**Documentación API CRM**

**📋 Información General**

**Base URL:** http://localhost:3000  
**Versión:** 1.0.0  
**Autenticación:** JWT Bearer Token

⚠️ **Nota**: Esta API está en desarrollo activo y será escalable. Se agregarán más endpoints para clientes, ventas, tareas y recursos humanos.

**🔐 Autenticación**

Todas las rutas protegidas requieren un token JWT en el header:

Authorization: Bearer <token>

**📊 Endpoints Disponibles**

**🔓 Autenticación (/api/auth)**

**Registrar Usuario**

http

POST /api/auth/register

**Body:**

json

{

"email": "usuario@ejemplo.com",

"password": "password123",

"firstname": "Juan",

"lastname": "Pérez",

"rol": "EMPRENDEDOR",

"subRol": "COMERCIAL",

"phone": "+1234567890"

}

**Respuesta exitosa:**

json

{

"message": "Usuario creado exitosamente",

"user": {

"id": "clx1...",

"email": "usuario@ejemplo.com",

"firstname": "Juan",

"lastname": "Pérez",

"rol": "EMPRENDEDOR",

"subRol": "COMERCIAL",

"isActive": true,

"createdAt": "2024-01-01T00:00:00.000Z"

},

"token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9..."

}

**Iniciar Sesión**

http

POST /api/auth/login

**Body:**

json

{

"email": "usuario@ejemplo.com",

"password": "password123"

}

**Respuesta exitosa:**

json

{

"message": "Login exitoso",

"user": {

"id": "clx1...",

"email": "usuario@ejemplo.com",

"firstname": "Juan",

"lastname": "Pérez",

"rol": "EMPRENDEDOR",

"subRol": "COMERCIAL",

"isActive": true

},

"token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9..."

}

**Obtener Perfil**

http

GET /api/auth/profile

**Headers:** Authorization: Bearer <token>

**Actualizar Perfil**

http

PUT /api/auth/profile

**Headers:** Authorization: Bearer <token>

**Body:**

json

{

"firstname": "Juan Carlos",

"lastname": "Pérez López",

"phone": "+1234567890"

}

**Cambiar Contraseña**

http

PUT /api/auth/change-password

**Headers:** Authorization: Bearer <token>

**Body:**

json

{

"currentPassword": "password123",

"newPassword": "newpassword456"

}

**Verificar Token**

http

GET /api/auth/verify

**Headers:** Authorization: Bearer <token>

**Cerrar Sesión**

http

POST /api/auth/logout

**Headers:** Authorization: Bearer <token>

**👥 Usuarios (/api/users)**

**Permisos:** La mayoría requieren rol SUPER\_ADMIN o DISTRIBUIDOR

**Listar Usuarios**

http

GET /api/users?page=1&limit=50&search=juan&rol=EMPRENDEDOR&isActive=true

**Headers:** Authorization: Bearer <token>

**Respuesta:**

json

{

"users": [

{

"id": "clx1...",

"email": "usuario@ejemplo.com",

"firstname": "Juan",

"lastname": "Pérez",

"rol": "EMPRENDEDOR",

"subRol": "COMERCIAL",

"phone": "+1234567890",

"isActive": true,

"createdAt": "2024-01-01T00:00:00.000Z",

"updatedAt": "2024-01-01T00:00:00.000Z"

}

],

"pagination": {

"total": 25,

"page": 1,

"limit": 50,

"pages": 1

}

}

**Obtener Usuario por ID**

http

GET /api/users/:id

**Crear Usuario**

http

POST /api/users

**Body:** (mismo que registro)

**Actualizar Usuario**

http

PUT /api/users/:id

**Cambiar Contraseña de Usuario**

http

PUT /api/users/:id/password

**Desactivar/Activar Usuario**

http

PUT /api/users/:id/deactivate

PUT /api/users/:id/activate

**Eliminar Usuario**

http

DELETE /api/users/:id

**Estadísticas de Usuarios**

http

GET /api/users/stats

**🎯 Pipeline de Ventas (/api/pipeline)**

**Listar Items del Pipeline**

http

GET /api/pipeline?page=1&limit=50&status=NUEVO&priority=ALTA&search=juan

**Headers:** Authorization: Bearer <token>

**Respuesta:**

json

{

"items": [

{

"id": "clx1...",

"name": "Juan Pérez",

"phone": "+1234567890",

"products": ["Producto A", "Producto B"],

"value": "1500",

"priority": "ALTA",

"status": "NUEVO",

"lastContact": "2024-01-01T00:00:00.000Z",

"demoDate": null,

"deliveryDate": null,

"paymentPlan": null,

"notes": "Cliente interesado en productos premium",

"tags": ["premium", "urgente"],

"client": {

"id": "clx2...",

"nombre": "Juan",

"apellido": "Pérez",

"email": "juan@ejemplo.com"

},

"assignedTo": {

"id": "clx3...",

"firstname": "María",

"lastname": "González"

},

"createdAt": "2024-01-01T00:00:00.000Z",

"updatedAt": "2024-01-01T00:00:00.000Z"

}

],

"pagination": {

"total": 15,

"page": 1,

"limit": 50,

"pages": 1

}

}

**Vista Kanban (Agrupado por Estado)**

http

GET /api/pipeline/kanban?assignedTo=clx1...

**Respuesta:**

json

{

"pipeline": {

"NUEVO": [

{

"id": "clx1...",

"name": "Juan Pérez",

"phone": "+1234567890",

*// ... más campos*

}

],

"CONTACTADO": [

*// items en estado CONTACTADO*

],

"CITA\_AGENDADA": [

*// items con citas agendadas*

]

*// ... otros estados*

}

}

**Crear Item del Pipeline**

http

POST /api/pipeline

**Headers:** Authorization: Bearer <token>

**Body:**

json

{

"name": "Juan Pérez",

"phone": "+1234567890",

"products": ["Producto A", "Producto B"],

"value": "1500",

"priority": "ALTA",

"status": "NUEVO",

"lastContact": "2024-01-01T12:00:00.000Z",

"demoDate": "2024-01-05T15:00:00.000Z",

"deliveryDate": "2024-01-10T10:00:00.000Z",

"paymentPlan": "3 cuotas sin interés",

"notes": "Cliente muy interesado",

"tags": ["premium", "referido"],

"clientId": "clx2...",

"assignedToId": "clx3..."

}

**Obtener Item por ID**

http

GET /api/pipeline/:id

**Actualizar Item**

http

PUT /api/pipeline/:id

**Cambiar Estado (Drag & Drop)**

http

PATCH /api/pipeline/:id/status

**Body:**

json

{

"status": "CONTACTADO",

"position": 1

}

**Duplicar Item**

http

POST /api/pipeline/:id/duplicate

**Eliminar Item**

http

DELETE /api/pipeline/:id

**Estadísticas del Pipeline**

http

GET /api/pipeline/stats?assignedTo=clx1...&period=30

**Respuesta:**

json

{

"total": 45,

"converted": 12,

"conversionRate": 26.67,

"byStatus": [

{

"status": "NUEVO",

"\_count": {

"\_all": 15

}

},

{

"status": "CONTACTADO",

"\_count": {

"\_all": 8

}

}

],

"byPriority": [

{

"priority": "ALTA",

"\_count": {

"\_all": 10

}

}

],

"byUser": [

{

"assignedToId": "clx1...",

"\_count": {

"\_all": 25

},

"user": {

"id": "clx1...",

"firstname": "María",

"lastname": "González"

}

}

]

}

**Buscar Items**

http

GET /api/pipeline/search?q=juan&status=NUEVO&priority=ALTA&limit=10

**Actualización Masiva**

http

POST /api/pipeline/bulk-update

**Permisos:** Solo SUPER\_ADMIN y DISTRIBUIDOR

**Body:**

json

{

"itemIds": ["clx1...", "clx2...", "clx3..."],

"updates": {

"status": "CONTACTADO",

"assignedToId": "clx4..."

}

}

**🔑 Roles y Permisos**

**Roles Disponibles:**

* **SUPER\_ADMIN**: Acceso total al sistema
* **DISTRIBUIDOR**: Gestión de usuarios y vista global
* **EMPRENDEDOR**: Vista limitada a sus propios datos
* **ASISTENTE**: Roles específicos según subRol

**SubRoles para ASISTENTE:**

* RRHH - Asistente de Recursos Humanos
* COMERCIAL - Asistente Comercial

**📈 Estados del Pipeline**

javascript

const PIPELINE\_STATUSES = [

'NUEVO',

'CONTACTADO',

'CITA\_AGENDADA',

'SIN\_RESPUESTA',

'REPROGRAMAR',

'NO\_VENTA',

'IRRELEVANTE',

'NO\_QUIERE\_SPV',

'NO\_QUIERE\_DEMO',

'VENTA\_AGREGADO',

'VENTA\_NUEVA',

'VENTA\_CAIDA'

];

**🎯 Prioridades**

javascript

const PRIORITIES = [

'ALTA',

'MEDIA',

'BAJA'

];

**⚡ Rate Limiting**

* **Autenticación**: 5 intentos cada 15 minutos
* **Registro**: 3 registros por hora por IP
* **General**: 100 requests cada 15 minutos

**🚨 Códigos de Error**

| **Código** | **Significado** |
| --- | --- |
| 400 | Bad Request - Datos inválidos |
| 401 | Unauthorized - Token requerido/inválido |
| 403 | Forbidden - Sin permisos |
| 404 | Not Found - Recurso no encontrado |
| 429 | Too Many Requests - Rate limit excedido |
| 500 | Internal Server Error |

**🔄 Ejemplo de Integración**

javascript

*// Configuración del cliente*

const API\_BASE\_URL = 'http://localhost:3000';

let authToken = null;

*// Login*

async function login(email, password) {

const response = await fetch(`${API\_BASE\_URL}/api/auth/login`, {

method: 'POST',

headers: {

'Content-Type': 'application/json',

},

body: JSON.stringify({ email, password })

});

const data = await response.json();

if (response.ok) {

authToken = data.token;

localStorage.setItem('token', authToken);

return data.user;

} else {

throw new Error(data.error);

}

}

*// Obtener pipeline*

async function getPipeline() {

const response = await fetch(`${API\_BASE\_URL}/api/pipeline`, {

headers: {

'Authorization': `Bearer ${authToken}`,

}

});

return await response.json();

}

*// Crear item en pipeline*

async function createPipelineItem(itemData) {

const response = await fetch(`${API\_BASE\_URL}/api/pipeline`, {

method: 'POST',

headers: {

'Content-Type': 'application/json',

'Authorization': `Bearer ${authToken}`,

},

body: JSON.stringify(itemData)

});

return await response.json();

}

**🔮 Próximos Endpoints (En Desarrollo)**

* **/api/clients** - Gestión de clientes
* **/api/sales** - Registro y seguimiento de ventas
* **/api/tasks** - Sistema de tareas y recordatorios
* **/api/hr** - Gestión de candidatos RRHH
* **/api/reports** - Reportes y analytics
* **/api/integrations** - Integraciones externas

**📞 Health Check**

http

GET /health

**Respuesta:**

json

{

"status": "OK",

"message": "CRM API funcionando correctamente",

"timestamp": "2024-01-01T00:00:00.000Z",

"uptime": 3600,

"environment": "development"

}