**Pasos en un Proyecto de Software**

1) Entrevista con el Cliente que necesita le desarrollemos Software**. (En Análisis y**

**Diseño Estructurado y en Análisis y Diseño Orientado a Objetos)**

2) Del resultado de la Entrevista surge un **Texto con los Requerimientos o Requisitos**

**del Cliente. (En Análisis y Diseño Estructurado y en Análisis y Diseño Orientado a**

**Objetos)**

3) Se analiza el **Texto** y se marcan todos los sustantivos que aparecen y que son

importantes y vitales para el negocio del Cliente. **.(En Análisis y Diseño**

**Estructurado y en Análisis y Diseño Orientado a Objetos)**

**En Orientación a Objetos este Texto también dará origen a los Casos de Uso. Los Casos de Uso están conformados por Texto y Diagramas de Casos de Uso.(No se estudian los Casos de Uso en Programación 2. Los estudiarán en la materia Metodología de Sistemas.)**

4) Esos sustantivos del **Texto**, en la jerga de la Ingeniería de Software se las llamará

**ENTIDADES. (En Análisis y Diseño Estructurado)**

**Y se llamarán CLASES.(En Análisis y Diseño Orientado a Objetos, se las llamará**

**CLASES)**

5) Con esas **ENTIDADES** se hace un gráfico o diagrama llamado **MODELO ENTIDAD**

**RELACIÓN sin atributos. (En Análisis y Diseño Estructurado)**

**En Análisis y Diseño Orientado a Objetos con esas Clases se hacen gráficos**

**llamados DIAGRAMAS de CLASES. Cada Clase en este momento ya consta de sus**

**atributos Prototipos de Métodos o Funciones. Esos Diagramas de Clases son Planos**

**de Software, que servirán para construir Software es decir escribir programas.**

6) **En Análisis y Diseño Estructurado**, al **MODELO ENTIDAD**

**RELACIÓN sin atributos** le agregan los atributos de las **ENTIDADES** y resulta el

**MODELO CONCEPTUAL DE DATOS** que se usa para el diseño de las tablas de las

Bases de Datos que pueda requerir el Sistema.(Todo eso lo verán en la materia Base

de Datos).

------------------------------------------------------------------------------------------------------------------

Desde el punto de vista de la Programación nos concentraremos en analizar que hará el Sistema con las **ENTIDADES (CLASES en Orientación a Objetos)** descubiertas, que surgieron del **Texto resultado de la Entrevista o Entrevistas con el Cliente.**

**En Orientación a Objetos los DIAGRAMAS de CLASES son Planos de Software para construir Software, es decir escribir los Programas correspondientes. En Programación 2 se aprenderá a escribir estos Programas en C++. El Programador siguiendo el DIAGRAMA de CLASES construirá Software escribiendo Programas.**

---------------------------------------------------------------------------------------------------------------

Una vez identificada la Clase, usando UML(Unified Modelling Language) se puede hacer el Gráfico de esa clase, y luego el Programador observando el Gráfico de la Clase la puede programar en C++ o Java.

Un Diagrama de Clases pude tener como mínimo una Clase, pero lo normal es que tengan varias clases y relacionadas según las siguientes Relaciones Relaciones:

1) **Asociación,** con sus dos componentes **“Parte de” o “Has to”,** **Agregación Simple** y

**Composición**. Dependencia,

2) **Dependencia.**

3) **Generalización o Herencia.**

En Programación 2 vamos a aprender estas Relaciones entre Clases y a graficarlas en los Diagramas de Clases usando UML(Unified Modelling Language) y a programarlas en C++ usando como “Plano de Software” el Diagrama de Clases realizado previamente.

**Resumiendo:**

**Primero)** Entrevista con el Cliente (una o varias) para obtener los Requisitos o Requerimientos de ese Cliente con lo que espera que el Sistema de Software que le diseñemos le resuelva para su negocio . Se tomará cuidadosa nota de todas sus necesidades y en ese escrito seleccionar los sustantivos que nombran a Entidades o Clases que son esenciales para su negocio.

**Ejemplos:** En Software para Bancos la entidad o clase Cliente.

En Software para Instituciones Educativas la entidad o clase Alumno.

**Segundo**) Generar un Texto o eventualmente **Casos de Uso** de lo que hará el Sistema

de Software aplicado a su negocio para satisfacer los Requerimientos.

**Tercero)** Habiendo realizado los pasos **Primero** y S**egundo** ya se podrá Hacer el

**Diagrama de Clases** donde se visualizará cada **Clase( Nombre de la Clase,**

**atributos y funcionalidades de la Clase( Prototipos de Métodos o Funciones,**

**cuyos cuerpos tenemos que programar))** y las **Relaciones** entre ellas.

Este Diagrama servirá como **Plano de Software** que nos guiará para escribir

**Programas en C++, Java o algún otro lenguaje Orientado a Objetos.**

**Recuerden que en Diseño Orientado a Objetos vimos Diagramas de Estado y Diagramas de Actividades que también ayudan para la Programación Orientada a Objetos.**