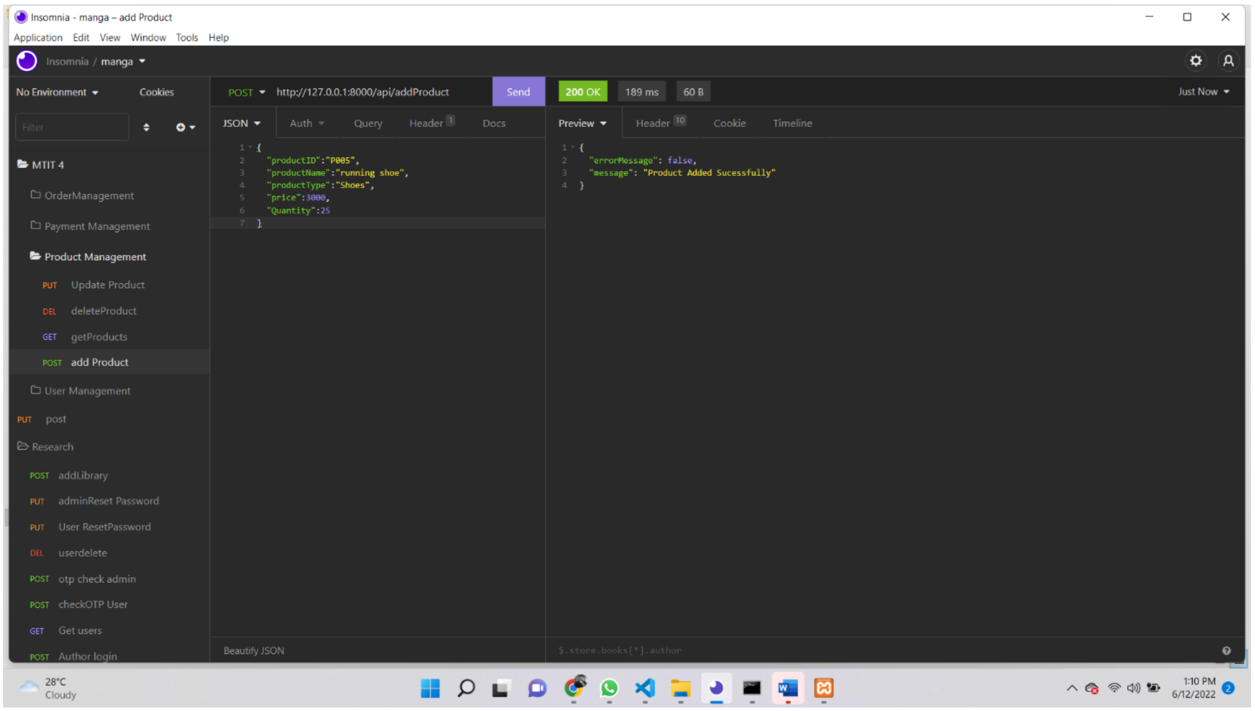


GET http://127.0.0.1:8000/api/getProduct

200 OK 167 ms 586 B

```
[
  {
    "productID": "P002",
    "productName": "Hoodies full sleeve",
    "productType": "T shirt",
    "price": 1250,
    "Quantity": 75,
    "created_at": "2022-06-05T01:28:12.000000Z",
    "updated_at": "2022-06-04T19:58:12.000000Z",
    "id": 5
  },
  {
    "productID": "P003",
    "productName": "running shoe",
    "productType": "Shoes",
    "price": 3000,
    "Quantity": 25,
    "created_at": "2022-06-04T19:59:25.000000Z",
    "updated_at": "2022-06-04T19:59:25.000000Z",
    "id": 6
  },
  {
    "productID": "P005",
    "productName": "running shoe",
    "productType": "Shoes",
    "price": 3000,
    "Quantity": 25,
    "created_at": "2022-06-12T07:40:43.000000Z",
    "updated_at": "2022-06-12T07:40:43.000000Z",
    "id": 7
  }
]
```

API for register Product to the database (POST Request)



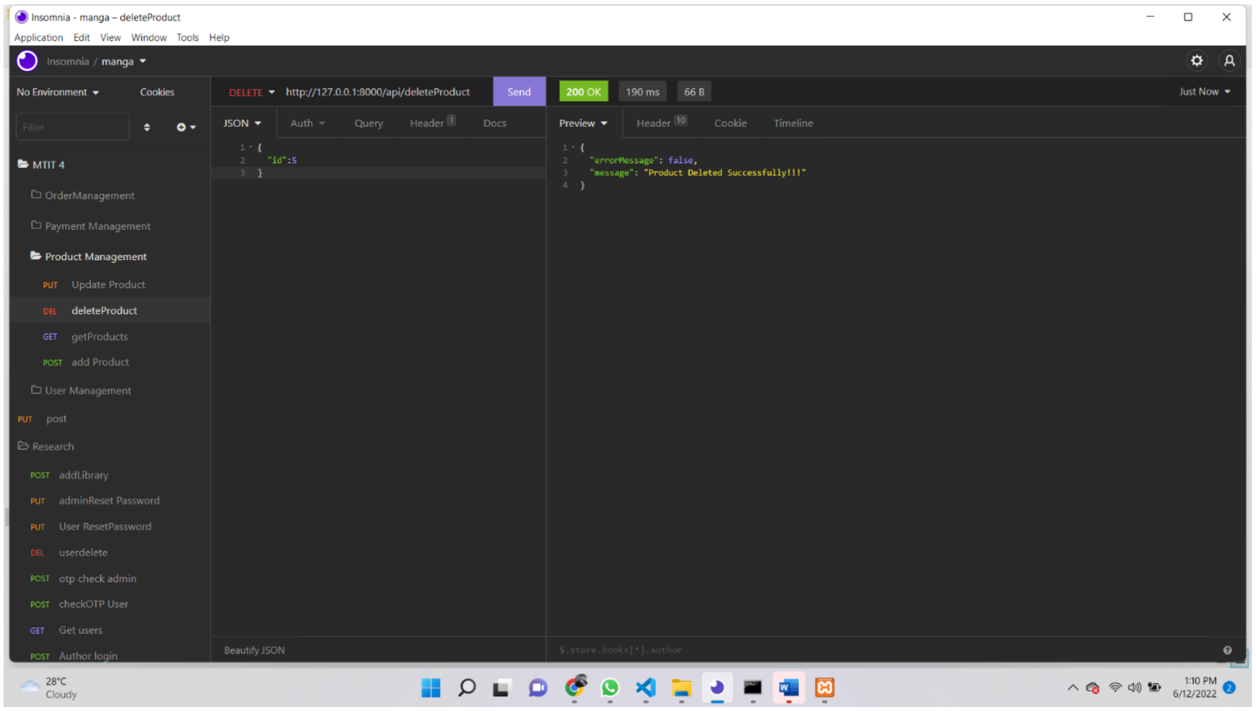
POST http://127.0.0.1:8000/api/addProduct

200 OK 189 ms 60 B

```
{
  "productID":"P005",
  "productName":"running shoe",
  "productType":"Shoes",
  "price":3000,
  "Quantity":25
}
```

```
{
  "errorMessage": false,
  "message": "Product Added Sucessfully"
}
```

API for Delete Product to the database (DELETE Request)



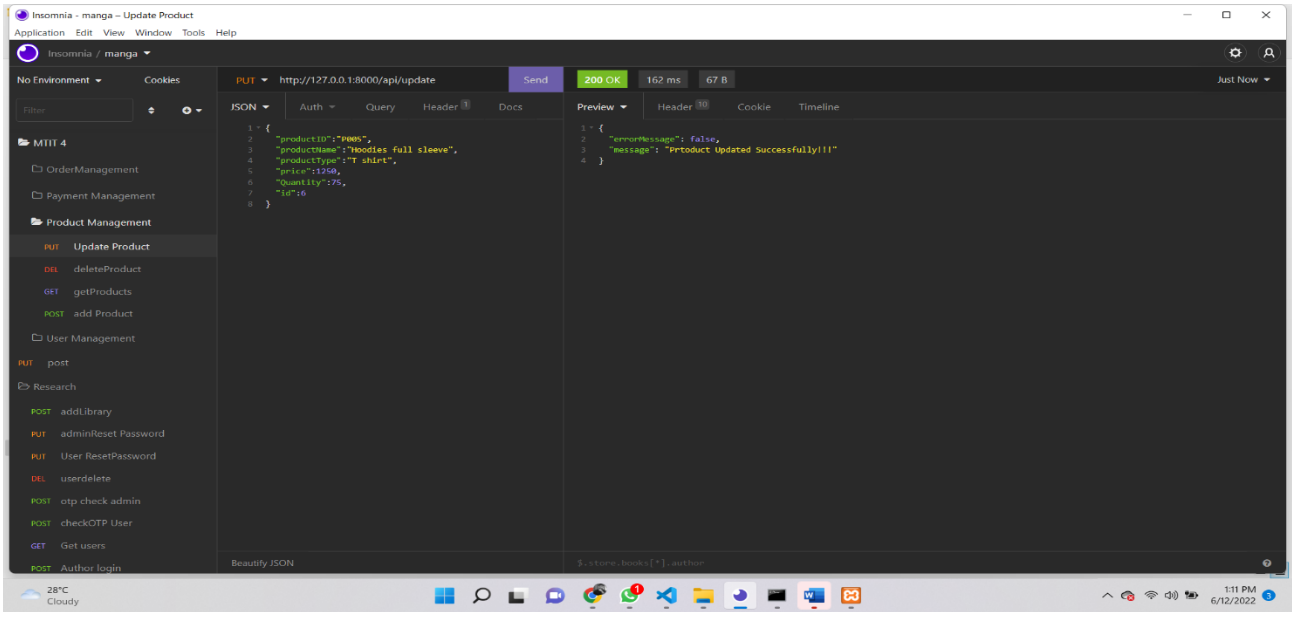
DELETE http://127.0.0.1:8000/api/deleteProduct

200 OK 190 ms 66 B

```
{
  "id":5
}
```

```
{
  "errorMessage": false,
  "message": "Product Deleted Successfully!!!"
}
```

API for Update Product to the database (PUT Request)



PUT http://127.0.0.1:8000/api/update

200 OK 162 ms 67 B

```
{
  "productID":"P005",
  "productName":"Hoodies full sleeve",
  "productType":"T shirt",
  "price":1250,
  "Quantity":75,
  "id":6
}
```

```
{
  "errorMessage": false,
  "message": "Prtoduct Updated Successfully!!!"
}
```

Student IT Number: IT19029832 **Web Service: Order Management**

//Read

API for get all the Product details in the database (GET Request)

URL: http://127.0.0.1:8000/api/getOrders

Request: getOrders

Responce: productName , orderAddress , phoneNumber, email, totalCost, customer_name

//Insert

API for register Order to the database (POST Request)

URL: http://127.0.0.1:8000/api/addOrder

Media: JSON

```
{
        "customer_name":"ram",
        "productName":"shoe",
        "orderAddress":"dehiwala",
        "phoneNumber":776165366,
        "email":"ram@gmail.com",
        "totalCost":3500
}
```

Response:

```
        {
                "errorMessage": false,
                "message": "Order Placed Sucessfully"
        }
```

//Delete

API for Delete Orders to the database (DELETE Request)

URL: http://127.0.0.1:8000/api/deleteOrder

Request: DELETE

Media: Application JSON

```
{
        "id":5
}
```

Response:

```
 {
        "errorMessage": false,
        "message": "Product Deleted Successfully!!!"
}
```

//Update

API for Update Order to the database (PUT Request)

URL: http://127.0.0.1:8000/api/updateOrder

Request: PUT

Media: Application JSON

```
{
        "id":4,
        "customer_name":"Kithu",
        "productName":"shirt",
        "orderAddress":"batti",
        "phoneNumber":776165366,
        "email":"kithu@gmail.com",
        "totalCost":1800
}
```

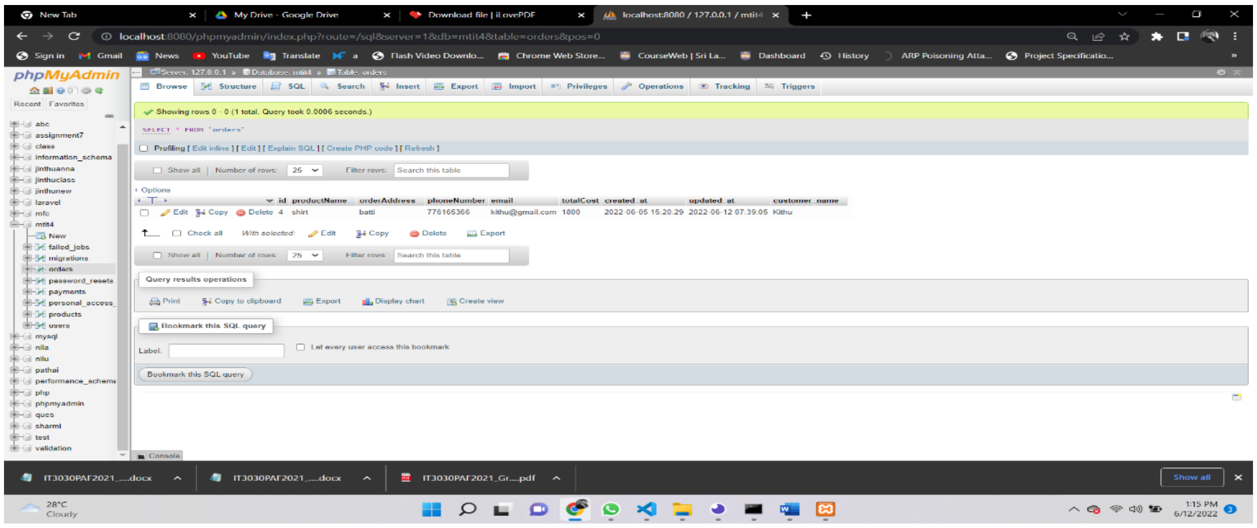
```
Response: {
        "errorMessage": false,
        "message": "Order Updated Successfully!!!"
}
```

Testing Methodology and Results

Test ID	Test Description	Test Inputs	Expected Output	Actual Output	Result (pass/fail)
01	Inserting Data	"customer_name":"ram", "productName":"shoe", "orderAddress":"dehiwala", "phoneNumber":776165366, "email":"ram@gmail.com", "totalCost":3500	Inserted successfully	Inserted successfully	Pass
02	Updating Data	"id":4, "customer_name":"Kithu", "productName":"shirt", "orderAddress":"batti", "phoneNumber":776165366, "email":"kithu@gmail.com", "totalCost":1800	Updated successfully	Updated successfully	Pass
03	Deleting Data	id= 1	Deleted successfully	Deleted successfully	Pass

Screenshots

Database



localhost:8080/phpmyadmin/index.php?route=/sql&server=1&db=mtit4&table=orders&pos=0

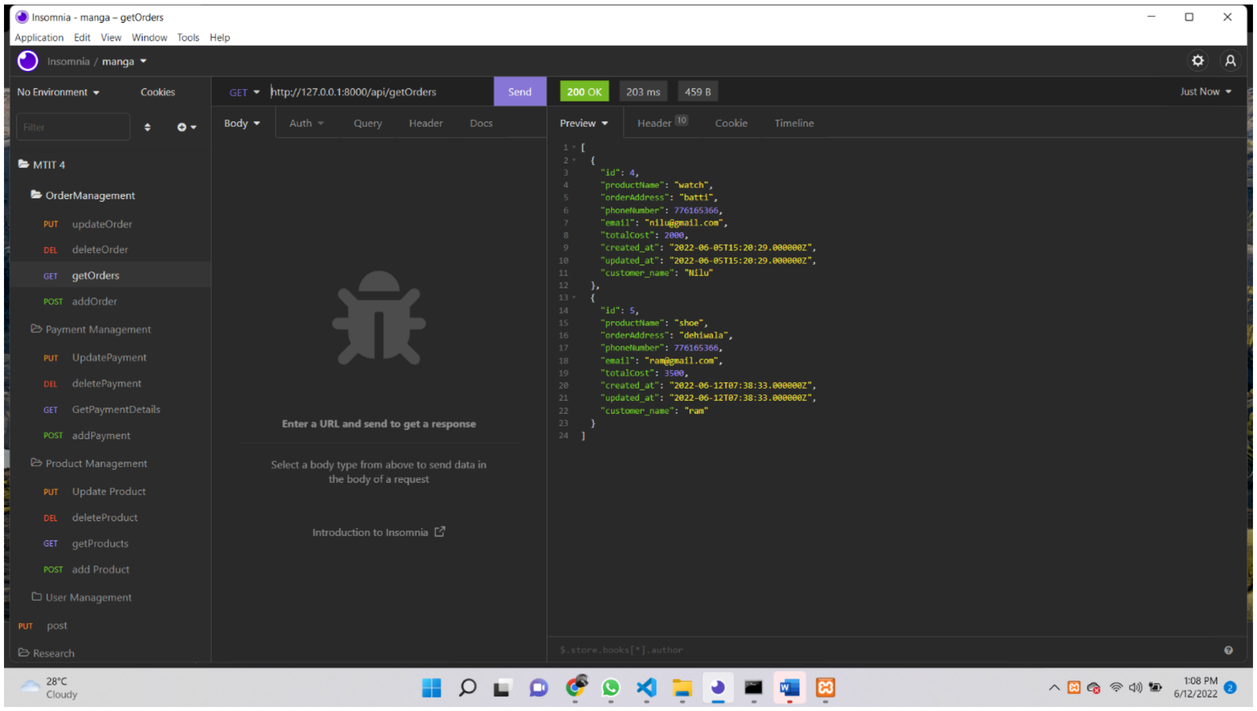
Server: 127.0.0.1 » Database: mtit4 » Table: orders

Showing rows 0 - 0 (1 total, Query took 0.0006 seconds.)

```
SELECT * FROM `orders`
```

id	productName	orderAddress	phoneNumber	email	totalCost	created_at	updated_at	customer_name
4	shirt	batti	776165366	kithu@gmail.com	1800	2022-06-05 15:20:29	2022-06-12 07:39:05	Kithu

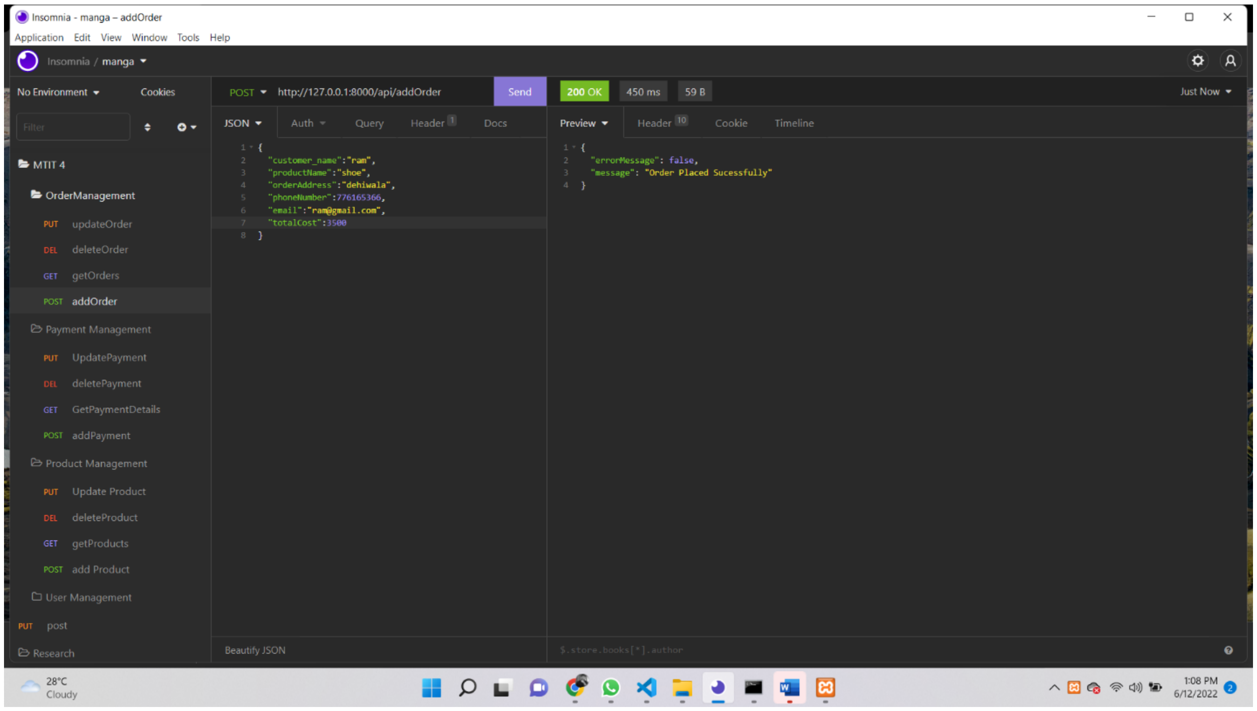
API for get all the Order details in the database (GET Request)



http://127.0.0.1:8000/api/getOrders

```
[
  {
    "id": 4,
    "productName": "watch",
    "orderAddress": "batti",
    "phoneNumber": 776165366,
    "email": "nilu@gmail.com",
    "totalCost": 2000,
    "created_at": "2022-06-05T15:20:29.000000Z",
    "updated_at": "2022-06-05T15:20:29.000000Z",
    "customer_name": "Nilu"
  },
  {
    "id": 5,
    "productName": "shoe",
    "orderAddress": "dehiwala",
    "phoneNumber": 776165366,
    "email": "ram@gmail.com",
    "totalCost": 3500,
    "created_at": "2022-06-12T07:38:33.000000Z",
    "updated_at": "2022-06-12T07:38:33.000000Z",
    "customer_name": "ram"
  }
]
```

API for register Order to the database (POST Request)

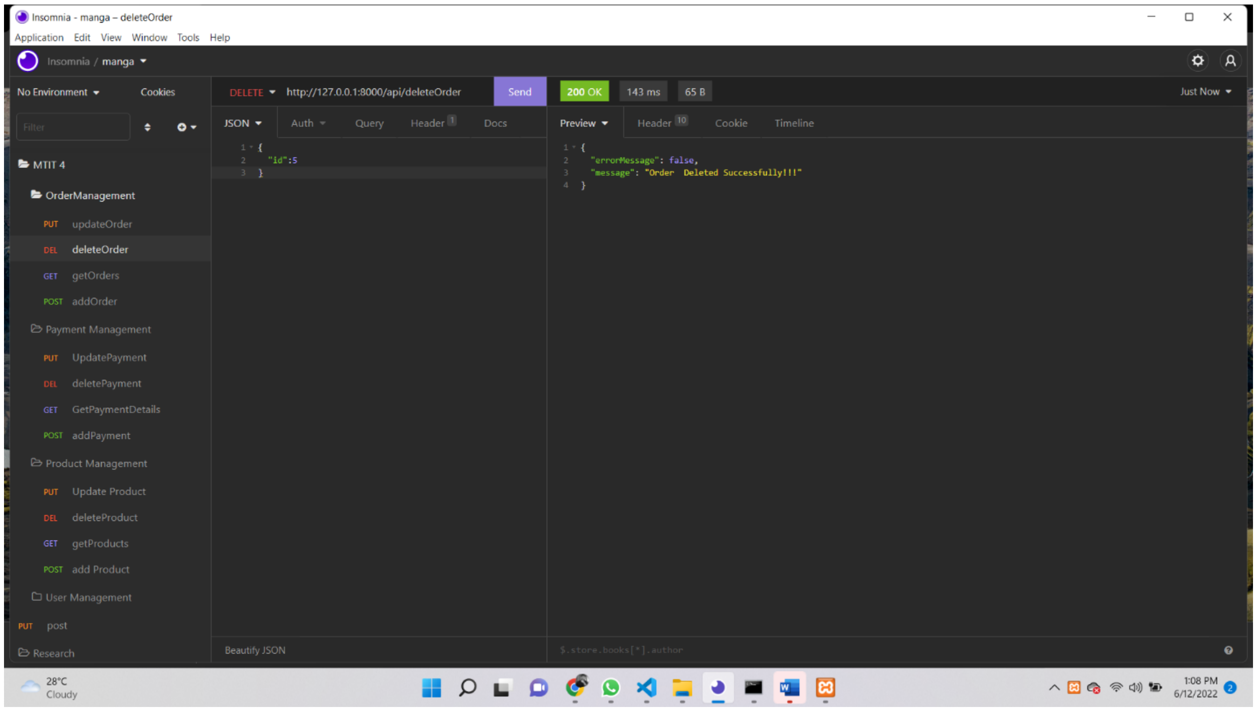


http://127.0.0.1:8000/api/addOrder

```
{
  "customer_name":"ram",
  "productName":"shoe",
  "orderAddress":"dehiwala",
  "phoneNumber":776165366,
  "email":"ram@gmail.com",
  "totalCost":3500
}
```

```
{
  "errorMessage": false,
  "message": "Order Placed Sucessfully"
}
```

API for Delete Orde to the database (DELETE Request)



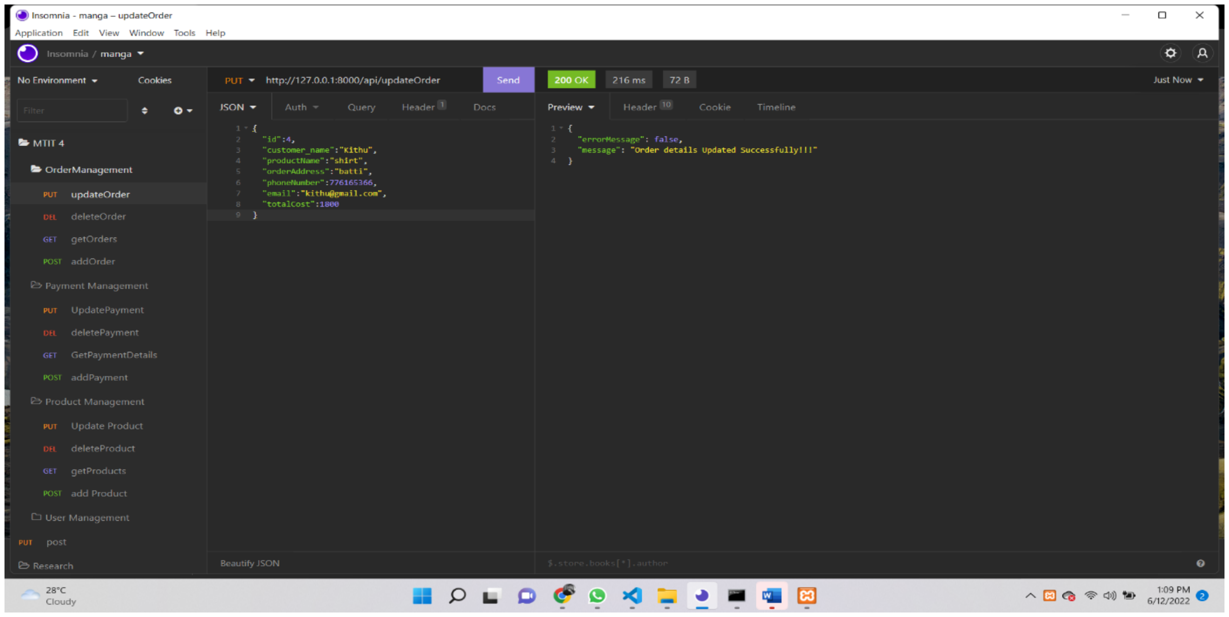
DELETE http://127.0.0.1:8000/api/deleteOrder

200 OK 143 ms 65 B

```
{
  "id":5
}
```

```
{
  "errorMessage": false,
  "message": "Order  Deleted Successfully!!!"
}
```

API for Update Order to the database (PUT Request)



PUT http://127.0.0.1:8000/api/updateOrder

200 OK 216 ms 72 B

```
{
  "id":4,
  "customer_name":"Kithu",
  "productName":"shirt",
  "orderAddress":"batti",
  "phoneNumber":776165366,
  "email":"kithu@gmail.com",
  "totalCost":1800
}
```

```
{
  "errorMessage": false,
  "message": "Order details Updated Successfully!!!"
}
```

Student IT Number: IT19119618 **Web Service: User Management**

//Read

API for get all the User details in the database (GET Request)

URL: http://127.0.0.1:8000/api/get

Request: GET api/get

Responce: User's Details like name , email,

address

//Insert

API for register user to the database (POST Request)

URL: http://127.0.0.1:8000/api/addUser

Request: POST api/addUser

Media: Application JSON

```
{
        "name":"Kithusshand",
        "email":"kithusshand0@gmail.com",
        "password":"Kithusshand@99",
        "phone":"0772502342"
}
```

Response:

```
{
        "errorMessage": false,
        "message": "User Added Sucessfully"
}
```

//Delete

API for Delete User to the database (DELETE Request)

URL: http://127.0.0.1:8000/api/deleteUser

Request: DELETE api/deleteUser Media:

Application JSON

```
{
        "id":8
}
```

Response:

```
{
        "errorMessage": false,
        "message": "User Deleted Successfully!!!"
}
```

//Update

API for Update User to the database (PUT Request)

URL: http://127.0.0.1:8000/api/updateUser

Request: PUT api/updateUser

Media: Application JSON

```
{
        "id":8,
        "name":"sharmilan",
        "email":"kithu0@gmail.com",
        "password":"Kithusshand@99",
        "phone":"0772502342"
}
```

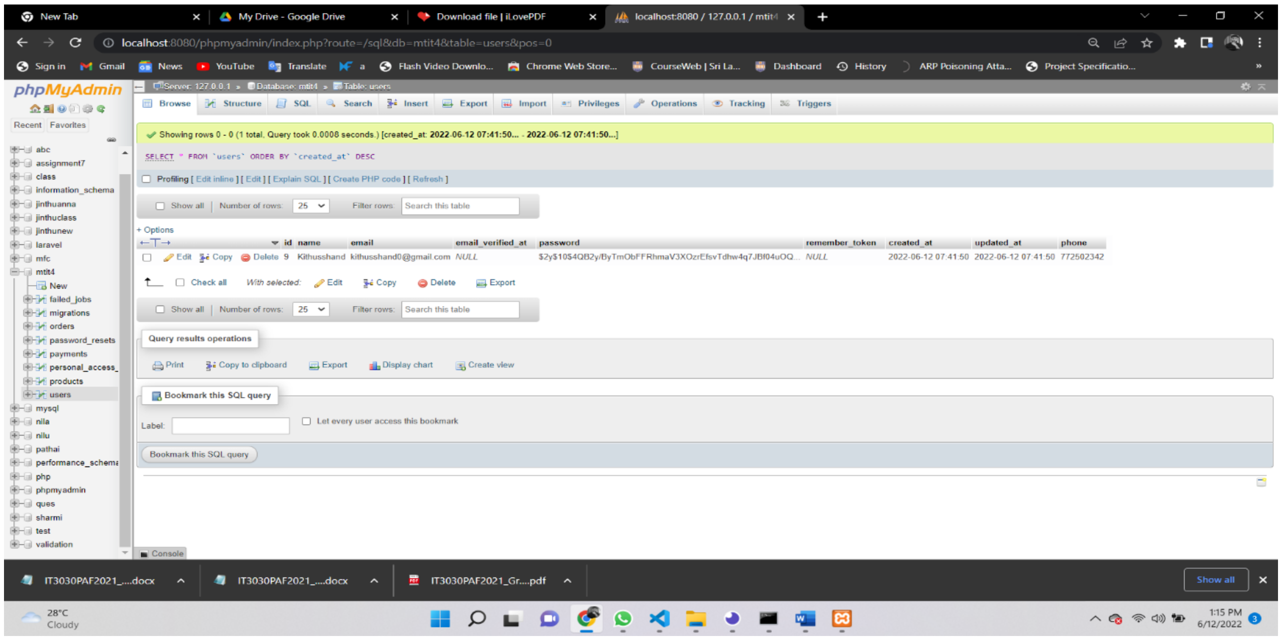
```
Response: {
        "errorMessage": false,
        "message": "User Updated Successfully!!!"
```

Testing Methodology and Results

Test ID	Test Description	Test Inputs	Expected Output	Actual Output	Result (pass/fail)
01	Inserting Data	"name":"Kithusshand", "email":"kithusshand0@gmail.com", "password":"Kithusshand@99", "phone":"0772502342"	Inserted successfully	Inserted successfully	Pass
02	Updating Data	"id":8, "name":"sharmilan", "email":"kithu0@gmail.com", "password":"Kithusshand@99", "phone":"0772502342"	Updated successfully	Updated successfully	Pass
03	Deleting Data	id = "1"	Deleted successfully	Deleted successfully	Pass

Screenshots

Database



localhost:8080/phpmyadmin/index.php?route=/sql&db=mtit4&table=users&pos=0

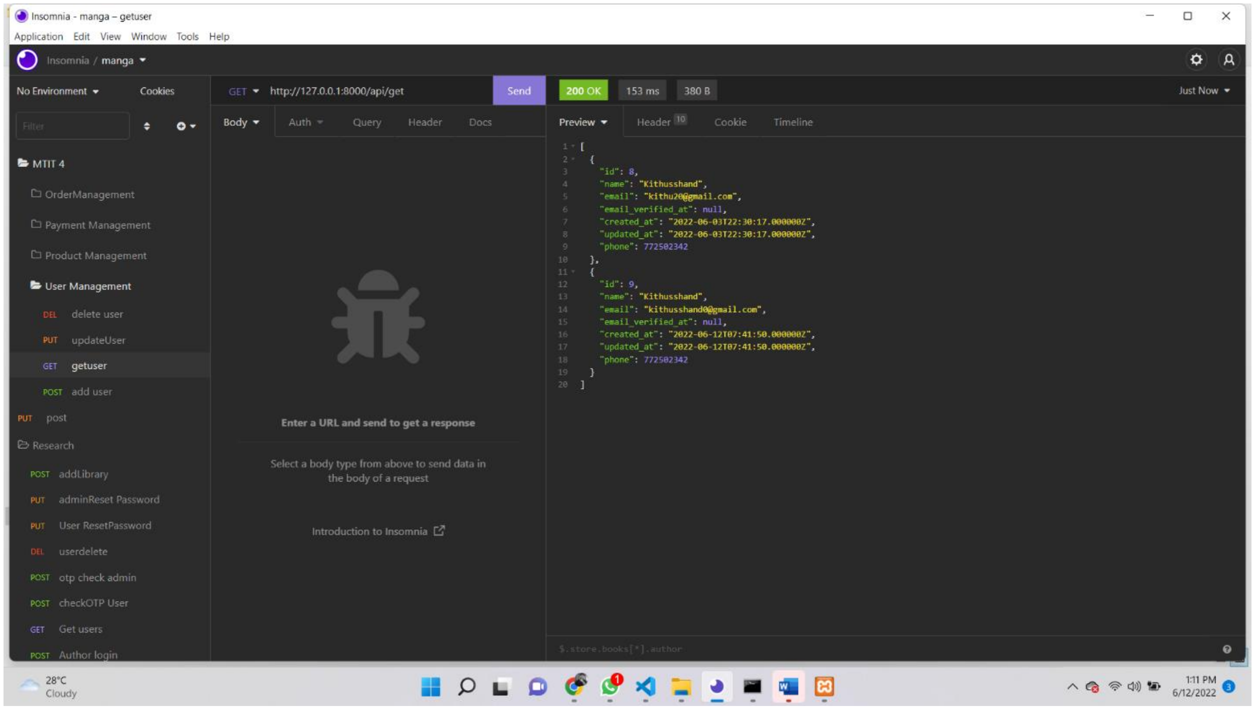
Server: 127.0.0.1 » Database: mtit4 » Table: users

Showing rows 0 - 0 (1 total, Query took 0.0008 seconds.) [created_at: 2022-06-12 07:41:50... - 2022-06-12 07:41:50...]

```
SELECT * FROM `users` ORDER BY `created_at` DESC
```

id	name	email	email_verified_at	password	remember_token	created_at	updated_at	phone
9	Kithusshand	kithusshand0@gmail.com	NULL	$2y$10$4QB2y/ByTmObFFRhmaV3XOzrEfsvTdhw4q7JBf04uOQ...	NULL	2022-06-12 07:41:50	2022-06-12 07:41:50	772502342

API for get all the User details in the (GET Request)

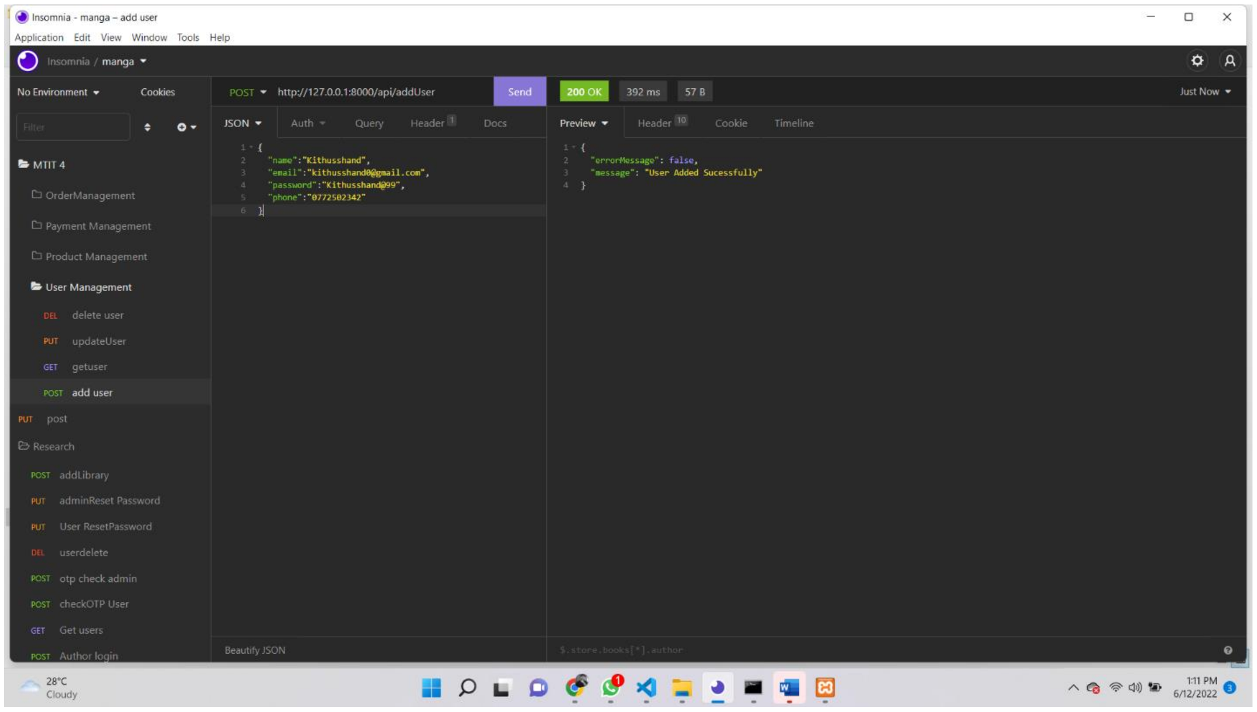


GET http://127.0.0.1:8000/api/get

200 OK 153 ms 380 B

```
[
  {
    "id": 8,
    "name": "Kithusshand",
    "email": "kithu20@gmail.com",
    "email_verified_at": null,
    "created_at": "2022-06-03T22:30:17.000000Z",
    "updated_at": "2022-06-03T22:30:17.000000Z",
    "phone": 772502342
  },
  {
    "id": 9,
    "name": "Kithusshand",
    "email": "kithusshand0@gmail.com",
    "email_verified_at": null,
    "created_at": "2022-06-12T07:41:50.000000Z",
    "updated_at": "2022-06-12T07:41:50.000000Z",
    "phone": 772502342
  }
]
```

API for register User to the database (POST Request)



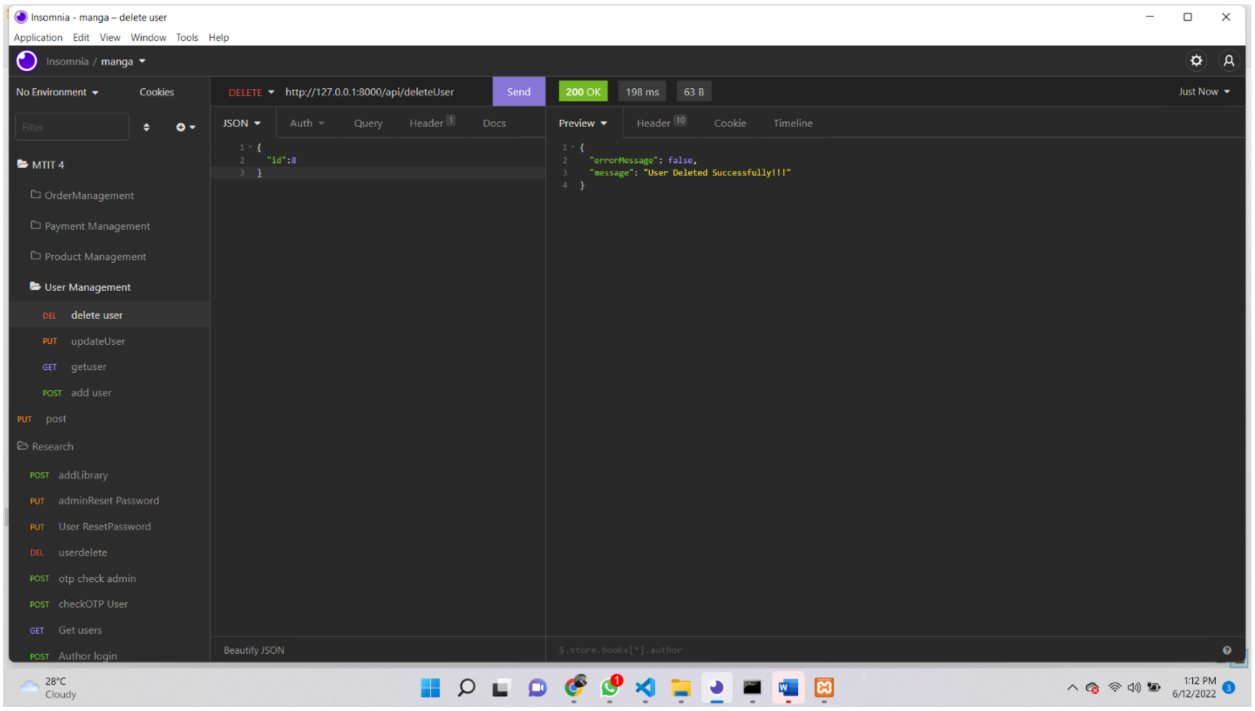
POST http://127.0.0.1:8000/api/addUser

```
{
  "name":"Kithusshand",
  "email":"kithusshand0@gmail.com",
  "password":"Kithusshand@99",
  "phone":"0772502342"
}
```

200 OK 392 ms 57 B

```
{
  "errorMessage": false,
  "message": "User Added Sucessfully"
}
```

API for Delete User to the database (DELETE Request)



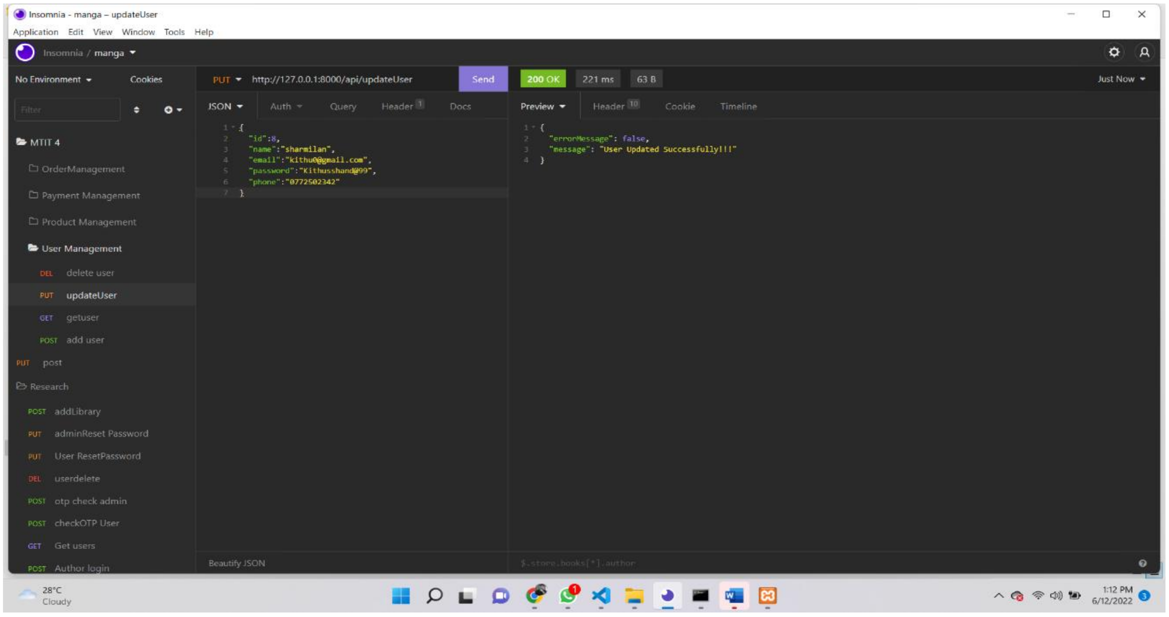
DELETE http://127.0.0.1:8000/api/deleteUser Send

200 OK 198 ms 63 B

```
{
  "id":8
}
```

```
{
  "errorMessage": false,
  "message": "User Deleted Successfully!!!"
}
```

API for Update User to the database (PUT Request)



PUT http://127.0.0.1:8000/api/updateUser Send

200 OK 221 ms 63 B

```
{
  "id":8,
  "name":"sharmilan",
  "email":"kithu0@gmail.com",
  "password":"Kithusshand@99",
  "phone":"0772502342"
}
```

```
{
  "errorMessage": false,
  "message": "User Updated Successfully!!!"
}
```