



ACTIVIDAD FINAL 2ª EVALUACIÓN SISTEMAS DE GESTIÓN EMPRESARIAL

DESARROLLO DEL CRM: DATA NEXUS



MARIO TOMÉ
2º DAM





ÍNDICE DEL INFORME

1. Introducción.....	3
1.1 Objetivo del informe.....	3
1.2 Descripción general del proyecto.....	3
1.3 Alcance.....	4
2. FASE 1: Análisis.....	5
2.1 Requerimientos funcionales.....	5
2.2 Requerimientos no funcionales.....	7
2.3 Análisis de usuarios.....	8
2.4 Análisis de competencia.....	9
3. FASE 2: Diseño.....	10
3.1 Arquitectura de sistema.....	10
3.2 Diseño de la base de datos.....	11
3.3 Diseño de la interfaz de usuario.....	13
3.4 Asistentes por visita.....	15
4. FASE 3: Desarrollo.....	17
4.1 Configuración del entorno de desarrollo.....	17
4.2 Implementación de la base de datos.....	18
4.3 Desarrollo de las vistas principales.....	19
4.4 Desarrollo de asistente.....	24
4.5 Implementación de elementos visuales.....	24
5. Conclusiones.....	25
5.1 Resumen de logros.....	25
5.2 Retos y soluciones.....	26
5.3 Recomendaciones futuras.....	27





1. INTRODUCCIÓN

1.1. OBJETIVO DEL INFORME

Se ha desarrollado este informe con el propósito de documentar detalladamente el proceso de creación y desarrollo del CRM (*Customer Relationship Management*) diseñado para optimizar la gestión empresarial. El objetivo principal es proporcionar una visión integral de las fases de **análisis**, **diseño** y **desarrollo** que han constituido el proyecto.

Además, se busca resaltar las **funcionalidades implementadas**, las **decisiones técnicas** tomadas y los **desafíos enfrentados** durante el desarrollo.

1.2. DESCRIPCIÓN GENERAL DEL PROYECTO

Se ha implementado un sistema CRM robusto utilizando el lenguaje de programación **Python**, enfocado en mejorar la gestión de relaciones con los clientes y optimizar los procesos internos de la empresa.

El proyecto abarca diversas funcionalidades que permiten la administración eficiente de *clientes, oportunidades de negocio, presupuestos, inventario, tareas, calendarios, y un perfil* personalizado para los usuarios.

Además, se ha incorporado una interfaz de usuario **intuitiva** con un menú lateral navegable y desplegable, y la capacidad de ocultarlo para mejorar la experiencia del usuario.

El CRM desarrollado incluye las siguientes vistas principales:

- 1. Información del CRM:** Ofrece un saludo personalizado, accesos directos a funciones clave y enlaces a redes sociales.
- 2. Clientes:** Permite la creación, edición y eliminación de clientes, gestionando información esencial como ID/CIF, nombre, dirección, teléfono, persona de contacto y correo electrónico.
- 3. Oportunidades:** Facilita la gestión de oportunidades de negocio vinculadas a clientes y presupuestos, con opciones para actualizar el estado de cada oportunidad.
- 4. Presupuestos:** Administra la creación y modificación de presupuestos, incluyendo detalles como fechas de creación y expiración, subtotales y totales.
- 5. Inventario:** Gestiona el inventario de productos, permitiendo añadir, editar y eliminar productos con información detallada.
- 6. Mandar Informe:** Automatiza el envío de informes por WhatsApp o correo electrónico, utilizando los datos de los clientes.





7. **Tareas:** Permite la creación y gestión de tareas asignadas, con detalles sobre prioridad y estado.
8. **Calendario:** Integra un calendario interactivo para la programación y gestión de eventos empresariales.
9. **Dudas:** Gestiona una sección de preguntas y respuestas frecuentes (FAQ) para facilitar la resolución de dudas comunes.
10. **Mi Perfil:** Ofrece opciones para personalizar el perfil del usuario, incluyendo la posibilidad de añadir una foto, cambiar datos personales y describir la empresa.

Adicionalmente, se ha desarrollado una carpeta denominada [asistentes](#), donde se encuentran los asistentes específicos para cada vista, **proporcionando orientación y soporte al usuario**.

Para añadir un toque innovador y visualmente atractivo, se ha creado la clase [rotatable_label.py](#), que permite la **rotación de 360°** del logo de la empresa, aportando un aspecto futurista al diseño de la interfaz.

1.3. ALCANCE

- **Funcionalidades Integrales:** Se ha desarrollado un CRM completo que abarca múltiples áreas de gestión empresarial, incluyendo clientes, oportunidades, presupuestos, inventario, tareas, calendarios y soporte al usuario.
- **Interfaz de Usuario Intuitiva:** La interfaz ha sido diseñada para ser fácil de usar, con un menú lateral navegable y la capacidad de ocultarlo según las preferencias del usuario.
- **Asistentes:** Se han implementado asistentes específicos para cada vista, que facilita la comprensión y el uso de la aplicación.
- **Elementos Visuales Innovadores:** La inclusión de la clase [rotatable_label.py](#) para la rotación del logo añade un elemento visual distintivo y moderno a la interfaz.
- **Gestión de Datos Eficiente:** La clase *DatabaseManager* maneja de manera efectiva la conexión y operaciones con la base de datos SQLite, garantizando la integridad y accesibilidad de los datos.
- **Soporte Multilenguaje:** La aplicación está desarrollada en español, lo que puede limitar su uso en entornos multilingües o en mercados internacionales sin una localización adecuada.





2. FASE 1: ANÁLISIS

2.1. REQUERIMIENTOS FUNCIONALES

Se han identificado y definido los requerimientos funcionales necesarios para el desarrollo del CRM, asegurando que el sistema cumpla con las necesidades específicas de gestión empresarial.

Entre los principales requerimientos funcionales se han establecido:

- **Gestión de Empresas y Autenticación:**
 - Se ha implementado la funcionalidad de registro de empresas, permitiendo a los usuarios crear una cuenta proporcionando el nombre de la empresa, correo electrónico y contraseña.
 - Se ha desarrollado un sistema de autenticación seguro para que los usuarios puedan iniciar sesión utilizando sus credenciales registradas.
- **Gestión de Clientes:**
 - Se ha creado una interfaz para la creación, edición y eliminación de clientes, gestionando información esencial como ID/CIF, nombre, dirección, teléfono, persona de contacto y correo electrónico.
 - Se ha integrado una tabla dinámica que muestra todos los clientes registrados, con opciones para editar o eliminar cada entrada.
- **Gestión de Oportunidades:**
 - Se ha desarrollado una funcionalidad para crear y gestionar oportunidades de negocio, vinculándolas a clientes específicos y presupuestos.
 - Se ha implementado la posibilidad de actualizar el estado de cada oportunidad (en proceso, perdido, ganado) y de visualizar un resumen de las mismas.
- **Gestión de Presupuestos:**
 - Se ha diseñado una sección para la creación y modificación de presupuestos, incluyendo detalles como fechas de creación y expiración, subtotales y totales.
 - Se ha integrado una funcionalidad para asociar presupuestos a clientes y gestionar su estado.





- **Gestión de Inventario:**
 - Se ha implementado un módulo para la administración de productos, permitiendo añadir, editar y eliminar productos con información detallada como proveedor, nombre, descripción, IVA, precio y stock.
 - Se ha desarrollado una tabla interactiva que muestra el inventario actual y facilita su gestión.
- **Envío de Informes:**
 - Se ha creado una funcionalidad que permite enviar informes automáticos por WhatsApp o correo electrónico, utilizando los datos de los clientes almacenados en el sistema.
- **Gestión de Tareas:**
 - Se ha implementado una sección para la creación y administración de tareas asignadas a diferentes miembros de la empresa, con detalles sobre prioridad y estado.
 - Se ha integrado la posibilidad de actualizar el estado de las tareas y de gestionar su vencimiento.
- **Gestión de Calendario:**
 - Se ha desarrollado un calendario interactivo que permite programar y gestionar eventos empresariales, con detalles como título, hora, lugar y descripción.
 - Se ha añadido la funcionalidad de asignar eventos a usuarios específicos dentro de la empresa.
- **Gestión de Preguntas Frecuentes (FAQ):**
 - Se ha implementado una sección para la creación y gestión de preguntas y respuestas frecuentes, facilitando la resolución de dudas comunes de los usuarios.
 - Se ha desarrollado una funcionalidad de búsqueda y filtrado para acceder rápidamente a la información relevante.
- **Gestión de Perfil de Usuario:**
 - Se ha creado una interfaz para que los usuarios puedan personalizar su perfil, incluyendo la opción de añadir una foto, cambiar datos personales como nombre, correo electrónico y contraseña, y describir la empresa.





- **Asistentes:**

- Se ha implementado una carpeta `asistentes` que contiene asistentes específicos para cada vista, proporcionando orientación y soporte a los usuarios.
- Se ha integrado un asistente/bot accesible desde la interfaz, que explica el funcionamiento de la aplicación y responde a consultas comunes de los usuarios.

- **Elementos Visuales Innovadores:**

- Se ha desarrollado la clase `rotatable_label.py` para la rotación de 360° del logo de la empresa, añadiendo un toque futurista y dinámico a la interfaz de usuario.

2.2. REQUERIMIENTOS NO FUNCIONALES

Se han establecido los requerimientos no funcionales para asegurar que el CRM no solo cumpla con las funcionalidades necesarias, sino que también ofrezca un rendimiento óptimo y una experiencia de usuario satisfactoria.

Los principales requerimientos no funcionales identificados son:

- **Usabilidad:**

- Se ha diseñado una interfaz intuitiva y fácil de navegar, permitiendo a los usuarios interactuar con el sistema de manera eficiente sin necesidad de una formación extensa.
- Se ha implementado una estructura coherente en todas las vistas, garantizando una experiencia de usuario consistente.

- **Rendimiento:**

- Se ha optimizado el tiempo de carga de las vistas y la ejecución de consultas a la base de datos, asegurando una respuesta rápida del sistema incluso con un volumen significativo de datos.
- Se ha utilizado SQLite como sistema de gestión de bases de datos, adecuado para aplicaciones de tamaño medio con un rendimiento aceptable.

- **Seguridad:**

- Se ha implementado un sistema de autenticación seguro, protegiendo las credenciales de los usuarios mediante el uso de contraseñas encriptadas.
- Se ha asegurado la integridad de los datos mediante restricciones de clave foránea y validaciones en las entradas de datos.





2.3. ANÁLISIS DE USUARIOS

Se ha realizado un análisis detallado de los usuarios finales del CRM para comprender sus necesidades y expectativas, asegurando que el sistema desarrollado responda adecuadamente a sus requerimientos.

Los principales puntos analizados son:

1. Perfiles de Usuario:

- **Administradores:** Responsables de la configuración del sistema, gestión de usuarios y supervisión general de las operaciones.
- **Gerentes de Ventas:** Encargados de gestionar clientes, oportunidades y presupuestos, monitoreando el progreso de las ventas.
- **Personal de Soporte:** Utilizan el sistema para gestionar tareas, responder a preguntas frecuentes y mantener el inventario.
- **Usuarios Finales:** Empleados que interactúan con el sistema para realizar sus tareas diarias, como registrar clientes, crear presupuestos o gestionar eventos.

2. Necesidades y Expectativas:

- **Facilidad de Uso:** Los usuarios esperan una interfaz intuitiva que les permita realizar sus tareas de manera eficiente sin complicaciones.
- **Acceso Rápido a Información:** Necesitan acceder rápidamente a datos relevantes como clientes, oportunidades y presupuestos para tomar decisiones informadas.
- **Soporte y Asistencia:** Valorarán la presencia de asistentes y un chatbot que les brinde orientación y respuestas a sus consultas en tiempo real.
- **Personalización:** Desean la posibilidad de personalizar su perfil y adaptar algunas funcionalidades del sistema a sus preferencias individuales.
- **Seguridad de Datos:** Requieren que la información sensible esté protegida y que solo usuarios autorizados puedan acceder a determinadas secciones del sistema.





2.4. ANÁLISIS DE LA COMPETENCIA

Se ha llevado a cabo un análisis comparativo con otros sistemas CRM disponibles en el mercado para identificar ventajas competitivas y áreas de mejora. Los principales aspectos considerados son:

1. Características Comparativas:

- **Funcionalidades Básicas:** Se ha destacado la inclusión de elementos innovadores como asistentes personalizados y un chatbot interactivo.
- **Interfaz de Usuario:** Muchos CRM ofrecen interfaces complejas, contraste, el CRM desarrollado se enfoca en la simplicidad y facilidad de uso, con un diseño intuitivo y elementos visuales atractivos como el logo rotativo.
- **Personalización y Flexibilidad:** Algunos CRM permiten una alta personalización, pero a menudo a un costo elevado. El sistema desarrollado ofrece una buena flexibilidad sin comprometer la usabilidad.
- **Costo:** Al ser una solución desarrollada internamente, se ha optimizado para minimizar costos, ofreciendo una alternativa económica en comparación con soluciones comerciales que suelen tener tarifas recurrentes.
- **Soporte y Asistencia:** La inclusión de asistentes específicos y un chatbot proporciona un nivel de soporte al usuario que puede no estar presente en todos los CRM comerciales, mejorando la experiencia de usuario.

2. Ventajas Competitivas:

- **Innovación Visual:** La implementación de la clase ``rotatable_label.py`` para la rotación del logo aporta un elemento distintivo y moderno que mejora la estética de la interfaz.
- **Asistentes Personalizados:** Estos elementos mejoran significativamente la usabilidad y el soporte al usuario, facilitando la adopción del sistema y reduciendo la curva de aprendizaje.
- **Costo Efectivo:** Al no depender de licencias comerciales, el CRM desarrollado representa una opción más accesible para pequeñas y medianas empresas.
- **Modularidad y Escalabilidad:** La arquitectura modular permite una fácil expansión y personalización, adaptándose a las necesidades cambiantes de la empresa sin requerir una reestructuración completa.





3. FASE 2: DISEÑO

En la fase de diseño, se ha **planificado y estructurado la arquitectura del sistema, la base de datos, la interfaz de usuario y los componentes específicos** que conforman el CRM.

Este diseño detallado ha sido fundamental para garantizar que el desarrollo del sistema sea coherente, eficiente y alineado con los requerimientos identificados en la fase de análisis.

3.1. ARQUITECTURA DE SISTEMAS

Se ha diseñado una arquitectura modular y escalable que permite una fácil expansión y mantenimiento del CRM. La arquitectura se basa en el paradigma de **Modelo-Vista-Controlador (MVC)**, que separa las responsabilidades del sistema en tres componentes principales:

- **Modelo:** Encargado de la gestión de datos y la lógica de negocio. Incluye la clase ``DatabaseManager`` que maneja las operaciones CRUD con la base de datos SQLite.
- **Vista:** Responsable de la presentación de la información al usuario. Comprende todas las vistas desarrolladas, como ``clientes_view.py``, ``oportunidades_view.py``, entre otras.
- **Controlador:** Actúa como intermediario entre el modelo y la vista, manejando las interacciones del usuario, procesando las entradas y actualizando tanto el modelo como la vista según sea necesario.

Además, se ha incorporado una **capa de servicios** que gestiona funcionalidades transversales como la autenticación, la gestión de sesiones y la integración de componentes adicionales como los asistentes y el chatbot.

Esta arquitectura permite una alta cohesión y bajo acoplamiento entre los componentes, facilitando así la mantenibilidad y la escalabilidad del sistema.





3.2. DISEÑO DE LA BASE DE DATOS

Se ha realizado un diseño detallado de la base de datos para asegurar una gestión eficiente y coherente de la información.

La base de datos está estructurada en múltiples tablas interrelacionadas que reflejan las distintas entidades del CRM.

A continuación, se describen los principales aspectos del diseño:

A. Modelo Entidad-Relación (ER):

- Se ha elaborado un diagrama ER que representa las entidades principales del sistema (*Identificación, Clientes, Oportunidades, Presupuestos, Productos, Perfil, Tareas, Eventos, FAQ*) y sus relaciones.
- Este modelo ha sido fundamental para definir las tablas, sus atributos y las claves primarias y foráneas que garantizan la integridad referencial.

B. Descripción de las Tablas y Relaciones:

- **IDENTIFICACIÓN:** Almacena la información básica de las empresas registradas, incluyendo nombre, correo y contraseña.
- **CLIENTES:** Gestiona los datos de los clientes, relacionados con las oportunidades y presupuestos.
- **OPORTUNIDADES:** Registra las oportunidades de negocio vinculadas a los clientes y presupuestos.
- **PRESUPUESTOS:** Almacena los presupuestos creados para los clientes, incluyendo fechas y montos.
- **PRODUCTOS:** Gestiona el inventario de productos disponibles, con detalles como proveedor, descripción, precio y stock.
- **PERFIL:** Almacena la información de perfil de los usuarios, permitiendo personalizaciones.
- **TAREAS:** Registra las tareas asignadas a los miembros de la empresa, con detalles de prioridad y estado.
- **EVENTOS:** Gestiona los eventos programados en el calendario empresarial.
- **FAQ:** Almacena las preguntas y respuestas frecuentes para soporte al usuario.





C. Gestión de la Base de Datos con DatabaseManager.py:

- Se ha implementado la clase `DatabaseManager` para manejar todas las interacciones con la base de datos.
- Esta clase incluye métodos específicos para cada entidad, permitiendo operaciones CRUD y garantizando la integridad de los datos mediante el uso de claves foráneas y validaciones.

```
database_manager.py
1  import sqlite3
2  import os
3
4  class DatabaseManager:
5      def __init__(self, db_name):
6          self.db_name = db_name
7          self.connection = None
8          self.cursor = None
9
10     def connect(self):
11         if not os.path.exists(self.db_name):
12             print("La base de datos no existe, creando la base de datos y tablas...")
13             self.create_db()
14         else:
15             self.connection = sqlite3.connect(self.db_name)
16             self.cursor = self.connection.cursor()
17
18             self.check_tables()
19
20     def create_db(self):
21         self.connection = sqlite3.connect(self.db_name)
22         self.cursor = self.connection.cursor()
23
24         print("Creando las tablas...")
25
26         # =====
27         # Tabla IDENTIFICACION
28         # =====
29         self.cursor.execute("""
30             CREATE TABLE IF NOT EXISTS IDENTIFICACION (
31                 NOMBRE_EMPRESA TEXT NOT NULL,
32                 MAIL TEXT NOT NULL,
33                 PASSWORD TEXT NOT NULL
34             )
35         """)
```





3.3. DISEÑO DE LA INTERFAZ DE USUARIO

Se ha diseñado una interfaz de usuario **intuitiva y amigable**, enfocada en mejorar la experiencia del usuario y facilitar la navegación a través de las distintas funcionalidades del CRM. Los principales aspectos del diseño de la interfaz son:

A. Pantalla de Bienvenida:

- Se ha creado una pantalla de bienvenida que ofrece opciones claras para *iniciar sesión* o *registrar* una nueva empresa.
- Esta pantalla incluye campos para el nombre de la empresa, correo electrónico y contraseña, asegurando una entrada de datos sencilla y directa.





B. Pantalla Principal (*main_window.py*):

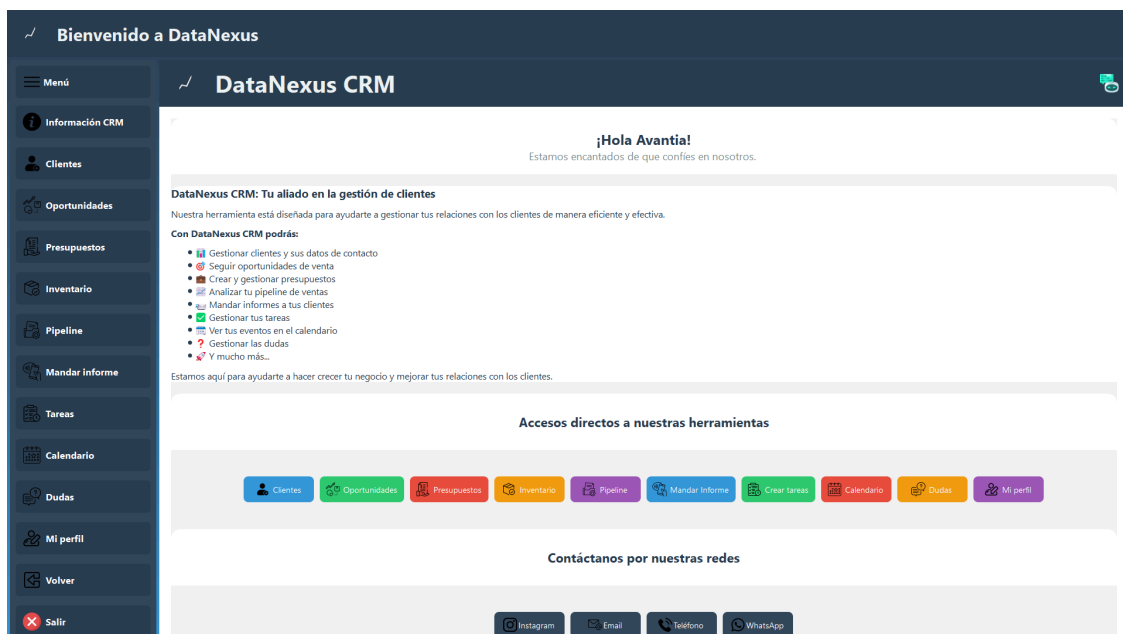
- La pantalla principal incorpora un **menú lateral** característico que puede ocultarse según las preferencias del usuario.
- Este menú incluye botones que llevan a las diferentes vistas del CRM, permitiendo una navegación rápida y eficiente.

C. Diseño Responsivo y Usabilidad:

- La interfaz ha sido diseñada para adaptarse a diferentes resoluciones de pantalla, garantizando una experiencia consistente en distintos dispositivos.
- Se ha priorizado la usabilidad mediante la disposición lógica de los elementos, el uso de colores contrastantes y la incorporación de iconos descriptivos.

D. Elementos Interactivos:

- Se han implementado elementos interactivos como botones, tablas dinámicas y formularios, que permiten a los usuarios interactuar fácilmente con el sistema.
- Además, se ha añadido un **asistente** accesible desde la interfaz para proporcionar información de las funcionalidades de cada pantalla.





3.4. DISEÑO DE COMPONENTES ESPECÍFICOS

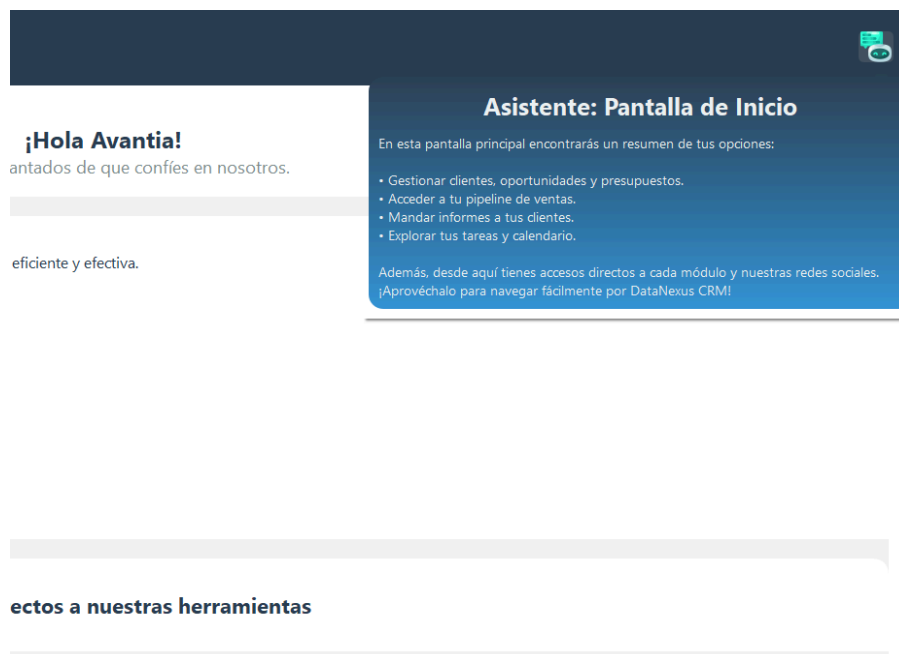
Se han diseñado componentes específicos que aportan funcionalidades adicionales y mejoran la experiencia del usuario en el CRM. A continuación, se detallan estos componentes:

3.4.1. Asistentes por Vista

Se ha creado una carpeta denominada **asistentes** que contiene asistentes específicos para cada vista del CRM, como ``bubujAsistente_cliente.py``, ``bubujAsistente_oportunidad.py``, entre otros.

Estos asistentes proporcionan orientación y soporte contextual al usuario, guiándolos a través de las funcionalidades disponibles en cada sección. Los principales aspectos del diseño de los asistentes son:

- **Personalización:** Cada asistente está diseñado para adaptarse a la funcionalidad específica de su vista correspondiente, ofreciendo consejos y explicaciones relevantes.
- **Interactividad:** Los asistentes son interactivos, permitiendo al usuario hacer preguntas y recibir respuestas en tiempo real sobre el uso de la interfaz.
- **Integración:** Se han integrado de manera fluida en la interfaz, apareciendo como burbujas de ayuda accesibles desde la vista correspondiente.





3.4.2. Elementos Visuales Innovadores (Rotable Label)

Se ha desarrollado la clase ``rotable_label.py``, que permite la rotación de **360°** del logo de la empresa en la interfaz, añadiendo un toque futurista y dinámico al diseño.

Los principales aspectos del diseño de este elemento son:

- **Animación Suave:** La rotación del logo se ha implementado con animaciones suaves que no afectan el rendimiento de la aplicación, proporcionando un efecto visual atractivo sin distraer al usuario.
- **Configurabilidad:** La clase permite ajustar la velocidad y la dirección de la rotación, ofreciendo flexibilidad para adaptarse a diferentes estilos de diseño.
- **Integración Visual:** El logo rotativo se ha posicionado estratégicamente en la interfaz para complementar otros elementos visuales sin interferir con la funcionalidad principal.





4. FASE 3: DESARROLLO

En la fase de desarrollo, se ha llevado a cabo la **implementación práctica del diseño** previamente establecido, transformando los planes y diagramas en un sistema funcional.

Esta fase ha abarcado desde la configuración del entorno de desarrollo hasta la integración y despliegue final del CRM.

A continuación, se detallan los principales aspectos desarrollados en esta fase:

4.1. CONFIGURACIÓN DEL ENTORNO DE DESARROLLO

Se ha configurado un entorno de desarrollo robusto y eficiente para facilitar la programación, pruebas y depuración del CRM.

Los pasos clave en esta configuración han sido:

1. Selección de Herramientas y Tecnologías:

- **Lenguaje de Programación:** Se ha utilizado Python, versatilidad y amplia gama de librerías disponibles para el desarrollo de aplicaciones de escritorio.
- **Entorno de Desarrollo Integrado:** Se ha optado por utilizar [Visual Studio Code](#) por su ligereza y extensibilidad mediante plugins.
- **Librerías y Frameworks:** Se han utilizado librerías como [PyQt5](#) para el desarrollo de la interfaz gráfica de usuario (*GUI*) y [sqlite3](#) para la gestión de la base de datos.

2. Estructura del Proyecto:

- Se ha organizado el proyecto en una estructura de carpetas clara y modular, separando los distintos componentes como vistas, iconos, asistentes y recursos visuales.

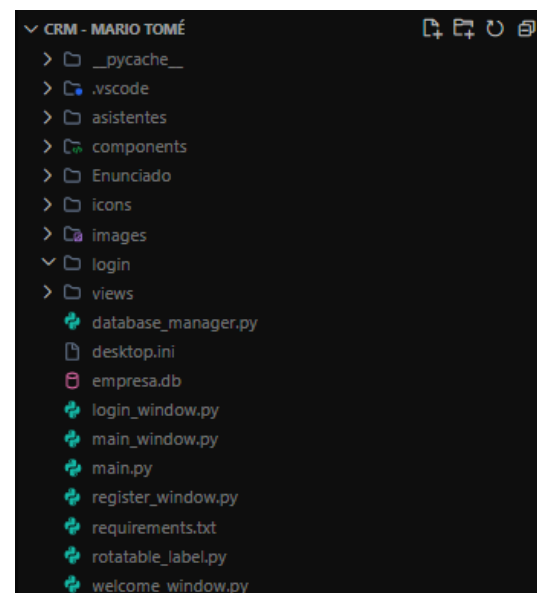
3. Configuración de Dependencias:

- Se ha creado un archivo `requirements.txt` que lista todas las dependencias necesarias para el proyecto, facilitando la instalación y gestión de librerías mediante **pip**.

4. Configuración del Entorno Virtual:

- Se ha establecido un entorno virtual utilizando **venv** para aislar las dependencias del proyecto y evitar conflictos con otras aplicaciones:

```
python -m venv env
source env\Scripts\activate
pip install -r requirements.txt
```





4.2. IMPLEMENTACIÓN DE LA BASE DE DATOS

Se ha procedido a la implementación de la base de datos siguiendo el diseño establecido en la fase de diseño. Los principales pasos han sido:

1. Creación de la Clase DatabaseManager:

- Se ha desarrollado la clase `DatabaseManager.py` encargada de manejar todas las operaciones relacionadas con la base de datos, incluyendo la conexión, creación de tablas y ejecución de consultas CRUD.

2. Definición de las Tablas:

- Se han implementado todas las tablas necesarias (`IDENTIFICACION`, `CLIENTES`, `OPORTUNIDADES`, `PRESUPUESTOS`, `PRODUCTOS`, `PERFIL`, `TAREAS`, `EVENTOS`, `FAQ`) con sus respectivas relaciones y restricciones de integridad referencial.

```
database_manager.py
36
37 # =====
38 # Tabla CLIENTES
39 # =====
40 self.cursor.execute("""
41     CREATE TABLE IF NOT EXISTS CLIENTES (
42         ID_CLIENTE TEXT PRIMARY KEY,
43         NOMBRE TEXT NOT NULL,
44         DIRECCION TEXT NOT NULL,
45         TELEFONO TEXT NOT NULL,
46         PERSONA_CONTACTO TEXT NOT NULL,
47         EMAIL TEXT NOT NULL
48     )
49 """)
50
51 # =====
52 # Tabla OPORTUNIDADES
53 # =====
54 self.cursor.execute("""
55     CREATE TABLE IF NOT EXISTS OPORTUNIDADES (
56         ID_OPORTUNIDAD TEXT PRIMARY KEY,
57         NOMBRE_OPORTUNIDAD TEXT NOT NULL,
58         CLIENTE TEXT NOT NULL,
59         FECHA DATE NOT NULL,
60         PRESUPUESTO TEXT NOT NULL,
61         INGRESO_ESPERADO REAL NOT NULL,
62         ESTADO TEXT NOT NULL,
63         FOREIGN KEY (CLIENTE) REFERENCES CLIENTES(ID_CLIENTE)
64     )
65 """)
66
67 # =====
68 # Tabla PRESUPUESTOS
69 # =====
70 self.cursor.execute("""
71     CREATE TABLE IF NOT EXISTS PRESUPUESTOS (
72         ID_PRESUPUESTO TEXT PRIMARY KEY,
73         NOMBRE TEXT NOT NULL,
74         CLIENTE TEXT NOT NULL,
75         FECHA_CREACION DATE NOT NULL,
76         FECHA_EXPIRACION DATE NOT NULL,
77         SUBTOTAL REAL NOT NULL,
78         TOTAL REAL NOT NULL,
79         FOREIGN KEY (CLIENTE) REFERENCES CLIENTES(ID_CLIENTE)
80     )
81 """)
```



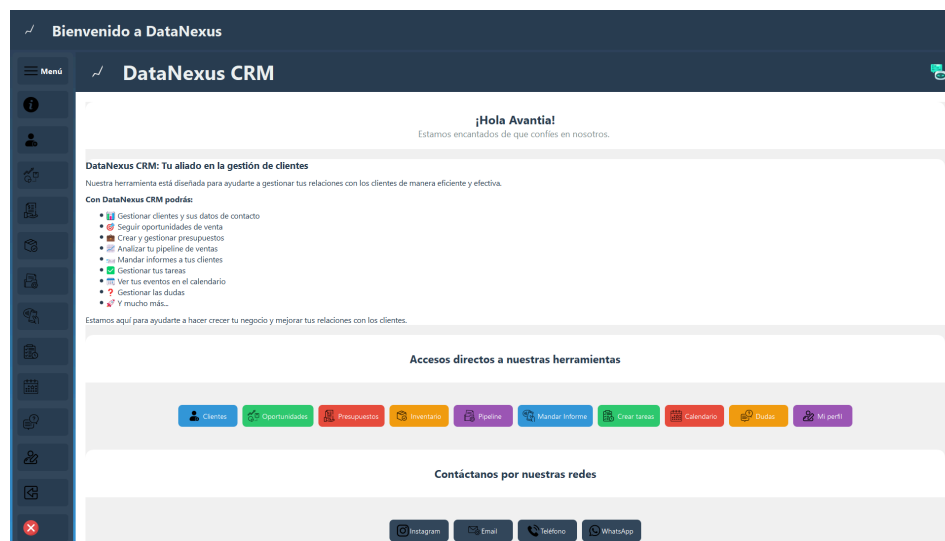
4.3. DESARROLLO DE LAS VISTAS PRINCIPALES

Se han desarrollado todas las vistas principales del CRM, asegurando una interfaz de usuario intuitiva y funcional.

Cada vista ha sido implementada siguiendo los principios de diseño establecidos, integrando funcionalidades específicas y garantizando una experiencia de usuario coherente.

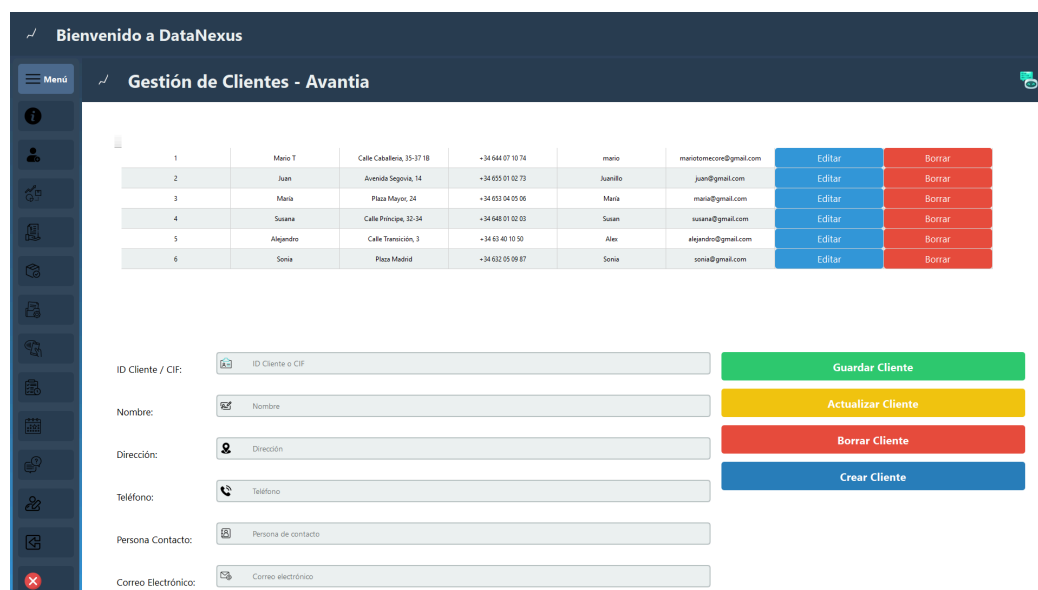
1. Información del CRM ('infoCRM_view.py'):

- Se ha implementado una vista que muestra un saludo personalizado, accesos directos a funciones clave y enlaces a redes sociales.



2. Clientes ('clientes_view.py'):

- Se ha creado una interfaz para la gestión de clientes, incluyendo formularios para añadir, editar y eliminar clientes, y una tabla dinámica que muestra la lista de clientes.





3. Oportunidades (`oportunidades_view.py`):

- Se ha desarrollado una vista para gestionar oportunidades de negocio, permitiendo asociarlas a clientes y presupuestos, y actualizar su estado.

Bienvenido a DataNexus

Menú

Gestión de Oportunidades - Avantia

1	Venta de iPhones	2	2025-01-27	1	2000.0	En proceso	Editar	Borrar
2	Venta monitores HP	4	2025-01-27	2	2500.0	En proceso	Editar	Borrar
3	Venta cargadores iPhone	3	2025-01-27	3	120.0	Ganada	Editar	Borrar
4	Compra teclados oficina	2	2025-01-27	4	150.0	Perdida	Editar	Borrar
5	Venta AirPods	1	2025-01-27	5	1000.0	Ganada	Editar	Borrar

ID Oportunidad: Guardar Oportunidad

Nombre Oportunidad: Actualizar Oportunidad

Cliente: Borrar Oportunidad

Fecha: Limpiar formulario

Presupuesto:

Ingreso Esperado:

Estado:

4. Presupuestos (`presupuestos_view.py`):

- Se ha implementado una interfaz para la creación y modificación de presupuestos, incluyendo campos para fechas y montos.

Bienvenido a DataNexus

Menú

Gestión de Presupuestos - Avantia

1	venta iphones	Mario	2025-01-24	2025-02-23	200.0	220.0	Editar
2	Venta monitores HP	Sonia	2025-01-27	2025-02-26	400.0	420.0	Editar
3	Venta cargadores iPhone Lightning	5	2025-01-27	2025-02-26	20.0	21.5	Editar
4	Compra teclados	6	2025-01-27	2025-02-26	150.0	150.5	Editar
5	Venta AirPods	3	2025-01-27	2025-02-26	200.0	220.0	Editar

ID Presupuesto: Guardar Presupuesto

Nombre: Actualizar Presupuesto

Cliente: Borrar Presupuesto

Fecha Creación: Ver Inventario

Fecha Expiración:

Subtotal:

Total:

LIMPIAR FORMULARIO



5. Inventario (`inventario_view.py`):

- Se ha creado una vista para la gestión del inventario de productos, con funcionalidades para añadir, editar y eliminar productos.

Bienvenido a DataNexus

Menú Gestión de Inventario - Avantia

1	Apple	iPhone 16 Pro Max	Smartphone con chip A15 y pantalla Super Retina XDR.	21	1000.0	20
2	MediaMarkt	Cargador iPhone lightning	Pack de dos cargadores iPhone tipo lightning. Cable trenzado	21	20.0	100
3	Samsung	Galaxy S21	Smartphone con cámara de 64MP y 120Hz.	21	799.0	50
4	Apple	iPhone 14	Smartphone con chip A15 y pantalla Super Retina XDR.	21	999.0	30
5	Sony	WH-1000XM4	Auriculares inalámbricos con cancelación de ruido.	21	350.0	75
6	LG	OLED55CX6LA	Televisor OLED de 55" con 4K HDR.	21	1499.0	20
7	Microsoft	Surface Pro 7	Tableta convertible con teclado extraíble y Windows 10.	21	899.0	40
8	Xiaomi	MI Band 6	Pulsera de actividad con monitor de frecuencia cardíaca.	21	45.0	100
9	Canon	EOS R6	Cámara sin espejo de fotograma completo para fotógrafos profesionales.	21	2499.0	10
10	Bose	SoundLink Revolve+	Altavoz Bluetooth portátil con sonido 360°	21	229.0	60
11	FitBit	Versa 3	Reloj inteligente con GPS y seguimiento de la actividad.	21	229.0	70
12	Dell	XPS 13	Portátil ultradelgado con pantalla InfinityEdge.	21	1299.0	15

Añadir Producto Editar Producto Actualizar

6. Mandar Informe (`mandarInform_view.py`):

- Se ha desarrollado una vista que automatiza el envío de informes por WhatsApp o correo electrónico, utilizando los datos de los clientes.

Bienvenido a DataNexus

Menú Enviar Informe - Avantia

Ciente: 1

Enviar Informe por WhatsApp

Número: +34 644 07 10 74

Mensaje:

Hola Mario T,

Te enviamos este informe de cliente.
Dirección: Calle Caballería, 35-37 1B
Persona contacto: mario

Saludos,
Avantia

Enviar por WhatsApp

Enviar Informe por Email

Para: maricomecore@gmail.com

Asunto: Informe para Mario T

Mensaje:

Hola Mario T,

Te enviamos este informe de cliente.
Dirección: Calle Caballería, 35-37 1B
Persona contacto: mario

Saludos,
Avantia

Enviar por Email



7. Tareas (`tareas_view.py`):

- Se ha implementado una interfaz para la creación y administración de tareas asignadas, con detalles sobre prioridad y estado.

Bienvenido a DataNexus

Menú Gestión de Tareas

0900987d Revisar Stock de iPhones Ir al apartado de stock par... 2025-01-27 2025-02-14 Juan Alta Pendiente

ID Tarea: ID de la Tarea (autogenerado)

Título: Título breve de la tarea

Descripción: Descripción más detallada

Creación: 2025-01-29

Vencimiento: 2025-01-01

Asignado a: Persona responsable o equipo

Prioridad: Baja

Estado: Pendiente

Crear Tarea Actualizar Tarea Borrar Tarea Limpiar Formulario

8. Calendario (`calendario_view.py`):

- Se ha creado un calendario interactivo para la programación y gestión de eventos empresariales.

Bienvenido a DataNexus

Menú Calendario de Eventos

enero 2025

	lu.	ma.	mi.	ju.	vi.	sá.	do.
1	30	31	1	2	3	4	5
2	6	7	8	9	10	11	12
3	13	14	15	16	17	18	19
4	20	21	22	23	24	25	26
5	27	28	29	30	31	1	2
6	3	4	5	6	7	8	9

ID Evento: ID del evento (auto)

Título: Título del evento

Hora: 09:43

Lugar: Lugar (sala, oficina, etc.)

Descripción: Descripción u observaciones

Asignado a: Asignado a...

Crear Evento Actualizar Evento Borrar Evento Limpiar



9. Dudas (`dudas_view.py`):

- Se ha desarrollado una sección de preguntas y respuestas frecuentes (FAQ) con funcionalidades de búsqueda y filtrado.

Bienvenido a DataNexus

Menú Preguntas Frecuentes (FAQ)

Buscar pregunta / respuesta... Buscar

ID FAQ: ID de la FAQ (auto)

Pregunta: Pregunta...

Respuesta: Respuesta detallada...

Categoría: General

Crear FAQ Actualizar FAQ Borrar FAQ Limpiar

10. Mi Perfil (`miPerfil_view.py`):

- Se ha implementado una interfaz para la personalización del perfil del usuario, incluyendo la subida de fotos y edición de datos personales.

Bienvenido a DataNexus

Menú Mi Perfil - Avantia

Cambiar Foto

Nombre: Nombre

Correo: Correo

Contraseña: Contraseña

Número de Teléfono: Número de Teléfono

Dirección: Dirección

Descripción: Descripción de la empresa / perfil...

Guardar Cambios Cancelar



4.4. DESARROLLO DE ASISTENTE

Se han desarrollado asistentes específicos para cada vista y un chatbot interactivo para mejorar la usabilidad y proporcionar soporte al usuario.

1. Asistentes por Vista:

- Se ha creado una carpeta `asistentes` que contiene scripts como `bubujAsistente_cliente.py`, encargados de proporcionar orientación y soporte contextual en cada vista.

4.5. IMPLEMENTACIÓN DE ELEMENTOS VISUALES

Se han implementado elementos visuales innovadores para mejorar la estética y la experiencia del usuario en la interfaz del CRM.

Rotable Label (`rotable_label.py`):

- Se ha desarrollado la clase `rotable_label.py` que permite la rotación de 360° del logo de la empresa, aportando un aspecto futurista y dinámico a la interfaz.



```
import math
from PyQt5.QtWidgets import QLabel
from PyQt5.QtCore import pyqtProperty
from PyQt5.QtGui import QPainter, QPixmap

class RotatableLabel(QLabel):
    def __init__(self, parent=None):
        super().__init__(parent)
        self._rotation = 0
        self._pixmap = None

    def setPixmap(self, pixmap):
        super().setPixmap(pixmap)
        self._pixmap = pixmap

    @pyqtProperty(float)
    def rotation(self):
        return self._rotation

    @rotation.setter
    def rotation(self, angle):
        self._rotation = angle
        self.update()

    def paintEvent(self, event):
        if not self._pixmap:
            super().paintEvent(event)
            return

        painter = QPainter(self)
        painter.setRenderHint(QPainter.Antialiasing, True)
        painter.setRenderHint(QPainter.SmoothPixmapTransform, True)

        w = self.width()
        h = self.height()

        # Traducción para centrar
        painter.translate(w / 2, h / 2)

        # Convertir grados a radianes
        angle_radians = math.radians(self._rotation)

        # Efecto flip horizontal:
        scale_x = math.cos(angle_radians)
        painter.scale(scale_x, 1.0)

        pw = self._pixmap.width()
        ph = self._pixmap.height()
        painter.translate(-pw / 2, -ph / 2)

        painter.drawPixmap(0, 0, self._pixmap)
```




5. CONCLUSIONES

5.1. RESUMEN DE LOGROS

Se ha desarrollado e implementado un CRM completo y funcional, diseñado para optimizar la gestión empresarial y mejorar las relaciones con los clientes.

Los principales logros alcanzados en este proyecto incluyen:

Funcionalidades Integrales:

- Se ha creado un sistema que abarca diversas áreas de gestión empresarial, incluyendo la administración de clientes, oportunidades de negocio, presupuestos, inventario, tareas, calendarios, preguntas frecuentes (FAQ) y perfiles de usuario.
- Cada módulo ha sido desarrollado con funcionalidades específicas que permiten realizar operaciones CRUD (*Crear, Leer, Actualizar, Eliminar*) de manera eficiente.

Interfaz de Usuario Intuitiva:

- Se ha diseñado una interfaz de usuario (UI) amigable y fácil de navegar, con un menú lateral característico que puede ocultarse según las preferencias del usuario.
- La disposición lógica de los elementos y el uso de colores contrastantes han contribuido a una experiencia de usuario coherente y agradable.

Elementos Visuales Innovadores:

- La implementación de la clase `rotatable_label.py` ha permitido la rotación de 360° del logo de la empresa, añadiendo un toque futurista y dinámico a la interfaz.
- Este elemento visual distintivo mejora la estética del CRM y proporciona una experiencia visual atractiva para los usuarios.

Asistentes Personalizados y Chatbot:

- Se ha incorporado una carpeta `asistentes` que contiene asistentes específicos para cada vista del CRM, proporcionando orientación contextual y soporte al usuario.

Gestión Eficiente de la Base de Datos:

- La clase `DatabaseManager` ha facilitado una gestión efectiva de la base de datos SQLite, garantizando la integridad y accesibilidad de los datos mediante la implementación de claves foráneas y validaciones en las entradas de datos.
- Se han creado todas las tablas necesarias con sus respectivas relaciones, permitiendo una manipulación de datos coherente y segura.

Pruebas y Validación Exhaustivas:

- Se han realizado diversas pruebas unitarias, de integración y de usabilidad para asegurar que el CRM funciona correctamente y cumple con los requisitos establecidos.
- Las pruebas han permitido identificar y corregir errores, optimizando el rendimiento y la estabilidad del sistema.





5.2. RETOS Y SOLUCIONES

Durante el desarrollo del CRM, se han enfrentado diversos retos que han requerido soluciones innovadoras y adaptativas.

A continuación, se describen los principales desafíos y cómo se han abordado:

1. Escalabilidad de la Base de Datos:

- **Reto:** La utilización de SQLite como sistema de gestión de bases de datos es adecuada para aplicaciones de tamaño medio, pero puede presentar limitaciones en términos de escalabilidad para empresas con un gran volumen de datos.
- **Solución:** Se ha optimizado la estructura de las tablas y las consultas para mejorar el rendimiento actual.

2. Optimización de Elementos Visuales:

- **Reto:** La implementación de la rotación del logo mediante la clase ``rotable_label.py`` podía afectar el rendimiento de la aplicación si no se gestionaba adecuadamente.
- **Solución:** Se ha optimizado el código de la clase para asegurar que las animaciones de rotación se realicen de manera suave y eficiente, sin comprometer el rendimiento general del sistema. Se han realizado pruebas de rendimiento para ajustar la velocidad y la frecuencia de rotación.

3. Seguridad de los Datos:

- **Reto:** Garantizar la seguridad de las credenciales de los usuarios y la integridad de los datos almacenados en la base de datos era crucial para proteger la información sensible.
- **Solución:** Se han implementado medidas básicas de seguridad, como la encriptación de contraseñas y la utilización de claves foráneas para mantener la integridad referencial.

4. Integración de Componentes Diversos:

- **Reto:** Asegurar que todos los componentes desarrollados, como las vistas principales, los asistentes, el chatbot y los elementos visuales, funcionen de manera cohesiva dentro de la aplicación.
- **Solución:** Se ha adoptado una arquitectura basada en el patrón Modelo-Vista-Controlador que separa claramente las responsabilidades entre los componentes, facilitando la integración y reduciendo el acoplamiento.





5.3. RECOMENDACIONES FUTURAS

Para potenciar aún más las capacidades del CRM desarrollado y abordar las limitaciones actuales, se proponen las siguientes recomendaciones para futuras expansiones y mejoras:

1. Migración a un Sistema de Gestión de Bases de Datos Más Escalable:

- **Recomendación:** Considerar la migración de SQLite a sistemas de bases de datos más robustos como *PostgreSQL* o *MySQL* para soportar un mayor volumen de datos y mejorar la escalabilidad.
- **Justificación:** Esto permitirá al CRM adaptarse al crecimiento de la empresa y manejar eficientemente un mayor número de transacciones y registros.

2. Implementación de Funcionalidades Avanzadas:

- **Recomendación:** Añadir funcionalidades avanzadas como análisis predictivo, generación de informes detallados, integración con plataformas externas y automatización de procesos empresariales.
- **Justificación:** Estas mejoras proporcionarían un valor añadido significativo, permitiendo a las empresas tomar decisiones más informadas y optimizar sus operaciones.

3. Mejoras en la Seguridad de los Datos:

- **Recomendación:** Incorporar medidas de seguridad más avanzadas, como el cifrado de datos en tránsito y en reposo, autenticación multifactor y auditorías de seguridad periódicas.
- **Justificación:** Aumentar la seguridad protegerá mejor la información sensible de la empresa y de sus clientes, cumpliendo con estándares de seguridad más exigentes.

4. Desarrollo de una Versión Móvil del CRM:

- **Recomendación:** Crear aplicaciones móviles para plataformas iOS y Android, permitiendo el acceso al CRM desde dispositivos móviles.
- **Justificación:** La accesibilidad móvil ampliará las posibilidades de uso del CRM, facilitando el acceso a la información y la gestión de tareas en cualquier momento y lugar.

5. Soporte Multilenguaje:

- **Recomendación:** Implementar soporte para múltiples idiomas, permitiendo que el CRM sea utilizado en entornos multilingües o por equipos internacionales.
- **Justificación:** La localización del CRM aumentará su atractivo en mercados internacionales y facilitará su adopción por parte de empresas con equipos diversificados.

En resumen, estas recomendaciones buscan ampliar las capacidades del CRM, mejorar su rendimiento y seguridad, y adaptarlo a las necesidades cambiantes de las empresas. La implementación de estas sugerencias permitirá que el CRM evolucione y siga siendo una herramienta valiosa para la gestión empresarial en el futuro.

