**Método de la ingeniería**

**Identificación del problema.**

***Descripción del problema.***

En el grupo de clase de Modelado y Sistemas de Información está conformado por una cantidad de estudiantes. Adicionalmente los estudiantes cuentan con materias matriculadas en el semestre.

Teniendo en cuenta lo anterior, el problema que esto representa al maestro del curso, es el cómo manejar toda esta información incluyendo las materias matriculadas por estudiante. Esto para poder tener de forma ordenada los datos de los estudiantes.

Seguidamente y, para respaldar que los datos se encuentren correctos, es fundamental la información que fue recogida en un archivo csv. Para que el maestro del curso pueda manejar de manera eficiente y correcta la información.

***Definición del problema.***

Debido a la gran cantidad de datos e información acerca de las materias de los estudiantes, el maestro de Modelado y Sistemas de Información sabiendo que ustedes cuentan con un gran conocimiento acerca de programación orientada a objetos, le pide que desarrolle una agenda donde se puedan guardar de forma eficiente y ordenada los datos e información de el estudiante y materias respectivamente.

***Requerimientos funcionales.***

1. Añadir un contacto a la agenda
2. Eliminar un contacto de la agenda
3. Modificar un contacto de la agenda
4. Buscar un estudiante.
5. Verificar si el contacto existe.
6. Mostrar la información guardada.
7. Actualizar cambios en la información de un estudiante.
8. Crear y eliminar una materia.
9. Actualizar información de una materia matriculada.
10. Mostrar la información de las materias matriculadas
11. Mostrar el promedio de materias matriculadas.
12. Mostrar la materia con menos estudiantes matriculados
13. Mostrar la materia con más estudiantes matriculados.

***Requerimientos no funcionales.***

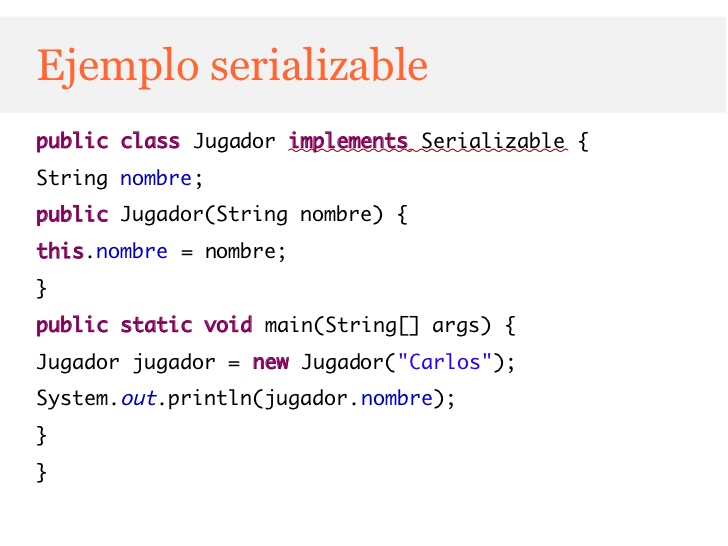
1. El programa debe estar implementado en java.
2. El programa debe contar con una interfaz gráfica.

**Recolección de información necesaria.**

Con el objetivo de obtener una claridad absoluta de los conceptos involucrados, es necesaria cierta investigación sobre las definiciones de los términos relevantes para el contexto problemático.

* Agenda: La agenda es el recurso utilizado por las personas para gestionar sus tareas cotidianas con un orden temporal determinado. Las personas utilizan con más frecuencia dicha agenda para anotar sus tareas y compromisos profesionales. Desde este punto de vista, el uso de la agenda es un medio para tener una buena gestión del tiempo y aprovechar de un modo efectivo la jornada [laboral](https://www.definicionabc.com/social/laboral.php).( <https://www.definicionabc.com/negocios/agenda.php>)
* Materia: Son las asignaturas que forman una carrera o un plan de estudios, y que se dictan en los centros educativos. (<https://definicion.de/asignaturas/>)
* Créditos: Un crédito es una unidad de medida del trabajo académico del estudiante. Cada una de las materias que se debe cursar tiene una valoración determinada por el número de créditos, que se especifica en el plan de estudios.  Para obtener un grado académico es necesario reunir un número mínimo de créditos mismos que se aprueban con cada una de las materias. El Sistema de Créditos permite flexibilidad de horarios y de tiempo en el que cursarás tu carrera. (<http://www.udg.mx/es/info/preguntas/en-qu-consiste-el-sistema-de-cr-ditos>)
* Archivo de texto plano: Los archivos de texto plano (en inglés pl ain text) son aquellos que están compuestos únicamente por texto sin formato, sólo caracteres. Estos caracteres se pueden codificar de distintos modos dependiendo de la lengua usada. Algunos de los sistemas de codificación más usados son: ASCII, ISO-8859-1 o Latín1, Unicode, etc…(<https://www.ecured.cu/Archivos_de_textos> ).
* Serialización: La serialización es el proceso de convertir un objeto en una secuencia de bytes para almacenarlo o transmitirlo a la memoria, a una base de datos o a un archivo. Su propósito principal es guardar el estado de un objeto para poder volver a crearlo cuando sea necesario. El proceso inverso se denomina deserialización.(<https://docs.microsoft.com/es-es/dotnet/csharp/programming-guide/concepts/serialization/>)

A continuación, se mostrará el método implementado:



Tomado de <https://es.slideshare.net/chernando/persistencia-en-java-serializacin>

**Búsqueda de soluciones creativas.**

Dadas las funcionalidades del aplicativo, se decide recurrir a lluvia de ideas, se establecieron las posibles soluciones a la visualización de los datos, de manera que sea fácil y comprensible por parte del usuario final. Por lo tanto, se expondrán a continuación las soluciones.

* Idea 1: Implementar un ArrayList, en donde se almacenen los estudiantes y otro donde se almacenen las materias.
* Idea 2: Implementar un HashMap, en donde se almacenen los estudiantes y otro donde se almacenen las materias.
* Idea 3: Implementar un árbol, en donde se almacenen los estudiantes y otro donde se almacenen las materias.
* Idea 4: Implementar listas enlazadas, en donde se almacenen los estudiantes y otra donde se almacenen las materias.
* Idea 5: Implementar serialización para guardar los datos de forma binaria.
* Idea 6: Lectura de datos, para almacenar los datos de los contactos.

**Transición de las ideas a los diseños preliminares.**

En un consenso colectivo las ideas 1, 3 y 4 se descartaron, aquí el por qué: Teniendo a la mano, la implementación de ArrayList debido que si en un futuro se llegaran a utilizar una gran cantidad de datos sería muy ineficiente en cuestión de tiempo, por otro lado con árboles de búsqueda las desventajas que tendrían serían las siguientes, si el árbol no es equilibrado las búsquedas son ineficientes, si los nodos se van insertando aparecerán en orden aleatorio, y para finalizar las listas enlazadas, debido a que ocupan más espacio de memoria debido a sus punteros. A continuación, se explicarán a más a detalle las ideas restantes:

* Idea 2: Implementar un HashMap, en donde se almacenen los estudiantes y otro donde se almacenen las materias. Una de las ventajas de este método es la rapidez en la inserción de los elementos.
* Idea 5: Implementar serialización para guardar los datos de forma binaria. Es un método de persistencia de objetos que es más conveniente que escribir en un archivo de texto plano
* Idea 6: