DineWise API Documentation

Table of Contents

Overview

Base URL

Development: http://localhost:5000/api/v1
Production: https://api.dinewise.com/v1

Authentication

- All authenticated endpoints require a Bearer token in the Authorization header
- Tokens are obtained through the login endpoint
- Example: `Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...`

Response Format

All responses follow this general format:

```
{
 "success": true,
 "data": {},
 "message": "Operation successful",
 "errors": null
}
```

Error Handling

Error responses include:

```
{
 "success": false,
 "data": null,
 "message": "Error description",
 "errors": {
   "field": ["Error details"]
 }
}
```

Common HTTP Status Codes: - 200: Success - 201: Created - 400: Bad Request - 401: Unauthorized - 403: Forbidden - 404: Not Found - 500: Internal Server Error

Authentication Endpoints

User Login

POST /auth/login

Request:

```
{
 "email": "user@example.com",
 "password": "password123",
 "userType": "customer"
}
```

Response:

```
{
  "success": true,
  "data": {
    "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...",
    "user": {
      "id": "123",
      "email": "user@example.com",
      "firstName": "John",
      "lastName": "Doe",
      "userType": "customer",
      "preferencesCompleted": true
    }
  },
  "message": "Login successful"
}
```

User Registration

POST /auth/register

Request:

```
{
  "email": "user@example.com",
  "password": "password123",
  "firstName": "John",
  "lastName": "Doe",
  "phone": "+1234567890",
  "userType": "customer"
}
```

Response:

```
{
  "success": true,
  "data": {
    "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...",
    "user": {
      "id": "123",
      "email": "user@example.com",
      "firstName": "John",
      "lastName": "Doe",
      "userType": "customer",
      "preferencesCompleted": false
    }
  },
  "message": "Registration successful"
}
```

Restaurant Endpoints

List Restaurants

GET /restaurants

Query Parameters: - page (optional): Page number (default: 1) - limit (optional): Items per page (default: 10) - category (optional): Filter by cuisine category - search (optional): Search term

Response:

```
{
  "success": true,
  "data": {
    "restaurants": [
      {
        "id": "123",
        "businessName": "Spice Garden",
        "cuisine": "Indian",
        "rating": 4.5,
        "businessAddress": "123 Curry Lane",
        "businessPhone": "+1234567890"
      }
    ],
    "pagination": {
      "currentPage": 1,
      "totalPages": 5,
      "totalItems": 50
    }
  },
  "message": "Restaurants retrieved successfully"
}
```

Get Restaurant Details

GET /restaurants/{restaurantId}

Response:

```
{
  "success": true,
  "data": {
    "id": "123",
    "businessName": "Spice Garden",
    "cuisine": "Indian",
    "rating": 4.5,
    "businessAddress": "123 Curry Lane",
    "businessPhone": "+1234567890",
    "operatingHours": {
      "monday": { "open": "09:00", "close": "22:00" },
      "tuesday": { "open": "09:00", "close": "22:00" }
    }
  },
  "message": "Restaurant details retrieved successfully"
}
```

Menu Management Endpoints

Add Menu Item

POST /restaurants/{restaurantId}/menu

Request:

```
{
  "name": "Butter Chicken",
  "description": "Creamy tomato-based curry with tender chicken",
  "price": 15.99,
  "category": "Main Course",
  "isSpicy": true,
  "dietaryRestrictions": ["Gluten-Free"],
  "preparationTime": 20
}
```

Response:

```
{
  "success": true,
  "data": {
    "id": "item123",
    "name": "Butter Chicken",
    "description": "Creamy tomato-based curry with tender chicken",
    "price": 15.99,
    "category": "Main Course",
    "isSpicy": true,
    "dietaryRestrictions": ["Gluten-Free"],
    "preparationTime": 20
  },
  "message": "Menu item added successfully"
}
```

User Preference Endpoints

Update User Preferences

PUT /users/preferences

Request:

```
{
 "dietaryRestrictions": ["Vegetarian", "Gluten-Free"],
 "spicePreference": "medium",
 "priceRange": "moderate",
 "favoriteCategories": ["Indian", "Italian"],
 "allergies": ["nuts", "shellfish"]
}
```

Response:

```
{
 "success": true,
 "data": {
  "preferences": {
   "dietaryRestrictions": ["Vegetarian", "Gluten-Free"],
   "spicePreference": "medium",
   "priceRange": "moderate",
   "favoriteCategories": ["Indian", "Italian"],
   "allergies": ["nuts", "shellfish"],
   "updatedAt": "2024-12-10T12:00:00Z"
  }
 },
 "message": "Preferences updated successfully"
}
```

Recommendation Endpoints

Get Personalized Recommendations

GET /recommendations

Query Parameters: - `location` (optional): User’s location coordinates - `maxDistance` (optional): Maximum distance in kilometers - `mealTime` (optional): breakfast, lunch, dinner - `priceRange` (optional): budget, moderate, premium

Response:

```
{
 "success": true,
 "data": {
  "recommendations": [
   {
    "restaurantId": "123",
    "businessName": "Spice Garden",
    "menuItems": [
     {
      "id": "item123",
      "name": "Butter Chicken",
      "matchScore": 0.95,
      "price": 15.99,
      "dynamicPrice": 14.99,
      "recommendationReason": "Matches your spice preference and dietary restrictions"
     }
    ]
   }
  ]
 },
 "message": "Recommendations retrieved successfully"
}
```

Rate Limiting

- All endpoints are rate-limited to prevent abuse
- Rate limits are based on IP address and API key
- Default limits:
  - Authenticated: 1000 requests per hour
  - Unauthenticated: 100 requests per hour

Webhooks

For real-time updates, subscribe to webhooks: - POST /webhooks/subscribe - Available events: - order.created - order.updated - menu.updated - preferences.updated

SDK Support

Official SDKs are available for: - JavaScript/TypeScript - Python - Java - Ruby

API Versioning

- Current version: v1
- Version is specified in the URL path
- Breaking changes will result in a new version
- Old versions are supported for 6 months after deprecation

Best Practices

1. Always check the response status code
2. Implement proper error handling
3. Use pagination for large datasets
4. Cache responses when appropriate
5. Include proper authentication headers
6. Follow rate limiting guidelines