API Document

The **PIZZA DELIVERY API** has five routes i.e /users, /tokens, /menu, /carts and /orders. The use and usage of these routes are shown below.

USERS: /users

The users route is used for creating the individual user and it is used for creating the authentication tokens for accessing the other routes in the api. In the /users route, four methods are there i.e

POST: For creating the individual user and new cart of the user for storing the cart data.

URL: http://localhost:3000/users

HEADERS:

- Content-Type: application/json

BODY: (For Example)

```
{
        "firstname": "john",
        "lastname" : “carter",
        "phone": "1234567890",
        "email": "john@gmail.com",
        "password": "password",
        "streetAddress": "221B, Baker street, London"
}
```

RESPONSE: None

RESPONSE HTTP STATUS CODES:

- **200 - Success**
- **400 - Bad Request**
- **500 - Internal Server Error**

GET: For creating the individual user details. Need to send the email as query string as shown in the URL.

URL: http://localhost:3000/users?email=john@gmail.com

HEADERS:

- Content-Type: application/json
- token : (please see the /tokens route Post method for getting the token)

BODY: None

RESPONSE:

```
{
  "firstname": "john",
  "lastname": "carter",
  "email": "john@gmail.com",
  "phone": "1234567890",
  "streetAddress": "vskp",
  "cart": "221B, Baker street, London",
}
```

RESPONSE HTTP STATUS CODES:

- **200 - Success**
- **400 - Bad Request**
- **403 - Forbidden**
- **404 - Not Found**

PUT: For updating the previously created user. Need to send the email as query string as shown in the URL

URL: **http://localhost:3000/users?email=john@gmail.com**
HEADERS:

- Content-Type: application/json
- token : (please see the /tokens route Post method for getting the token)

BODY: (For Example) Except email we can update any one or all properties in the body**.**

```
{
        "firstname": "john",
        "lastname" : “carter",
        "phone": "1234567890",
        "email": "john@gmail.com",
        "streetAddress": "221B, Baker street, London"
}
```

RESPONSE: None
RESPONSE HTTP STATUS CODES:

- **200 - Success**
- **400 - Bad Request**
- **403 - Forbidden**
- **404 - Not Found**
- **500 - Internal Server Error**

DELETE: For deleting the created user along with user cart. Need to send the email as query string as shown in the URL

URL: **http://localhost:3000/users?email=john@gmail.com**
HEADERS:

- Content-Type: application/json
- token : (please see the /tokens route Post method for getting the token)

BODY: None
RESPONSE: None
RESPONSE HTTP STATUS CODES:

- **200 - Success**
- **400 - Bad Request**
- **403 - Forbidden**
- **404 - Not Found**
- **500 - Internal Server Error**

TOKENS: /tokens

The tokens route is used for creating the authorization token for individual user and it is used for accessing the other routes in the api. In the /tokens route, four methods are there i.e

POST: For creating the token for user, which is valid for one hour.

URL: http://localhost:3000/tokens

HEADERS:

- Content-Type: application/json

BODY: (For Example)

```
{
        "email": "john@gmail.com",
        "password": "password"
}
```

RESPONSE:

```
{
         "email": "john@gmail.com",
         "id": "8bxuak6o5jeccir2i6n4",
         "expires": 1545827726979
}
```

RESPONSE HTTP STATUS CODES:

- **200 - Success**
- **400 - Bad Request**
- **500 - Internal Server Error**

GET: For getting the token information. Need to send the token id (generated in the token post request)l as query string as shown in the URL.

URL: http://localhost:3000/users?email=john@gmail.com

HEADERS:

- Content-Type: application/json

BODY: None

RESPONSE:

```
{
         "email": "john@gmail.com",
         "id": "8bxuak6o5jeccir2i6n4",
         "expires": 1545827726979
}
```

RESPONSE HTTP STATUS CODES:

- **200 - Success**
- **400 - Bad Request**
- **404 - Not Found**

PUT: For Extending the token authorization time.

URL: http://localhost:3000/tokens

HEADERS:

- Content-Type: application/json

BODY: (For Example).

```
{
        "id": "8bxuak6o5jeccir2i6n4",
        "extend": true
}
```

RESPONSE: None

RESPONSE HTTP STATUS CODES:

- **200 - Success**
- **400 - Bad Request**
- **500 - Internal Server Error**

DELETE: For deleting the authorization token. Need to send the token id (generated in the token post request)l as query string as shown in the URL.

URL: http://localhost:3000/tokens?id=8bxuak6o5jeccir2i6n4

HEADERS:

- Content-Type: application/json

BODY: None

RESPONSE: None

RESPONSE HTTP STATUS CODES:

- **200 - Success**
- **400 - Bad Request**
- **404 - Not Found**
- **500 - Internal Server Error**

MENU: /menu

The menu route is used get the pizza related details. In the /menu route, only one method is there i.e

GET: For getting the pizzas information. Need to send the email as query string shown in the URL.

URL:http://localhost:3000/menu?email=john@gmail.com

HEADERS:

- Content-Type: application/json
- token : (please see the /tokens route Post method for getting the token)

BODY: None

RESPONSE:

```
{
  "pizzas": [
    {
      "name": "veg extravaganza",
      "type": "veg",
      "basePrice": 230
    },
    {
      "name": "Farm House",
      "type": "veg",
      "basePrice": 200
    },
    {
      "name": "Mexican Herbs",
      "type": "veg",
      "basePrice": 230
    },
    {
      "name": "Paneer Makhani",
      "type": "veg",
      "basePrice": 200
    },
    {
      "name": "Chicken Dominator",
      "type": "non-veg",
      "basePrice": 330
    },
    {
      "name": "Chicken Golden Delight",
      "type": "non-veg",
      "basePrice": 300
    },
    {
      "name": "Chicken Supreme",
      "type": "non-veg",
      "basePrice": 350
```

```json
        },
        {
            "name": "Peri Peri Chicken",
            "type": "non-veg",
            "basePrice": 330
        }
    ],
    "toppings": [
        {
            "name": "Chicken Tikka",
            "price": 80,
            "Type": "non-veg"
        },
        {
            "name": "Chicken Sausage",
            "price": 80,
            "Type": "non-veg"
        },
        {
            "name": "Grilled Chicken Rasher",
            "price": 90,
            "Type": "non-veg"
        },
        {
            "name": "Jalpeno",
            "price": 30,
            "Type": "veg"
        },
        {
            "name": "Golden Cron",
            "price": 50,
            "Type": "veg"
        },
        {
            "name": "Paneer",
            "price": 60,
            "Type": "veg"
        }
    ],
    "crusts": [
        {
            "name": "New Hand Tossed",
            "price": 10
        },
        {
            "name": "Wheat Thin Crust",
            "price": 15
        },
        {
            "name": "Cheese Brust",
            "price": 20
        },
        {
            "name": "Fresh Pan Pizza",
            "price": 20
        },
        {
            "name": "Classic Hand Tossed",
            "price": 30
        }
    ],
    "size": [
        {
            "regular": {
                "price": 100,
                "serve": 1
            },
            "medium": {
                "price": 170,
                "serve": 2
            },
            "large": {
                "price": 280,
                "serve": 4
            }
        }
    ]
}
```

RESPONSE HTTP STATUS CODES:

- **200 - Success**
- **400 - Bad Request**
- **403 - Forbidden**
- **404 - Not Found**

CARTS: /carts

The carts route is used for storing the items which user like. No post method for carts route as the cart is created automatically when a user is created. In the /carts route, three methods are there i.e

GET: For getting the cart item information. Need to send email as query string as shown in the URL.

URL: http://localhost:3000/carts?email=john@gmail.com

HEADERS:

- Content-Type: application/json
- token : (please see the /tokens route Post method for getting the token)

BODY: None

RESPONSE: returns array of items

```
[
  {
        "id": "j6vx7viubg3nmbk7ij7g",
        "pizzaName": "veg extravaganza",
        "type": "veg",
        "size": "regular",
        "toppings": false,
        "crust": "Cheese Brust",
        "quantity": 1,
        "grandTotal": 350
  }
]
```

RESPONSE HTTP STATUS CODES:

- **200 - Success**
- **400 - Bad Request**
- **403 - Forbidden**
- **500 - Internal Server Error**

PUT: For updating the cart items. Need to send email as query string as shown in the URL.

URL: http://localhost:3000/carts?email=john@gmail.com

HEADERS:

- Content-Type: application/json
- token : (please see the /tokens route Post method for getting the token)

BODY:

```
{
        "pizza": {
                "name": "veg extravaganza",
                "type": "veg",
                "basePrice": 230
        },
        "crust": {
                "name": "Cheese Brust",
                "price": 20
        },
        "size": {
                "name": "regular",
                "price": 100
        },
        "quantity": 1,
        "grandTotal": 350
}
```

RESPONSE: None

RESPONSE HTTP STATUS CODES:

- **200 - Success**
- **400 - Bad Request**
- **403 - Forbidden**
- **500 - Internal Server Error**

DELETE: For deleting particular item in the cart. Need to send email and cart product id as query string as shown in the URL.

URL: http://localhost:3000/carts?email=john@gmail.com&id=j6vx7viubg3nmbk7ij7g

HEADERS:

- Content-Type: application/json
- token : (please see the /tokens route Post method for getting the token)

BODY: None

RESPONSE: None

RESPONSE HTTP STATUS CODES:

- **200 - Success**
- **400 - Bad Request**
- **403 - Forbidden**
- **500 - Internal Server Error**

ORDERS: /orders

The orders route is used placing an order. In the /orders route, two methods are there i.e

POST: For placing an order. Need to send email and cart product id as query string as shown in the URL.

URL: http://localhost:3000/orders?email=john@gmail.com

HEADERS:

- Content-Type: application/json
- token : (please see the /tokens route Post method for getting the token)

BODY: None

RESPONSE: None

RESPONSE HTTP STATUS CODES:

- **200 - Success**
- **400 - Bad Request**
- **500 - Internal Server Error**

GET: For getting all orders details of a user. Need to send email as query string as shown in the URL.

URL: http://localhost:3000/orders?email=john@gmail.com

HEADERS:

- Content-Type: application/json
- token : (please see the /tokens route Post method for getting the token)

BODY: None

RESPONSE: returns array of items

```
[
  {
    "id": "x7h9x59284t4mmwzni4i",
    "name": "john carter",
    "email": "john@gmail.com",
    "phone": "1234567890",
    "address": "221B, Baker street, London",
    "items": [
      {
        "id": "oh40hvnfvykorj7goqzu",
        "pizzaName": "veg extravaganza",
        "type": "veg",
        "size": "regular",
        "toppings": false,
        "crust": "Cheese Brust",
        "quantity": 1,
        "grandTotal": 350
      }
    ],
    "quantity": 1,
    "amount": 350,
    "deliveryDate": 1545990127508,
    "createdDate": 1545817327508,
    "chargeId": "ch_1DlYtxLuENZcmi8UG07gQHxJ"
  }
]
```

RESPONSE HTTP STATUS CODES:

- **200 - Success**
- **400 - Bad Request**
- **403 - Forbidden**
- **500 - Internal Server Error**