Drive Controller API Documentation

Overview

The Drive Controller manages delivery drives/routes, including creation, assignment, tracking, and completion of milk delivery operations. It handles the entire lifecycle of a delivery drive from planning to completion.

Endpoints

1. GET /drives - Get All Drives

Purpose: Retrieve a paginated list of all drives with optional filtering capabilities.

Parameters:

- `page` (optional): Page number for pagination (default: 1)
- `limit` (optional): Number of records per page (default: 10)
- `status` (optional): Filter by drive status (pending, ongoing, completed)
- `date` (optional): Filter by date (YYYY-MM format)
- `delivery_guy_id` (optional): Filter by delivery person ID
- `route_id` (optional): Filter by route ID

Sample Request:

```
GET /api/drives?page=1&limit=10&status=pending&delivery_guy_id=1
Authorization: Bearer your-token-here
```

Sample Response:

```json
{
  "drives": [
    {
      "drive_id": 1,
      "route_id": 1,
      "delivery_guy_id": 1,
      "stock": 100,
      "sold": 0,
      "returned": 0,
      "status": "pending",
      "name": "Morning Route A",
      "remarks": "Regular morning delivery",
      "created_at": "2025-07-18T06:00:00Z",
      "route_name": "Downtown Route",
      "delivery_guy_name": "John Doe"
    }
  ],
  "pagination": {
    "page": 1,
    "limit": 10,
    "totalCount": 25,
    "totalPages": 3
  }
}
```

2. GET /drives/:id - Get Drive By ID

Purpose: Retrieve detailed information about a specific drive.

Path Parameters:

- `id` : Drive ID

Sample Request:

```
GET /api/drives/1
Authorization: Bearer your-token-here
```

Sample Response:

```json
{
  "drive": {
    "drive_id": 1,
    "route_id": 1,
    "delivery_guy_id": 1,
    "stock": 100,
    "sold": 85,
    "returned": 15,
    "status": "completed",
    "name": "Morning Route A",
    "start_time": "2025-07-18T06:00:00Z",
    "end_time": "2025-07-18T10:30:00Z",
    "total_amount": 2125.50,
    "route_name": "Downtown Route",
    "delivery_guy_name": "John Doe"
  }
}
```

3. POST /drives - Create Drive

Purpose: Create a new delivery drive with route and stock information.

Request Body:

```json
{
  "route_id": 1,
  "delivery_guy_id": 1,
  "stock": 100,
  "remarks": "Morning delivery route",
  "drive_name": "Morning Route 5",
  "start_time": "2025-05-23T08:00:00Z"
}
```

Sample Request:

```
POST /api/drives
Authorization: Bearer your-token-here
Content-Type: application/json

{
  "route_id": 1,
  "delivery_guy_id": 1,
  "stock": 100,
  "remarks": "Morning delivery route",
  "drive_name": "Morning Route 5"
}
```

Sample Response:

```
{
  "message": "Drive created successfully",
  "drive": {
    "drive_id": 15,
    "route_id": 1,
    "delivery_guy_id": 1,
    "stock": 100,
    "sold": 0,
    "returned": 0,
    "status": "pending",
    "name": "Morning Route 5",
    "remarks": "Morning delivery route",
    "created_at": "2025-07-18T06:00:00Z"
  }
}
```

4. PUT /drives/:id - Update Drive

Purpose: Update drive details. Some fields can only be updated if the drive is still pending.

Path Parameters:

- `id` : Drive ID

Request Body:

```json
{
  "delivery_guy_id": 2,
  "stock": 120,
  "sold": 100,
  "returned": 20,
  "remarks": "Updated remarks",
  "status": "completed",
  "drive_name": "Updated Route Name"
}
```

Sample Request:

```
PUT /api/drives/1
Authorization: Bearer your-token-here
Content-Type: application/json

{
  "delivery_guy_id": 2,
  "sold": 85,
  "returned": 15,
  "status": "completed"
}
```

Sample Response:

```json
{
  "message": "Drive updated successfully",
  "drive": {
    "drive_id": 1,
    "route_id": 1,
    "delivery_guy_id": 2,
    "stock": 100,
    "sold": 85,
    "returned": 15,
    "status": "completed",
    "updated_at": "2025-07-18T10:30:00Z"
  }
}
```

5. DELETE /drives/:id - Delete Drive

Purpose: Delete a drive. Only pending drives can be deleted.

Path Parameters:

- `id` : Drive ID

Sample Request:

```
DELETE /api/drives/1
Authorization: Bearer your-token-here
```

Sample Response:

```
{
  "message": "Drive deleted successfully"
}
```

6. GET /drives/:id/details - Get Drive Details

Purpose: Get comprehensive drive details including associated customers and their delivery status.

Path Parameters:

- `id` : Drive ID

Sample Request:

```
GET /api/drives/1/details
Authorization: Bearer your-token-here
```

Sample Response:

```
{
  "drive": {
    "drive_id": 1,
    "route_id": 1,
    "delivery_guy_id": 1,
    "stock": 100,
    "status": "ongoing",
    "route_name": "Downtown Route",
    "delivery_guy_name": "John Doe"
  },
  "customers": [
    {
      "customer_id": 1,
      "name": "Alice Johnson",
      "phone": "9876543210",
      "address": "123 Main St",
      "default_quantity": 2,
      "delivered_quantity": 2,
      "delivery_status": "success"
    }
  ],
  "customerCount": 15
}
```

7. POST /drives/:id/assign - Assign Delivery Person

Purpose: Assign a delivery person to a pending drive.

Path Parameters:

- `id` : Drive ID

Request Body:

```
{
  "delivery_guy_id": 1
}
```

Sample Request:

```
POST /api/drives/1/assign
Authorization: Bearer your-token-here
Content-Type: application/json

{
  "delivery_guy_id": 1
}
```

Sample Response:

```
{
  "message": "Delivery person assigned to drive successfully",
  "drive": {
    "drive_id": 1,
    "route_id": 1,
    "delivery_guy_id": 1,
    "stock": 100,
    "status": "pending",
    "updated_at": "2025-07-18T05:30:00Z"
  }
}
```

8. POST /drives/:id/start - Start Drive

Purpose: Start a pending drive, changing its status to ongoing.

Path Parameters:

- `id` : Drive ID

Sample Request:

```
POST /api/drives/1/start
Authorization: Bearer your-token-here
```

Sample Response:

```
{
  "message": "Drive started successfully",
  "drive": {
    "drive_id": 1,
    "route_id": 1,
    "delivery_guy_id": 1,
    "stock": 100,
    "status": "ongoing",
    "start_time": "2025-07-18T06:00:00Z",
    "updated_at": "2025-07-18T06:00:00Z"
  }
}
```

9. POST /drives/:id/end - End Drive

Purpose: End an ongoing drive, marking it as completed with final sales data.

Path Parameters:

- `id` : Drive ID

Request Body:

```
{
  "sold": 90,
  "returned": 10,
  "remarks": "Completed successfully. 2 customers were absent."
}
```

Sample Request:

```
POST /api/drives/1/end
Authorization: Bearer your-token-here
Content-Type: application/json

{
  "sold": 85,
  "returned": 15,
  "remarks": "Route completed successfully"
}
```

Sample Response:

```
{
  "message": "Drive ended successfully",
  "drive": {
    "drive_id": 1,
    "route_id": 1,
    "delivery_guy_id": 1,
    "stock": 100,
    "sold": 85,
    "returned": 15,
    "status": "completed",
    "end_time": "2025-07-18T10:30:00Z",
    "total_amount": 2125.50,
    "remarks": "Route completed successfully"
  }
}
```

10. GET /drives/:id/locations - Get Drive Locations

Purpose: Retrieve all GPS location logs for a specific drive.

Path Parameters:

- `id` : Drive ID

Sample Request:

```
GET /api/drives/1/locations
Authorization: Bearer your-token-here
```

Sample Response:

```
{
  "drive_id": 1,
  "locations": [
    {
      "drive_location_id": 1,
      "latitude": 12.9716,
      "longitude": 77.5946,
      "time": "2025-07-18T06:15:00Z"
    },
    {
      "drive_location_id": 2,
      "latitude": 12.9720,
      "longitude": 77.5950,
      "time": "2025-07-18T06:30:00Z"
    }
  ]
}
```

11. POST /drives/:id/locations - Log Drive Location

Purpose: Log a new GPS location for an ongoing drive.

Path Parameters:

- `id` : Drive ID

Request Body:

```
{
  "latitude": 12.9716,
  "longitude": 77.5946
}
```

Sample Request:

```
POST /api/drives/1/locations
Authorization: Bearer your-token-here
Content-Type: application/json

{
  "latitude": 12.9716,
  "longitude": 77.5946
}
```

Sample Response:

```
{
  "message": "Location logged successfully",
  "location": {
    "drive_location_id": 15,
    "latitude": 12.9716,
    "longitude": 77.5946,
    "time": "2025-07-18T06:45:00Z"
  }
}
```

12. GET /drives/:id/summary - Get Drive Summary

Purpose: Get comprehensive performance summary and statistics for a drive.

Path Parameters:

- `id` : Drive ID

Sample Request:

```
GET /api/drives/1/summary
Authorization: Bearer your-token-here
```

Sample Response:

```json
{
  "drive": {
    "drive_id": 1,
    "route_id": 1,
    "delivery_guy_id": 1,
    "stock": 100,
    "sold": 85,
    "returned": 15,
    "status": "completed",
    "route_name": "Downtown Route",
    "delivery_guy_name": "John Doe",
    "total_amount": 2125.50
  },
  "summary": {
    "total_customers": 15,
    "totalquantity": 85,
    "total_amount": 2125.50,
    "successfuldeliveries": 13,
    "failed_deliveries": 2,
    "totalroutecustomers": 15,
    "deliveryRate": 86.67,
    "returnRate": 15.0
  }
}
```

13. GET /drives/:id/manifest - Get Drive Manifest

Purpose: Get detailed manifest of a drive including all customers and expected quantities.

Path Parameters:

- `id` : Drive ID

Sample Request:

```
GET /api/drives/1/manifest
Authorization: Bearer your-token-here
```

Sample Response:

```json
{
  "drive": {
    "drive_id": 1,
    "date": "2025-07-18T06:00:00Z",
    "route_name": "Downtown Route",
    "deliveryPerson": "John Doe",
    "stock": 100
  },
  "customers": [
    {
      "customer_id": 1,
      "name": "Alice Johnson",
      "location": "Downtown",
      "phone": "9876543210",
      "address": "123 Main St",
      "defaultquantity": 2,
      "points": 50,
      "status": "active"
    },
    {
      "customer_id": 2,
      "name": "Bob Smith",
      "location": "Downtown",
      "phone": "9876543211",
      "address": "456 Oak Ave",
      "defaultquantity": 3,
      "points": 75,
      "status": "active"
    }
  ],
  "customerCount": 15,
  "totalExpectedQuantity": 45
}
```

Authentication

All endpoints require Bearer token authentication:

```
Authorization: Bearer your-jwt-token-here
```

Error Responses

All endpoints return consistent error responses:

400 Bad Request:

```
{
  "message": "Route ID and stock are required"
}
```

404 Not Found:

```
{
  "message": "Drive not found"
}
```

500 Internal Server Error:

```
{
  "message": "Server error"
}
```

Business Rules

1. **Drive Status Flow**: pending → ongoing → completed
2. **Deletion**: Only pending drives can be deleted
3. **Route/Stock Modification**: Can only modify route_id and stock for pending drives
4. **Location Logging**: Only allowed for ongoing drives
5. **Assignment**: Delivery person must be assigned before starting a drive
6. **Manifest**: Shows expected customer deliveries based on route assignments

Database Tables Used

- `drives` : Main drive information
- routes: Route details
- `delivery_guys` : Delivery person information
- `customers` : Customer information

- `route_customers` : Route-customer relationships
- `drive_customers_sales` : Individual delivery records
- `drive_locations_log` : GPS tracking data