

Integrantes:

- Daniel Ceballos Monsalve
- Jose Santiago Perez Piedrahita
- Miguel Angel Restrepo Tangarife
- Carlos Daniel Urresty Ascuntar

[Link del repositorio](#)

Descripción general de la solución:

La **Gestión de las Relaciones con Clientes (CRM)**, como el término es conocido en español, va más allá de una plataforma o un software: es todo el proceso utilizado por startups, pequeñas y grandes empresas para **administrar y analizar las interacciones con clientes**, anticipar necesidades y deseos, optimizar la rentabilidad, aumentar las ventas y personalizar campañas de captación de nuevos clientes.

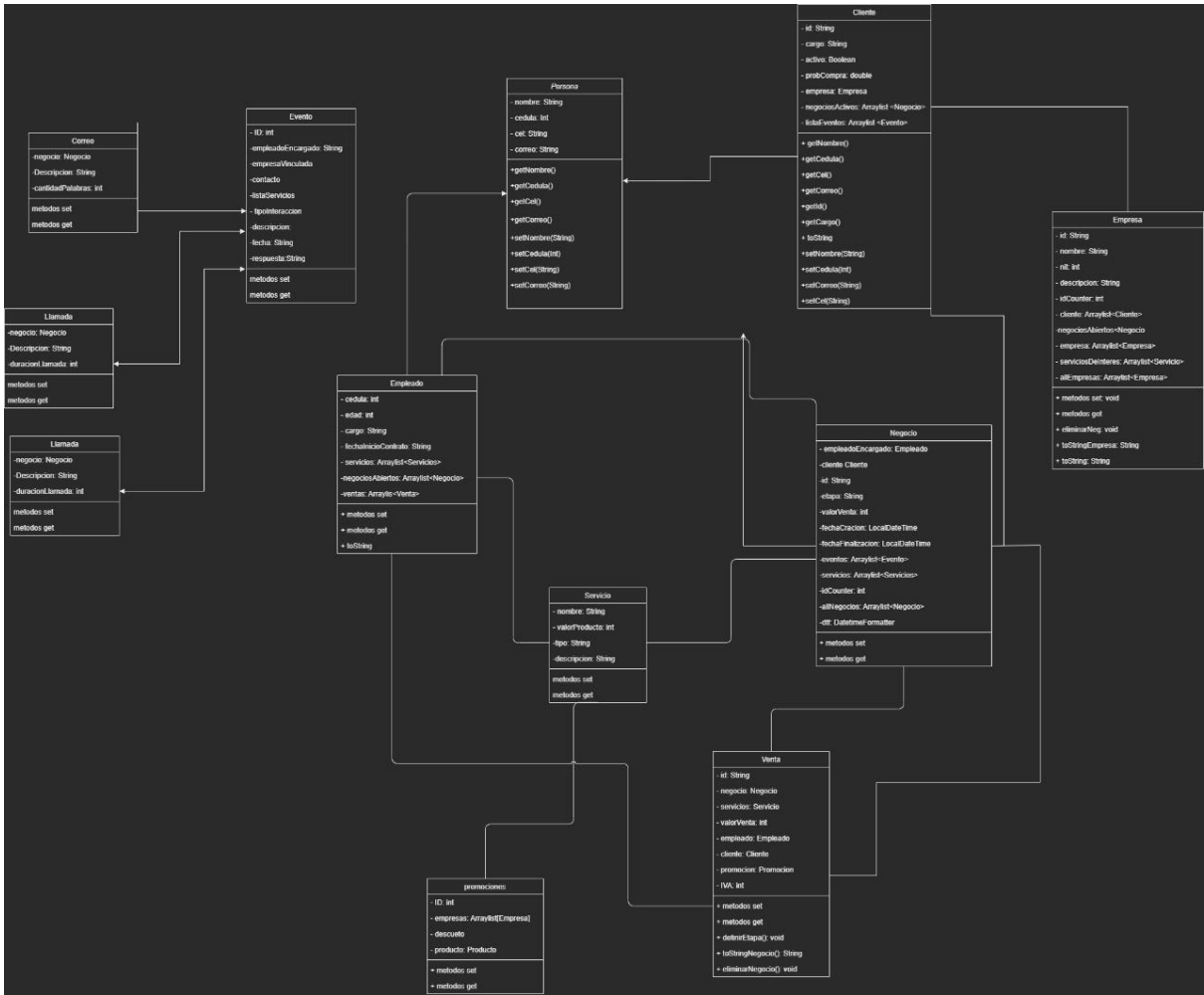
El software de CRM registra la información de contacto de los clientes, tales como el correo electrónico, el teléfono, el sitio web, el perfil de redes sociales, etc. También puede introducir otra información, como las noticias recientes sobre la actividad de la empresa, además de almacenar detalles tales como las preferencias personales de un cliente respecto a las comunicaciones. Estamos en la era del cliente, de la transformación digital, de las nuevas tecnologías. En ese escenario, **la relación también evoluciona**, llevando a un nuevo concepto también conocido como **experiencia del cliente**. El concepto de CRM implica estar centrado en el cliente. Es estrategia, es un proceso, es herramienta y tecnología.

- **CRM como tecnología:** este es un producto de tecnología, a menudo en la nube, que los equipos utilizan para registrar y analizar las interacciones entre la empresa y los usuarios, y generar informes sobre ellas. A esto también se le llama una solución o un [sistema de CRM](#).
- **CRM como estrategia:** se trata de una filosofía de empresa sobre cómo deben gestionarse las relaciones con los clientes y los clientes potenciales.
- **CRM como proceso:** piense en este concepto como un sistema que una empresa adopta para cuidar y gestionar las relaciones.

Nuestro proyecto: El proyecto está orientado a la creación de un sistema con el cual el usuario (vendedor) podrá almacenar los posibles clientes, empresas, incrementar el engagement con los anteriores a través de el envío de correos, llamadas y reuniones.

Con el propósito de acotar el alcance, se decidió realizar la función de intermediario entre algún vendedor y las empresas que se encuentran en el sistema, buscando el mejor vendedor / comprador en cada caso.

UML



Implementación de las características Orientadas a Objetos:

- Clases Abstracta (1) y Métodos Abstractos (1): La clase Persona es una clase Abstracta, la cual posee diferentes métodos abstractos.
- Interfaces (1) diferentes a los utilizados para serializar los objetos en el punto de la persistencia. Deberá ser propio del dominio a implementar: La clase Tiempo es una Interfaz, la cual define los métodos que deben ser implementados en la clase Persona
- Herencia (1): La clase Evento, es una clase que hereda a las clases Correo, Llamada, Reunión. Esta clase hereda los métodos y atributos que deben tener las otras clases
- Ligadura dinámica (2) asociadas al modelo lógico de la aplicación: La clase Negocio usa ligadura dinámica, para decidir el método a usar en según el tipo de Evento.
- Atributos de clase (1) y métodos de clase (1): La mayoría de clases tienen atributos de clase, con los cuales se almacena las instancias creadas según clase. Por lo tanto, se crean los métodos de clase para acceder a ellos.
- Uso de constante (1 caso): En la clase Tiempo, la cual es una interfaz, posee atributos, los cuales son constantes por defectos:
- Encapsulamiento (private, protected y public): Usado en todos los atributos y metodos
- Los siguientes conceptos asociados a la POO:
 - Sobrecarga de métodos (1 casos mínimo) y constructores (2 casos mínimo):
 - Manejo de referencias this para desambiguar y this() entre otras. 2 casos mínimo para cada caso: Todos los métodos *get* usan la referencia this, para desambiguar los atributos de instancias de otros parámetros