**Project: Full Ecommerce API with authentication**

**Dependencies: node, Express, nodemon, dotenv, jsonwebtoken, mongoose**

**Instruction:**

**1. please install all dependencies and Pull all files including “.env” which includes mongo URI and JWT secrect key to run the api locally.**

**2. after login please add the "accessToken" as “token” key in request header for all request.**

**Registration and Login**

**User registration:**

Endpoint: <http://localhost:3000/api/auth/register>

Req body:

{

    "username":"username",

    "email": "user@user.com",

    "password":"password"

}

Req type: POST

Res: will receive a response with all user details including password for testing case only can be changed later on.

{

    "\_id": "64e0c44d3f7117e46669e1bb",

    "username": "sayan Guha",

    "email": "sayan@sayan.com",

"password": "cGFzc3dvcmQ=",

    "cart password": [],

    "accessToken": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6IjY0ZTBjNDRkM2Y3MTE3ZTQ2NjY5ZTFiYiIsImlhdCI6MTY5MjQ1MjM3MCwiZXhwIjoxNjkyNzExNTcwfQ.oh4QwZqy9nCePpZh\_A9qnJmviRRynrRM\_TRcdESSW\_Y"

}

**User login**

Endpoint: <http://localhost:3000/api/auth/login>

Req body:

{

    "email":"sayan@sayan.com",

    "password":"password"

}

Req type: POST

Res: will receive user details without password along with id and access token which needs to be saved.

{

    "\_id": "64e0c44d3f7117e46669e1bb",

    "username": "sayan Guha",

    "email": "sayan@sayan.com",

    "cart": [],

"accessToken": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6IjY0ZTBjNDRkM2Y3MTE3ZTQ2NjY5ZTFiYiIsImlhdCI6MTY5MjQ1MjM3MCwiZXhwIjoxNjkyNzExNTcwfQ.oh4QwZqy9nCePpZh\_A9qnJmviRRynrRM\_TRcdESSW\_Y"

}

**User Cart**

For all this operation user needs to be authenticated using given access-token during login time

**Common Req Header: {token: access-token}**

**Adding item to Cart**

Endpoint: <http://localhost:3000/api/user/cart/:id?new=true>

Req body:

{

    "productId":"123456789",

    "quantity":"10"

}

Req params: id=user ID

Req query: new=true/false

Req type: POST

Resp: will receive a details of user without password along with cart info

{

    "\_id": "64e0c44d3f7117e46669e1bb",

    "username": "sayan Guha",

    "email": "sayan@sayan.com",

    "cart": [

        {

            "productId": "123456789",

            "quantity": 10,

            "\_id": "64e0c9728e736468ff7aecd7"

        }

    ],

}

**Removing item form cart**

Endpoint: <http://localhost:3000/api/user/cart/:id?remove=true>

Req body:

{

    "productId":"123456789"

}

Req params: id=user ID

Req query: remove=true/false

Req type: POST

Resp: will receive the updated cart without the item

{

    "\_id": "64e0c44d3f7117e46669e1bb",

    "username": "sayan Guha",

    "email": "sayan@sayan.com",

    "cart": [],

}

**Updating product quantity in cart**

Endpoint: <http://localhost:3000/api/user/cart/:id?update=true>

Req body

{

    "productId":"1234567890123456789",

    "quantity":1

}

Req params: id=user ID

Req query: update=true/false

Req type: POST

Resp: will receive cart with updated quantity

{

    "\_id": "64e0c44d3f7117e46669e1bb",

    "username": "sayan Guha",

    "email": "sayan@sayan.com",

    "cart": [

        {

            "productId": "123456789",

            "quantity": 10,

            "\_id": "64e0cb957287ebe324750685"

        },

        {

            "productId": "1234567890123456789",

            "quantity": 1,

            "\_id": "64e0cb9d7287ebe324750688"

        }

    ],

}

**Getting all the items currently in cart**

Endpoint: <http://localhost:3000/api/user/cart/:id>

Req params: id=user ID

Req type: GET

Res: will receive only the cart info

{

    "cart": [

        {

            "productId": "123456789",

            "quantity": 10,

            "\_id": "64e0cb957287ebe324750685"

        },

        {

            "productId": "1234567890123456789",

            "quantity": 1,

            "\_id": "64e0cb9d7287ebe324750688"

        }

    ],

}

**Product**

**Adding product**

Only via /admin route one can add product, have not implemented the authentication

Endpoint: <http://localhost:3000/api/products/admin?new=true>

Req body

{

    "title":"product15",

    "desc":"description15",

    "price":15,

    "categories":["veg"]

}

Note: currently there are two categories “veg , “non-veg” &”chicken

Req query:new=true/false

Req type: POST

Resp: will receive cart with added product

{

    "\_id": "64dfc81200bcac402d0ddb85",

    "name": "veg",

    "count": 3,

}

**Getting all products**

Endpoint: <http://localhost:3000/api/products/>

Req type: GET

Resp: will receive all product listing

[

    {

        "\_id": "64df9166e676e41ab12dfc1e",

        "title": "product1",

        "desc": "description",

        "price": 69,

        "categories": [

            "non-veg",

            "chicken"

        ]

    },

    {

        "\_id": "64df91aa95c626cad62cb890",

        "title": "product2",

        "desc": "description2",

        "price": 695,

        "categories": [

            "veg",

            "not-chicken"

        ]

    }

]

**Getting product by category**

Endpoint: http://localhost:3000/api/products?category=veg

Req type: GET

Req query : category=veg/non-veg/chicken (currently)

Resp: will receive all product sorted by category

[

    {

        "\_id": "64df9166e676e41ab12dfc1e",

        "title": "product1",

        "desc": "description",

        "price": 69,

        "categories": [

            "non-veg",

            "chicken"

        ]

    },

    {

        "\_id": "64df94cfc87f191315343444",

        "title": "product3",

        "desc": "description3",

        "price": 6955,

        "categories": [

            "non-veg"

        ]

    },

    {

        "\_id": "64dfcbb6d83f671cd3af0eac",

        "title": "title1",

        "desc": "description",

        "price": 555,

        "categories": [

            "veg",

            "non-veg"

        ]

    }

]

**Getting product by ID**

Endpoint: http://localhost:3000/api/products/:id

Req type: GET

Req params : id=product ID

Resp: will receive product having the IDy

[

    {

        "\_id": "64df9166e676e41ab12dfc1e",

        "title": "product1",

        "desc": "description",

        "price": 69,

        "categories": [

            "non-veg",

            "chicken"

        ]

    }

]

**Order**

user verification required

**Common Req Header: {token: access-token}**

**Making a order**

Endpoint; <http://localhost:3000/api/orders?new=true>

Req.body: (will require actual userID)

{

    "userId":"64e0c44d3f7117e46669e1bb",

    "products":[{"productId": "123456789"}],

    "amount":15,

    "address":"earth 1/7 asia, India"

}

Req query: new=true/false

Req.type: POST

Resp: will receive details of current order

{

    "userId": "64e0c44d3f7117e46669e1bb",

    "products": [

        {

            "productId": "123456789",

            "\_id": "64e0d59203b094be506ae073"

        }

    ],

    "amount": 15,

    "address": "earth 1/7 asia, India",

    "Status": "pending",

    "\_id": "64e0d59203b094be506ae072",

}

**Getting Order history by a specific user**

Endpoint; [http://localhost:3000/api/orders](http://localhost:3000/api/orders?new=true)

Req.type: GET

Resp: will receive details of all orders madde

{

    "userId": "64e0c44d3f7117e46669e1bb",

    "products": [

        {

            "productId": "123456789",

            "\_id": "64e0d59203b094be506ae073"

        }

    ],

    "amount": 15,

    "address": "earth 1/7 asia, India",

    "Status": "pending",

    "\_id": "64e0d59203b094be506ae072",

}

**Getting Order by ID by specific user**

Endpoint; <http://localhost:3000/api/orders/>

Req params: ID=order Id

Req.type: GET

Resp: will receive specific orders

{

    "\_id": "64e0d59203b094be506ae072",

    "userId": "64e0c44d3f7117e46669e1bb",

    "products": [

        {

            "productId": "123456789",

            "\_id": "64e0d59203b094be506ae073"

        }

    ],

    "amount": 15,

    "address": "earth 1/7 asia, India",

    "Status": "pending",

}

**Getting all orders made to this shop (admin only, test purpose , authentication not implemented)**

Endpoint: <http://localhost:3000/api/orders/admin>

Req type: GET

Resp: will receive all orders made to this platform

[

    {

        "\_id": "64dfbc25d5f54cd51871d9ac",

        "userId": "12312df232",

        "products": [

            {

                "quantity": 5,

                "\_id": "64dfbc25d5f54cd51871d9ad"

            }

        ],

        "amount": 500,

        "address": "1/2 earth, India",

        "Status": "pending",

    },

    {

        "\_id": "64dfbc5eb8d4c80905738de7",

        "userId": "12312df232",

        "products": [

            {

                "quantity": 5,

                "\_id": "64dfbc5eb8d4c80905738de8"

            },

            {

                "quantity": 2,

                "\_id": "64dfbc5eb8d4c80905738de9"

            }

        ],

        "amount": 50,

        "address": "1/2 earth, India",

        "Status": "pending",

    }]

**Categories**

**Making new categories (admin only authentication not implemented)**

Endpoint: <http://localhost:3000/api/categories/admin?new=true>

Req body:

{

    "name":"chicken"

}

Req query: new=true/false

Req type: POST

Resp: will receive created category

{

    "name": "chicken",

    "count": 0,

    "\_id": "64e0da030911ce38e829b724",

}

**Getting all categories**

Endpoint: [http://localhost:3000/api/categories/](http://localhost:3000/api/categories/admin?new=true)

Req type: GET

Resp: will receive all created category

[

    {

        "\_id": "64dfc81200bcac402d0ddb85",

        "name": "veg",

        "count": 3,

    },

    {

        "\_id": "64dfc82100bcac402d0ddb87",

        "name": "non-veg",

        "count": 1,

    },

    {

        "\_id": "64e0da030911ce38e829b724",

        "name": "chicken",

    }

]

**Getting category by ID**

Endpoint: <http://localhost:3000/api/categories/:id>

Req params: id= category ID

Req type: GET

Resp: will receive category having the ID

{

    "\_id": "64dfc81200bcac402d0ddb85",

    "name": "veg",

    "count": 3,

}