Helix Project - API Documentation

Base URL

http://127.0.0.1:8000/api/v1/

Authentication

Token-based (except register/login)

Header: Authorization: Token <your_token>

1. Register [POST /register/]

Request:

```
{
  "username": "john123",
  "email": "john@example.com",
  "password": "strongpass"
}
```

Response:

```
{
  "token": "abcd1234..."
}
```

2. Login [POST /login/]

Request:

```
{
  "username": "john123",
  "password": "strongpass"
}
```

Response:

```
{
  "token": "abcd1234..."
}
```

3. List Menu Items [GET /menu/]

Response:

```
[
  {
    "id": 1,
    "name": "Big Mac",
    "description": "Double beef burger",
    "price": "149.00",
    "available": true,
    "image_url": "http://localhost:8000/media/images/bigmac.jpg"
  }
]
```

4. Add Menu Item [POST /menu/] (Admin Only)

Form Data (multipart):

- name: "McFlurry"

- description: "Vanilla dessert"

- price: 99

- available: true

- image: (file)

Response:

```
{

  "id": 2,

  "name": "McFlurry",

  "description": "Vanilla dessert",

  "price": "99.00",

  "available": true

}
```

5. Get Menu Item [GET /menu/{id}/]

Response:

```
{

  "id": 2,

  "name": "McFlurry",

  "description": "Vanilla dessert",

  "price": "99.00",

  "available": true,

  "image_url": "http://localhost:8000/media/images/mcflurry.jpg"

}
```

6. Update Menu Item [PUT /menu/{id}/] (Admin Only)

Request:

```
{

  "name": "McFlurry Deluxe",

  "description": "Deluxe vanilla dessert",
```

```
  "price": 109,

  "available": true

}
```

Response:

```
{

  "id": 2,

  "name": "McFlurry Deluxe",

  "description": "Deluxe vanilla dessert",

  "price": "109.00",

  "available": true

}
```

7. Delete Menu Item [DELETE /menu/{id}/] (Admin Only)

Response: 204 No Content

8. List Orders [GET /orders/]

Response:

```
[

 {

   "id": 4,

   "user": 2,

   "status": "placed",

   "total_price": "298.00",

   "created_at": "...",

   "items": [
```

```
    {

      "id": 1,

      "menu_item": { "id": 1, "name": "Big Mac", "price": "149.00" },

      "quantity": 2

    }

  ]

 }

]
```

9. Create Order [POST /orders/]

Request:

```
{

  "items": [

    { "menu_item_id": 1, "quantity": 2 },

    { "menu_item_id": 3, "quantity": 1 }

  ]

}
```

Response:

```
{

  "id": 5,

  "user": 2,

  "status": "placed",

  "total_price": "447.00",

  "created_at": "...",

  "items": [
```

```
  {

    "id": 1,

    "menu_item": { "id": 1, "name": "Big Mac", "price": "149.00" },

    "quantity": 2

  }

 ]

}
```

10. Get Single Order [GET /orders/{id}/]

Response:

```
{

  "id": 5,

  "user": 2,

  "status": "placed",

  "total_price": "447.00",

  "created_at": "...",

  "items": [...]

}
```

11. Update Order Status [PUT /orders/{id}/] (Admin Only)

Request:

```
{

  "status": "cancelled"

}
```

Response:

```
{

  "id": 5,

  "user": 2,

  "status": "cancelled",

  "total_price": "447.00",

  "created_at": "..."

}
```

12. Delete Order [DELETE /orders/{id}/] (Admin Only)

Response: 204 No Content