

# Resumen

---

## Análisis de la programación visual

**Profesora:**

Torres Servín Emmanuel.  
Programación visual.

**Grupo:**

ingeniería en software  
4322-IS

**Estudiantes:**

1321124285 | Hernández Silva Raúl Abraham.  
1321124381 | Gutiérrez García Harold Guillermo.

# Contenido

<u>CONCEPTOS DE PROGRAMACIÓN ORIENTADA A OBJETOS.</u>	<u>2</u>
CARACTERÍSTICAS.	2
<u>APLICACIÓN DE EVENTOS.</u>	<u>2</u>
CARACTERÍSTICAS DE COMPONENTES.	3
<u>MÉTODOS VISUALES.</u>	<u>4</u>
<u>REQUERIMIENTOS VISUALES DE PROYECTOS DISTRIBUIDOS Y DE ESCRITOS.</u>	<u>5</u>
<u>HERRAMIENTAS Y LENGUAJES.</u>	<u>6</u>
<u>REFERENCIAS.</u>	<u>7</u>

## Conceptos de programación orientada a objetos.

Es un modelo o un estilo de la programación que nos guía a como trabajar con él. Se basa en el concepto de clases y objetos, este tipo de programación es utilizada para hacer una estructura de programación de software en piezas simples y reutilizables que son los códigos (clases) para crear instancias individuales de objetos.



### Características.

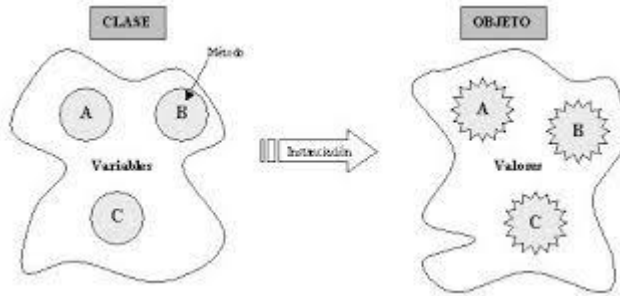
En este se manejan conceptos básicos como lo son clases, objetos, atributos, métodos y se caracteriza por emplear la abstracción de datos, herencia, encapsulamiento y polimorfismo. Como consejo estas características deben de estudiarse para ser comprendidas para poder aplicarlas en la programación orientada a objetos.

### Aplicación de eventos.

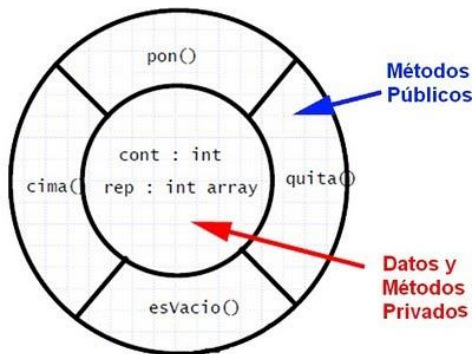
Es la base de lo que llamamos interfaz de usuario, aunque puede emplearse también para desarrollar interfaces entre componentes de Software o módulos del núcleo.

## Características de componentes.

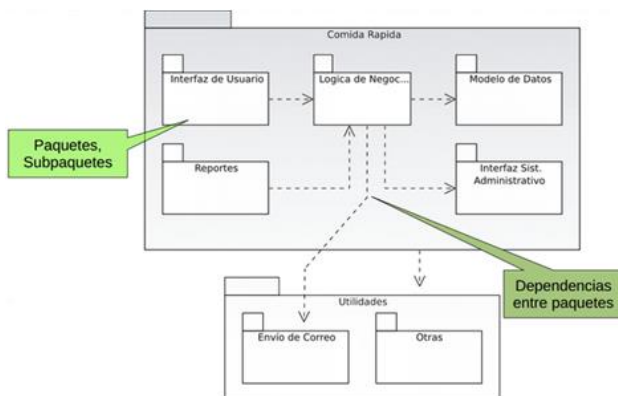
**Abstracción:** las características específicas de un objeto, aquellas que lo distinguen de los demás tipos de objetos.



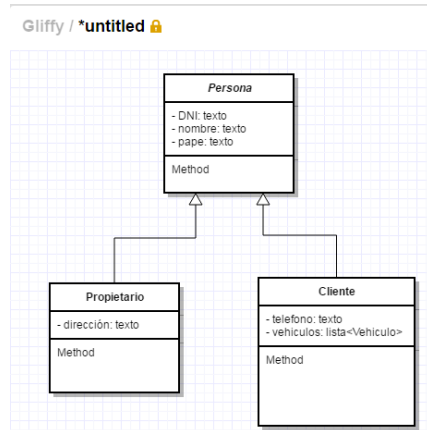
**Encapsulamiento:** Es el ocultamiento del estado, es decir, de los datos miembro, de un objeto de manera que sólo se puede cambiar mediante las operaciones definidas para ese objeto.



**Modularidad:** consiste en dividir un programa en módulos que puedan compilarse por separado, sin embargo, tendrá conexiones con otros módulos. El modularidad también tiene principios y es: la capacidad de descomponer un sistema complejo



**Jerarquía:** Se utiliza en programación orientada a objetos, para clasificar a las abstracciones y a eso se le conoce como Jerarquía.



Así mismo este modelo tiene elementos secundarios:

- Tipos (Tipificación)
- Concurrencia
- Persistencia.

## Métodos visuales.

La programación orientada a objetos (POO) es una forma de programación en donde se desarrollan soluciones utilizando componentes u objetos de software.

A diferencia de la programación estructurada, la POO se parece más al mundo real, porque se trabaja con objetos.

Hay muchos componentes de este tipo, atributos, etc. Hay muchos componentes de este tipo, como pueden ser los botones (TButton), etiquetas de cómo pueden ser los botones (TButton), etiquetas de texto (TLabel), formas (TShape), etc.

- Cronogramas
- Tableros Kanban
- Calendarios

Cada una de estas formas de ver tu trabajo tiene sus ventajas y desventajas, y elegir cuál funciona mejor para ti dependerá del tipo de proyecto que planifiques. Por ese motivo, es muy importante que el software de gestión visual de proyectos te brinde la posibilidad de cambiar entre diferentes tipos de vista. De esa forma, todos los involucrados, de colaboradores individuales a ejecutivos, pueden verse beneficiados por el software para gestión de proyectos.

## **Requerimientos visuales de proyectos distribuidos y de escritos.**

**Necesidad:** Un interesado demanda un requerimiento.

**Característica:** Un servicio proporcionado por el sistema, por lo general formulado por un analista de negocios.

**Caso de uso:** Una descripción del comportamiento del sistema descrito como una secuencia de acciones.

**Requisito complementario:** Otro requisito (generalmente no funcional) que no puede ser contemplado en los casos de uso.

**Caso de prueba:** Una especificación de las entradas necesarias para una prueba, las condiciones de ejecución y resultados esperados. Tiene el papel de comprobar si los casos de uso derivados de los casos de prueba y los requisitos complementarios se aplican correctamente.

**Escenario:** Una secuencia específica de acciones o una ruta de acceso específica a través de un caso de uso. Ayudan a derivar en casos de uso a partir de los casos de prueba y facilitan el diseño e implementación a través de los casos de uso.

## **Herramientas y lenguajes.**

Algunos ejemplos de estas herramientas son: Scratch, adecuada a partir de los 8 años, ScratchJr, que es la versión de Scratch adaptada para niños y niñas de entre 5 y 7 años, Lightbot, etc. Incluso la empresa Google ha creado un precioso doodle llamado Coding for Carrots que ejemplifica este tipo de herramientas

Los lenguajes de programación visual permiten a los usuarios crear programas mediante la manipulación de elementos gráficos, en lugar de especificarlos exclusivamente de manera textual.

Y estos son algunos lenguajes que ocupa la programación visual: HTML, CSS, JavaScript y JSON. Saque el máximo provecho de todo lo que le ofrecen LESS y Sass, y use PHP, Python o C# con ASP.NET

## Referencias.

*4.2 componentes del entorno visual.* (s. f.). CIDECAEME

UAEH. [http://cidecame.uaeh.edu.mx/lcc/mapa/PROYECTO/libro15/42\\_componentes\\_del\\_entorno\\_visual.html?fbclid=IwAR1Zhs5QzK8VQ-n2WMc51xksYbE5DPcLB5l8zr5XFP7pnmXJyO0Bed7fk4E](http://cidecame.uaeh.edu.mx/lcc/mapa/PROYECTO/libro15/42_componentes_del_entorno_visual.html?fbclid=IwAR1Zhs5QzK8VQ-n2WMc51xksYbE5DPcLB5l8zr5XFP7pnmXJyO0Bed7fk4E)

*Características de la POO.* (s. f.). Portal Académico del

CCH. <https://portalacademico.cch.unam.mx/cibernetica1/algoritmos-y-codificacion/caracteristicas->

[POO#:~:text=El%20paradigma%20de%20la%20programaci%C3%B3n,%20herencia,%20encapsulamiento%20y%20polimorfismo.](https://portalacademico.cch.unam.mx/cibernetica1/algoritmos-y-codificacion/caracteristicas-POO#:~:text=El%20paradigma%20de%20la%20programaci%C3%B3n,%20herencia,%20encapsulamiento%20y%20polimorfismo.)

*Lenguajes - visual studio.* (s. f.). Visual

Studio. [https://visualstudio.microsoft.com/es/vs/features/web/languages/?fbclid=IwAR3Qj7d\\_WrIbkVsXGKIP4U4iZNCooH9iwdhobLy4GNb1Z9FqoXO5gaeGmQ](https://visualstudio.microsoft.com/es/vs/features/web/languages/?fbclid=IwAR3Qj7d_WrIbkVsXGKIP4U4iZNCooH9iwdhobLy4GNb1Z9FqoXO5gaeGmQ)

*Programación orientada a eventos: Características, ejemplos, ventajas, aplicaciones.* (s. f.).

Lifeder. <https://www.lifeder.com/programacion-orientada-a-eventos/?fbclid=IwAR0Fh6nbnNaFkvDN7umlQuKli2gkrDMnAG3Qb0zAZP1pUt1ITkd26Qyhock>