**4.TRANSICIÓN DE LAS IDEAS A LOS DISEÑOS PRELIMINARES**

Las ideas presentadas en la fase de búsqueda de soluciones creativas fueron las siguientes:

**Para análisis de datos:**

1. Técnicas de análisis tradicionales (estadística)
2. Análisis de Asociación
3. Algoritmo de Clústeres de Microsoft
4. Árboles de Decisión
5. Redes Neuronales

**Para bases de datos:**

1. Bases de datos relacionales
2. Bases de datos columnares
3. Bases de datos gráficas
4. Key-Value
5. Archivos CSV

**Para consulta a las bases de datos (Lenguajes):**

1. LinQ sobre estructuras lineales
2. SQL
3. .QL
4. QUEL
5. LinQ sobre bases de datos

De acuerdo con estas ideas, se decidió desarrollar un poco más las siguientes ya que son más prometedoras y aplicables al contexto que estamos manejando:

**ANÁLISIS DE DATOS.**

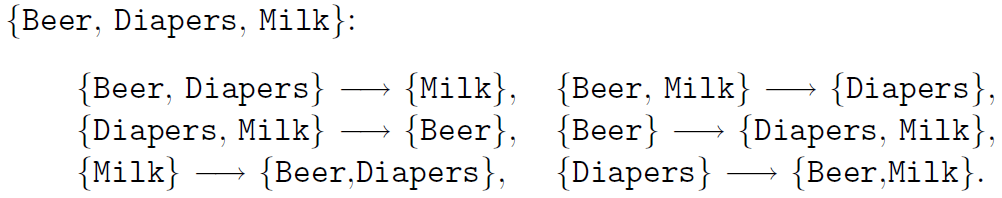
Se descartó seguir desarrollando la idea (1) sobre el uso de técnicas tradicionales de análisis estadístico ya que aparentemente dentro de las soluciones que ha buscado el cliente se han implementado y no han dado solución al problema, el cliente no busca información descriptiva de sus datos, ni valorar una hipótesis, el necesita que se creen dichas hipótesis y con ellas predecir relaciones que permitan tomar decisiones de marketing y ventas por lo que la estadística descriptiva e inferencial se sale del foco del problema. Tampoco parece factible la opción (4) sobre árboles de decisión ya que no se está buscando una solución a una pregunta específica de la cual se conocen los posibles caminos a seguir por lo que la hace una solución que no resolvería el problema planteado por Allers.

**Diseños Preliminares**

**Análisis de Asociación:** Ya que la empresa tiene la información de sus transacciones y sus artículos archivada, es posible analizar dichos datos mediante técnicas de análisis de asociación que permitirán crear predicciones de acuerdo con su historial de ventas que permitan a la empresa tener una mejor idea de cuál es la relación entre sus artículos y de cómo puede usar esta información para llegar de una manera más personalizada al cliente, creando en él un sentido de ser entendido por la compañía.

Usando los artículos se pueden crear conjuntos de artículos (itemsets), de acuerdo a su frecuencia de aparición en las ventas podemos usar dichos conjuntos frecuentes para formar reglas del tipo x→y, donde x es un conjunto de artículos que implica la posible compra del conjunto de artículos y.

1 Ejemplo itemset y reglas derivadas

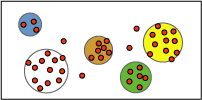


Como puede verse en la imagen de un solo conjunto de artículos pueden derivarse muchas reglas por lo que es importante realizar un proceso de filtrado basándonos en derivar reglas solamente de los conjuntos de artículos más frecuentes.

Luego del proceso de creación de reglas se pueden encontrar las reglas más confiables, es decir, que reglas pueden aplicarse más frecuentemente dentro de un conjunto de datos, en este caso las transacciones, aquellas más confiables son las que proporcionarán a la empresa información extremadamente valiosa, que permitirá tomar decisiones sobre las ventas, publicidad, ofertas, etc.

**Algoritmo de Clústeres:** Usando la información que Allers posee sobre sus clientes (Código, Nombre, Ciudad, Departamento, Método de pago) podemos usar este algoritmo para crear clústeres y grupos de clientes asociados de acuerdo con sus características.

2 Ejemplo de clustering

Los puntos azul, blanco, café, verde y amarillo son clústeres o agrupaciones y los puntos rojos en este contexto representarán a los clientes, luego de cargar la base de datos se procede a la creación de dichos clústeres, grupos que contienen elementos con un grado de similitud alta entre ellos y un grado de similitud baja comparados con los demás clústeres. Cada cliente tiene asociada probabilidad de pertenecer a los diferentes clústeres por lo que es muy útil para la empresa al momento de abordar a los clientes, ya que de antemano se los puede clasificar en un perfil de comportamiento con el cual la empresa puede tomar medidas más asertivas al momento de tratar con ellos. Es importante el hecho de que esta estructura de agrupamientos es dinámica ya que cada vez crea nuevos clústeres de acuerdo a las características, pero también desecha aquellos cuya población sea muy baja, reubicando sus elementos en otros grupos o creando un grupo nuevo que los asocie por características diferentes.

**Redes Neuronales:** Ya que sabemos que una de las dificultades de Allers es que se queda el inventario y se pierde con el tiempo por lo que esta solución podría ser de gran importancia para la empresa. Usando redes neuronales es posible entrenar un sistema de neuronas que predigan la cantidad de ventas e ingresos que se tendrán de acuerdo con una cantidad comprada y a que costo.

3 Ejemplo red neuronal



En nuestro contexto las variables X representarían los costos de ventas y la cantidad de ventas ocurridas en el pasado, dependiendo de la importancia de una u otra variable se le asigna un peso que tendrá repercusión en el resultado que en nuestro contexto serán las ventas. Con estos datos de “entrenamiento” las neuronas tendrán una base de datos de respaldo para buscar en el caso en que se le provean una cantidad de compra y su costo y poder llegar a una predicción de ventas. Por ejemplo: Entradas: Costo=100$, Cantidad=2, con estos datos mis neuronas van a comprar estas entradas con toda su base de datos de entrenamiento buscando una combinación de costo y cantidad que sea compatible con las entradas dadas encontrando un resultado asociado como predicción de ventas.

**LENGUAJE DE CONSULTA.**

Se decidió no desarrollar más a fondo las ideas (2), (3) y (4) ya que son lenguajes desconocidos para nosotros lo que representaría la necesidad de investigar y aprender este nuevo lenguaje desde el concepto más básico lo que requiere una cantidad de tiempo considerable, lo que dificultaría seguir el cronograma de entregas para el proyecto.

Se decide entonces seguir con las propuestas 1 y 5 de usar el lenguaje integrado de consulta LinQ para estructuras de datos lineales como listas, ya que simplifica el código para manejarlas y sobre bases de datos relacionales ya que permite hacer consultas complejas a la base de datos con relativa facilidad ya que tenemos un conocimiento previo sobre el lenguaje además que ya lo hemos puesto en práctica en otros proyectos. Al ser un lenguaje de consulta nativo de .NET hace mucho más fácil su uso ya que la aplicación será desarrollada en visual estudio con dicho framework, con este lenguaje podremos hacer uso de operadores de consulta estándar que si se necesitaran para este proyecto tendrían que ser implementados explícitamente por el programador por lo que sería un gasto de tiempo extra que no tendría beneficio alguno, con LinQ el uso de expresiones lambda (predicados lógicos) hace que se sintetice una gran parte del código permitiendo a los programadores enfocarse en el problema como tal, mas no en el uso del lenguaje.

**BASES DE DATOS.**

Se optó por no desarrollar más a fondo las ideas (2), (3) y (4). Las bases de datos columnares, aunque mejoran los tiempos de consulta requieren volver a cagar los datos cada vez que estos sean modificados, además que para usar la estructura columnar se debería cambiar la estructura de los datos proporcionados por Allers ya que están configurados en forma de filas. Las bases de datos gráficas prestan servicios muy especializados para el contexto que se maneja en este proyecto por lo que no es provechoso su uso además que requiere de contratar con externos para que presten el servicio. Por último, las bases de datos Key-Value al ser no estructuradas son usadas para almacenar cantidades de datos muy grandes (millones de millones) por lo que para la cantidad de datos usadas en este proyecto hace innecesario su uso, sólo migrar todos los datos hace de esta opción una carga de trabajo innecesaria con un beneficio casi nulo.

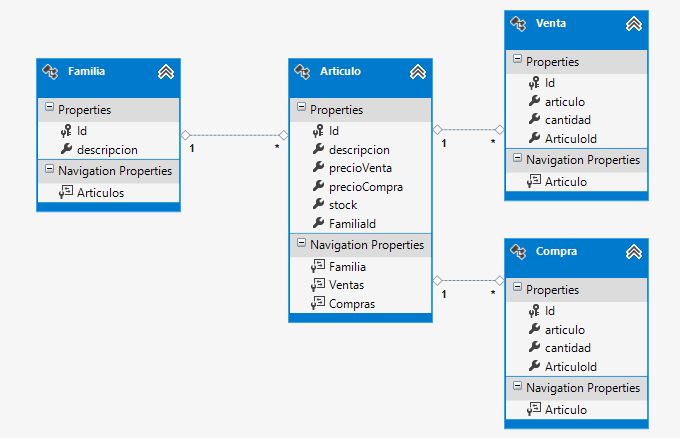
**Diseños Preliminares**

**Archivos CSV:** Con la información proporcionada por Allers de clientes, artículos y transacciones pueden generarse archivos de valores separados por comas dentro de un archivo de texto plano en los cuales se define un atributo como cada campo separado.

De esta forma los datos pueden ser leídos y modificados fácilmente mediante un método de lectura de archivos de texto plano que permita obtener la información y crear un objeto acorde con la información dada. Estos objetos creados posteriormente serían almacenados en estructuras de datos lineales fuertemente tipadas, es decir, se crearán listas que contengan únicamente objetos del mismo tipo por lo que habrá listas de objetos Cliente, Articulo, Transacción, etc.

**Bases de Datos Relacionales:** De acuerdo a la información proporcionada existe la posibilidad de crear un modelo de entidades, que contenga los campos de información para Clientes, Artículos, Transacciones, etc. Este tipo de base de datos me permite reconocer cada objeto de una entidad de acuerdo a una clave única lo que ayuda a que no haya inconsistencias en la información.

4 Ejemplo modelo de entidades



Con el modelo resultante podemos crear una base de datos relacional que nos permita almacenar la información de Allers mediante una sola carga de la información y así disminuir los tiempos de ejecución y cómputo ya que después de que se llenan las tablas de la base de datos relacional sólo se necesita crear un flujo hacia ella para acceder a su información.