**Sistemas Gestores de Bases de Datos**

Hoy día la mayoría de las bases de datos se presentan en formato digital, gracias a los avances tecnológicos en la informática y la electrónica. Esto ofrece un amplio abanico de soluciones al problema de almacenamiento de datos. Los gestores de bases de datos, Database Management System o DBMS (SGBD) son programas que permiten almacenar y luego acceder a los datos de forma estructurada y rápida. Las aplicaciones más usadas son para gestiones de empresas e instituciones públicas, así como en entornos científicos, para almacenar la información experimental. Una base de datos es un sistema compuesto por un conjunto de datos, los cuales están almacenados en discos, a los que se accede directamente y un conjunto de programas que regulen o manejen ese conjunto de datos. Mientras que un sistema de Gestión de Bases de Datos es un software que sirve de interfaz entre la base de datos, el usuario y las aplicaciones que se utilizan.

![image (12)](https://user-images.githubusercontent.com/106564121/177388190-2dbc415b-7876-433f-95b8-2ef547555589.png)

**Los mejores gestores de base de datos**

**Oracle**

Es de los más confiables sistemas de gestión de base de datos relacional, además del más usado. Es propiedad de Oracle Corporation y fue desarrollado en 1977.

Se accede directamente a los objetos, a través del lenguaje de consulta SQL, es muy utilizado en las empresas, con un componente de red que permite la comunicación a través de las redes.

Su versatilidad le facilita ejecutarse en casi todas las plataformas existentes, Windows, Unix, Linux, MAC OS, entre otros.

**SQL Server**

SQL Server se ejecuta en Transact-SQL, esto es un grupo de programas que pueden añadir características al programa, como tratamiento de errores y excepciones, extracción de datos de la web en forma directa, procesamiento de datos, uso de distintos lenguajes de programación y otros más, que lo hacen un gestor muy completo y competitivo. Su carácter administrativo es otro valor agregado, tanto en sus funciones y seguridad, como en su flexibilidad.

**Gestores de base de datos de acceso libre**

**MySQL**

Este es de simple instalación y actúa de lado del cliente o servidor, es de código abierto y tiene licencia comercial disponible. Pertenece a Oracle Corporation y gestiona las bases de datos relacionales, con funciones multiusuario y es el más usado dentro del software libre. Requiere de poca memoria y procesador para su funcionamiento, lo que se traduce en mayor velocidad en sus operaciones. Se usa principalmente para el desarrollo Web.

**Fire Bird**

De gran potencia y muy sencillo a la vez, este sistema de gestión de base de datos relacional SQL, es uno de los mejores gestores Open Source (Código abierto) o libres. Es compatible con Windows y Linux. Tiene buen soporte para los procedimientos almacenados, las transacciones compatibles con ACID y con los métodos de acceso múltiple como Nativo, Python, .NET, etc...

**Tipos de Sistemas Gestores de Bases de Datos**

**Diferencias entre el modelo lógico y el conceptual**

El modelo conceptual es independiente del DBMS que se vaya a utilizar. El lógico depende de un tipo de

SGBD en particular.

El modelo lógico está más cerca del modelo físico, el que utiliza internamente el ordenador.

El modelo conceptual es el más cercano al usuario, el lógico es el encargado de establecer el paso entre

el modelo conceptual y el modelo físico del sistema.

**Tipos de Sistemas Gestores de Bases de Datos**

Algunos ejemplos de modelos conceptuales son:

-Modelo Entidad Relación

-Modelo RM/T

-Modelo UML

Ejemplos de modelos lógicos son:

-Modelo relacional

-Modelo Codasyl

-Modelo Jerárquico

A continuación se comentarán los modelos lógicos más importantes.

**Modelo jerárquico.**

Era utilizado por los primeros SGBD, desde que IBM lo definió para su IMS (Information Management System, Sistema Administrador de Información) en 1970. Se le llama también modelo en árbol debido a que utiliza una estructura en árbol para organizar los datos.

La información se organiza con un jerarquía en la que la relación entre las entidades de este modelo siempre es del tipo padre / hijo. De esta forma hay una serie de nodos que contendrán atributos y que se relacionarán con nodos hijos de forma que puede haber más de un hijo para el mismo padre (pero un hijo sólo tiene un padre).

Los datos de este modelo se almacenan en estructuras lógicas llamadas segmentos. Los segmentos se relacionan entre sí utilizando arcos.

La forma visual de este modelo es de árbol invertido, en la parte superior están los padres y en la inferior los hijos.

Este esquema está en absoluto desuso ya que no es válido para modelar la mayoría de problemas de

bases de datos.

**Modelo en red (Codasyl).**

Es un modelo que ha tenido una gran aceptación (aunque apenas se utiliza actualmente). En especial se

hizo popular la forma definida por Codasyl a principios de los 70 que se ha convertido en el modelo en

red más utilizado.

El modelo en red organiza la información en registros (también llamados nodos) y enlaces. En los

registros se almacenan los datos, mientras que los enlaces permiten relacionar estos datos. Las bases de

datos en red son parecidas a las jerárquicas sólo que en ellas puede haber más de un padre.

En este modelo se pueden representar perfectamente cualquier tipo de relación entre los datos (aunque

el Codasyl restringía un poco las relaciones posibles), pero hace muy complicado su manejo.

**Modelo relacional.**

En este modelo los datos se organizan en tablas cuyos datos se relacionan. Es el modelo más popular y se

describe con más detalle en los temas siguientes que veremos.

**Modelo de bases de datos orientadas a objetos.**

Desde la aparición de la programación orientada a objetos (POO u OOP) se empezó a pensar en bases

de datos adaptadas a estos lenguajes. La programación orientada a objetos permite cohesionar datos y

procedimientos, haciendo que se diseñen estructuras que poseen datos (atributos) en las que se

definen los procedimientos (operaciones) que pueden realizar con los datos. En las bases orientadas a

objetos se utiliza esta misma idea.

A través de este concepto se intenta que estas bases de datos consigan arreglar las limitaciones de las

relacionales. Por ejemplo el problema de la herencia (el hecho de que no se puedan realizar relaciones

de herencia entre las tablas), tipos definidos por el usuario, disparadores (triggers) almacenables en la

base de datos, soporte multimedia...

Se supone que son las bases de datos de tercera generación (la primera fue las bases de datos en red y

la segunda las relacionales), lo que significa que el futuro parece estar a favor de estas bases de datos.

Pero siguen sin reemplazar a las relacionales, aunque son el tipo de base de datos que más está

creciendo en los últimos años.

Su modelo conceptual se suele diseñar en UML y el lógico actualmente en ODMG (Object Data

Management Group, grupo de administración de objetos de datos, organismo que intenta crear

estándares para este modelo).

**Bases de datos objeto-relacionales.**

Tratan de ser un híbrido entre el modelo relacional y el orientado a objetos. El problema de las bases de

datos orientadas a objetos es que requieren reinvertir capital y esfuerzos de nuevo para convertir las

bases de datos relacionales en bases de datos orientadas a objetos. En las bases de datos objeto

relacionales se intenta conseguir una compatibilidad relacional dando la posibilidad de integrar mejoras

de la orientación a objetos.

Estas bases de datos se basan en el estándar SQL 99. En ese estándar se añade a las bases

relacionales la posibilidad de almacenar procedimientos de usuario, triggers, tipos definidos por el

usuario, consultas recursivas, bases de datos OLAP, tipos LOB,...

Las últimas versiones de la mayoría de las clásicas grandes bases de datos relacionales (Oracle, SQL

Server, Informix, ...) son objeto relacionales.

**Bases de datos NoSQL**.

Bajo este nombre se agrupan las bases de datos (con arquitecturas muy diversas) pensadas para grabar

los datos de manera veloz para así poder atender a miles y miles de peticiones. Es decir, es el modelo

de las bases de datos que se utilizan en los grandes servicios de Internet (como twitter, Facebook,

Amazon,…).

La idea es que los datos apenas necesitan validarse y relacionarse y lo importante es la disponibilidad de

la propia base de datos. El nombre NoSQL, hace referencia a que este modelo de bases de datos rompe

con el lenguaje SQL (el lenguaje de las bases de datos relacionales, las bases de datos dominantes

hasta la actualidad) para poder manipular los datos con lenguajes de otro tipo.

**Diseño de bases de datos relacionales**

En este tema se estudia un aspecto fundamental de las bases de datos: su diseño. En las bases de datos se ha establecido un ciclo de desarrollo que consta de tres etapas de diseño: el diseño conceptual, el diseño lógico y el diseño físico. Mientras que las dos primeras etapas y el paso de una a otra están muy fundamentados, no ocurre lo mismo con la tercera, dado que las primeras son lo suficientemente abstractas como para no depender de ninguna implementación en concreto; sin embargo, el diseño físico depende del SGBD usado, y no hay reglas formales para llevarlo a cabo.

**Etapas de diseño**

La metodología de diseño de bases de datos relacionales se ha consolidado a lo largo de los años satisfaciendo las propiedades de generalidad (independencia de la plataforma hardware/software), calidad del producto (precisión, completitud y eficacia) y facilidad de uso.

**Consta de las siguientes etapas:**

1. Diseño conceptual.

Su objetivo es definir las entidades y las relaciones entre ellos de forma abstracta, sin centrarse en ningún modelo lógico en concreto (como el relacional, el orientado a objetos, el jerárquico o el de red).

Herramienta: Modelo conceptual de datos. Se usa alguna variante del modelo entidad-relación para las bases de datos relacionales.

Resultado: Esquema conceptual de la base de datos.

2. Diseño lógico.

Su objetivo es definir el esquema de la base de datos según el

modelo que implementa el SGBD objetivo.

Herramienta: Modelo lógico de datos. Se usa el modelo lógico que implemente el sistema de gestión de bases de datos objetivo, pero es independiente de los aspectos físicos. Se usan técnicas formales para verificar la calidad del esquema lógico; la más usual es la

normalización. En el modelo relacional se usan las tablas.

Resultado: Esquema lógico de la base de datos.

3. Diseño físico.

Su objetivo es definir el esquema físico de la base de datos de forma que se den todas las instrucciones para que un DBA pueda implementar la base de datos sin ninguna ambigüedad. Se considera el rendimiento como un aspecto que no se ha tratado en las etapas anteriores.

Herramienta: Modelo físico de datos. Se consideran todos los detalles de la implementación física: organización de archivos e índices para el SGBD considerado.

Resultado: Esquema físico de la base de datos.

La siguiente figura muestra resumido el ciclo de desarrollo clásico de bases de datos:

![image](https://user-images.githubusercontent.com/106564121/177389299-834f9409-e67f-4375-94b2-54d517810dd9.png)

**Guía de estudio Metodologías Agiles y Gestión de Proyectos**

**Metodología Scrum**

**Software**

El software son los programas, los documentos asociados y la configuración de datos que

se necesitan para hacer que estos programas operen de manera correcta

La documentación se refiere a documentos que describen:

● la organización del sistema,

● documentos para el usuario que les explica cómo utilizar el sistema

● sitios web que permitan a los usuarios descargar la información de

actualizaciones del producto

**Software (IEEE):** parte de un sistema que se puede codificar para ejecutarse en una computadora como un conjunto de instrucciones. Incluye la documentación asociada necesaria para comprender, transformar y usar esa solución.

**Desarrollo de Software**

**Proceso para el desarrollo de software**

Define un conjunto completo de actividades necesarias para transformar los requerimientos de un usuario en un producto.

**Ciclo de vida**

Un ciclo de vida es el conjunto de fases [o procesos] por las que pasa el sistema de software desde que se concibe [o inicio], se desarrolla hasta que se retira del servicio finalizando su uso. Las fases o procesos están estandarizados, es decir que existe un marco de referencia que contiene los procesos, las actividades y las tareas involucradas en el desarrollo, explotación y mantenimiento de un producto de software, abarcando la vida del sistema, desde la definición de requerimientos hasta la finalización de su uso.

**Scrum**

**Definición de Scrum**

Scrum (n): Es un marco de trabajo a través del cual las personas pueden abordar problemas complejos adaptativos, a la vez que se entregan productos de forma eficiente y creativa con el máximo valor.

● Es una enfoque ágil para la gestión de un proyecto. Más que una metodología o proceso, es un Marco de Trabajo.

● Utiliza procesos interactivos/incrementales.

● Orientado a resultados y compromisos.

● No está restringido a proyectos de software solamente.

● Su visión es opuesta a la propuesta por la metodología en cascada.

**Pilares**

Transparencia: Los aspectos significativos del proceso deben ser visibles para todos aquellos que son responsables del resultado. La transparencia requiere que dichos aspectos sean definidos en base a un estándar común, de tal modo que los observadores compartan un entendimiento común de lo que se están viendo. Por ejemplo:

• Deben compartir un lenguaje común todos los participantes para referirse al proceso.

• Aquellos que desempeñan el trabajo y quienes inspeccionan el incremento resultante deben compartir una definición común de “Terminado” (“Done”).

Inspección: Los usuarios de Scrum deben inspeccionar frecuentemente los Artefactos de Scrum y el progreso hacia un objetivo para detectar variaciones indeseadas. Su inspección no debe ser tan frecuente como para que pueda interferir en el trabajo. Las inspecciones son más beneficiosas cuando se realizan de forma diligente por inspectores expertos en el mismo lugar de trabajo.

Adaptación: Si un inspector determina que uno o más aspectos de un proceso se desvían de los límites aceptables y que el producto resultante será inaceptable, el proceso o el material que está siendo procesado deben ajustarse. Dicho ajuste deberá realizarse cuanto antes para minimizar desviaciones mayores.

**Sprint**

El Sprint es un período de corta duración que debe finalizar con un prototipo operativo o producto parcialmente entregable. El mismo se repite n veces a lo largo del proyecto y permite hacer entregas de producto en partes, donde cada entrega, es un incremento de funcionalidad respecto al anterior. Esto difiere del conocido ciclo de vida en cascada muy utilizado en el campo de desarrollo de software en que, las fases del ciclo de vida (requisitos, análisis, diseño, codificación, testing, etc.) se realizan una única vez y, el inicio de cada fase no comienza hasta que termina la que precede.

**Scrum Team**

Ceremonias que se realizan a lo largo del proyecto:

● Sprint Planning Meeting (reunión de planificación del sprint). Se produce al iniciar cada Sprint y tiene por objetivo decidir que se va a realizar en el Sprint.

● Daily Scrum Meeting (reunión periódica). Se produce diariamente, y tiene un máximo de 20 min. de duración. Tiene por objeto tratar qué es lo que se hizo, qué se va a hacer y qué problemas se han encontrado, esto a fines de encontrar soluciones en la diaria.

● Sprint Review Meeting (reunión de revisión del Sprint). Se produce al finalizar el Sprint y tiene por objeto mostrar qué es lo que se ha completado y qué no. Debe estar presente el Product Owner.

● Sprint Retrospective Meeting (reunión de retrospectiva del Sprint). Se produce también al finalizar el Sprint y tiene por objeto documentar qué ha funcionado y qué no ha funcionado en el Sprint. La idea de dicha reunión es centrar al equipo en lo que salió bien y en lo que debe mejorar para la próxima iteración. De ninguna manera se centra en lo que salió mal.

**Github**

Mediante la plataforma de *social coding* [GitHub](https://github.com/) puedes publicar repositorios de código para que se alojen en un sistema de control de versiones Git.

Este sistema te ofrece la posibilidad de colaborar en otros proyectos y publicar los tuyos propios.

La plataforma es de código abierto por defecto, por lo que cualquier persona puede utilizar tu código y tú también puedes ver el código de otros proyectos

**Git**

**Control de versiones**

¿Qué es un control de versiones, y por qué debería importarte? Un control de versiones es un sistema que registra los cambios realizados en un archivo o conjunto de archivos a lo largo del tiempo, de modo que puedas recuperar versiones específicas más adelante.

Dicho sistema te permite regresar a versiones anteriores de tus archivos, regresar a una versión anterior del proyecto completo, comparar cambios a lo largo del tiempo, ver quién modificó por última vez algo que pueda estar causando problemas, ver quién introdujo un problema y cuándo, y mucho más. Adicionalmente, obtendrás todos estos beneficios a un costo muy bajo.

**Sistemas de Control de Versiones Locales**  
  
Es fácil olvidar en qué directorio te encuentras y guardar accidentalmente en el archivo equivocado o sobrescribir archivos que no querías.  
  
**Sistemas de Control de Versiones Centralizados**  
  
Estos sistemas, como CVS, Subversion y Perforce, tienen un único servidor que contiene todos los archivos versionados y varios clientes que descargan los archivos desde ese lugar central. Por ejemplo, todas las personas saben hasta cierto punto en que están trabajando los otros colaboradores del proyecto. Si ese servidor se cae durante una hora, entonces durante esa hora nadie podrá colaborar o guardar cambios en archivos en los que hayan estado trabajando. Si el disco duro en el que se encuentra la base de datos central se corrompe, y no se han realizado copias de seguridad adecuadamente, se perderá toda la información del proyecto, con excepción de las copias instantáneas que las personas tengan en sus máquinas locales.  
  
**Sistemas de Control de Versiones Distribuidos**  
  
En un DVCS , los clientes no solo descargan la última copia instantánea de los archivos, sino que se replica completamente el repositorio. Cada clon es realmente una copia completa de todos los datos.  
  
**Una breve historia de Git**  
  
Durante la mayor parte del mantenimiento del kernel de Linux , los cambios en el software se realizaban a través de parches y archivos.  
  
Para ser eficiente, si los archivos no se han modificado Git no almacena el archivo de nuevo, sino un enlace al archivo anterior idéntico que ya tiene almacenado.  
  
**Casi todas las operaciones son locales**  
  
La mayoría de las operaciones en Git sólo necesitan archivos y recursos locales para funcionar. Por lo general no se necesita información de ningún otro computador de tu red. Debido a que tienes toda la historia del proyecto ahí mismo, en tu disco local, la mayoría de las operaciones parecen prácticamente inmediatas. Por ejemplo, para navegar por la historia del proyecto, Git no necesita conectarse al servidor para obtener la historia y mostrarla -simplemente la lee directamente de tu base de datos local.  
  
Esto significa que ves la historia del proyecto casi instantáneamente. Si quieres ver los cambios introducidos en un archivo entre la versión actual y la de hace un mes, Git puede buscar el archivo de hace un mes y hacer un cálculo de diferencias localmente, en lugar de tener que pedirle a un servidor remoto que lo haga, u obtener una versión antigua desde la red y hacerlo de manera local.  
  
**Git tiene integridad**  
  
Esto significa que es imposible cambiar los contenidos de cualquier archivo o directorio sin que Git lo sepa. No puedes perder información durante su transmisión o sufrir corrupción de archivos sin que Git sea capaz de detectarlo. Se trata de una cadena de 40 caracteres hexadecimales , y se calcula con base en los contenidos del archivo o estructura del directorio en Git.  
  
De hecho, Git guarda todo no por nombre de archivo, sino por el valor hash de sus contenidos.  
  
**Git generalmente solo añade información**  
  
Cuando realizas acciones en Git, casi todas ellas sólo añaden información a la base de datos de Git. Pero después de confirmar una copia instantánea en Git es muy difícil perderla, especialmente si envías tu base de datos a otro repositorio con regularidad.  
  
**Los Tres Estados**  
  
El directorio de Git es donde se almacenan los metadatos y la base de datos de objetos para tu proyecto. Es la parte más importante de Git, y es lo que se copia cuando se clona un repositorio desde otra computadora. El directorio de trabajo es una copia de una versión del proyecto. Estos archivos se sacan de la base de datos comprimida en el directorio de Git, y se colocan en disco para que los puedas usar o modificar.  
  
El área de preparación es un archivo, generalmente contenido en tu directorio de Git, que almacena información acerca de lo que va a ir en tu próxima confirmación. A veces se le denomina índice , pero se está convirtiendo en estándar el referirse a ella como el área de preparación.

ÉTICA Y DEONTOLOGÍA PROFESIONAL:

**LA ÉTICA EN LA INFORMÁTICA :**

La ética es una disciplina filosófica que se define como "principios directivos que orientan a las personas en cuanto a la concepción de la vida, el hombre, los juicios, los hechos, y la moral”.

La ética en la informática es una nueva disciplina que pretende abrirse campo dentro de las éticas aplicadas por lo cual encontramos varias definiciones.

Se define "como la disciplina que analiza los problemas éticos que son creados por la tecnología de los ordenadores o también los que son transformados o agravados por la misma". Es decir, por las personas que utilizan los avances de las tecnologías de la información.

"Es el análisis de la naturaleza y el impacto social de la tecnología informática y la correspondiente formulación y justificación de políticas para un uso ético de dicha tecnología", esta definición esta relacionada con los problemas conceptuales y los vacíos en las regulaciones que ha ocasionado la tecnología de la información.

También se define a la EI "como la disciplina que identifica y analiza los impactos de las tecnologías de la información en los valores humanos y sociales" . Estos valores

afectados son: la salud, la riqueza, el trabajo, la libertad, la democracia, el conocimiento, la privacidad, la seguridad o la autorrealización personal.

**Objetivos de la EI (Ética Informática) La EI (Ética Informática)**

se plantea varios objetivos:

Descubrir y articular dilemas éticos claves en informática.

Determinar en qué medida son agravados, transformados o creados por la tecnología informática.

Analizar y proponer un marco conceptual adecuado y formular principios de actuación para determinar qué hacer en las nuevas actividades ocasionadas por la informática en las que no se perciben con claridad líneas de actuación.

Utilizar la teoría ética para clarificar los dilemas éticos y detectar errores en el razonamiento ético.

Proponer un marco conceptual adecuado para entender los dilemas éticos que origina la informática y además establecer una guía cuando no existe reglamentación de dar uso a Internet.

**Contenidos :**

1-La Utilización de la Información

2-La propiedad de los bienes informáticos.

3-Lo informático como instrumento de actos potencialmente dañinos

4-la Inteligencia artificial

5-Miedos y amenazas de la informática y Dimensiones sociales de la informática

**LA DEONTOLOGÍA – LA DEONTOLOGÍA INFORMÁTICA.**

la deontología es la parte de la ética que trata sobre los deberes, especialmente relacionados con el ejercicio de una determinada profesión. La Deontología informática hace referencia a la ciencia o tratado de los deberes y normas éticas, que conciernen a los profesionales que a diario están en contacto con la tecnología y los medios masivos de comunicación como internet. Por tanto trata, de la moral o ética profesional en el manejo del activo más importante que tienen las empresas, un bien cada vez más apreciado, que es la información.

**LOS CÓDIGOS ÉTICOS:**

"Los códigos de ética son sistemas de reglas establecidos con el propósito general de guiar el comportamiento de los integrantes de la organización y de aquellos con los cuales ésta actúa habitualmente: clientes, proveedores y contratistas.

Los diez mandamientos de la ética informática.

1. No usarás una computadora para dañar a otros.
2. No interferirás con el trabajo ajeno.
3. No indagarás en los archivos ajenos.
4. No utilizarás una computadora para robar.
5. No utilizarás la informática para realizar fraudes.
6. No copiarás o utilizarás software que no hayas comprado.
7. No utilizarás los recursos informáticos ajenos sin la debida autorización.
8. No te apropiarás de los derechos intelectuales de otros.
9. Deberás evaluar las consecuencias sociales de cualquier código que desarrolles.
10. Siempre utilizarás las computadoras de manera de respetar los derechos de los demás.

**LA ÉTICA EN INTERNET (Ciberespacio).**

La cuestión ética consiste en saber si esto está contribuyendo al auténtico desarrollo humano y ayudando a las personas y a los pueblos a ser fieles a su destino trascendente. "el principio ético fundamental es el siguiente: la persona humana y la comunidad humana son el fin y la medida del uso de los medios de comunicación social; la comunicación debería realizarse de persona a persona, con vistas al desarrollo integral de las mismas" Internet tiene un conjunto de características impresionantes como lo describimos a continuación:

Instantáneo.

Inmediato.

Mundial.

Descentralizado.

Interactivo.

Los problemas éticos más significativos en Internet:

Una enumeración somera de los problemas éticos del ciberespacio nos ofrece la siguiente lista.

La privacidad: no tanto la posibilidad (real) de fallos en los sistemas de correo que provoquen mensajes que van a lugares equivocados, si no más bien la intromisión intencional. Esta intromisión puede ser desde la del compañero del trabajo que lee nuestro trabajo hasta el sistemático intervencionismo estatal.

La Identidad: es posible esconder la verdadera identidad a la hora de intervenir en una conversación.

La Inversa: la capacidad de manipulación se traslada de los mecanismos habituales a la sociedad (como la posición social económica) a las habilidades de aquellos que manejan mas el medio o que pueden intervenir de manera subrepticia en las comunicaciones ajenas.

**LA SOCIEDAD DE LA INFORMACIÓN.**

La Sociedad de la Información es un hecho permanentemente reconstruido por actores que pertenecen a sectores sociales de la gran mayoría de los países del globo. Actores que interactúan supra y transnacionalmente por medio del conjunto integrado, a través de Internet, de infraestructuras nacionales de información para ofrecer, vender, consumir y utilizar información y conocimiento. La Sociedad de la Información es resultado, de los procesos que podríamos llamar de informatización que se producen en los Estados Nacionales, por medio de los cuales se hace posible que actores de cada vez más amplios sectores sociales tengan acceso al uso de las TICs (tecnologías de información y comunicaciones) para su participación en la vida social. Vistas las cosas así, integrando la visión local con la global, una sociedad, un país, un Estado-Nación particular puede no ser una Sociedad de la Información pero, a través de algunos de sus sectores sociales, participar de la Sociedad de la Información. Pero en los países de América Latina y el Caribe los que forman parte de la Sociedad de la Información son minoritarios por lo que afirmamos que las sociedades latinoamericanas no son sociedades de la información. En ellas están presentes en grados relativamente bajos, en todo caso, son sociedades en proceso de informatización; en camino de llegar a ser sociedades de la información, si las vemos desde la óptica evolucionista que impregna mayoritariamente a la opinión especializada en el área.

**Ética y moral**

La ética es una rama de la filosofía y la moral forma parte de la vida diaria.

La ética seria la moral pensada, mientras que la moral sería la moral vivida.

Ética etimológica Proviene de la lengua griega Ethos manera de hacer costumbre. Ico relativo a. Filosofía de las costumbres. -Aristóteles –

**Campo de uso**

En el campo personal la ética suele apuntar a valores fundamentales, como diferenciar el bien del mal, que es justo y deseable y cuál debe ser nuestra postura en cuanto a temas, como la libertad, las desigualdades, los problema sociales, problemas ambientales, la vigencia o no de nuestros códigos morales entre otros temas. Podría decirse que la ética tiende a ser una postura personal e intima adoptada por cada individuo según sus principios

**¿Que la gobierna?**

En el campo laboral existe la llamada ética laboral. Honestidad, puntualidad sobriedad respeto etc. Y en muchas profesiones como la medicina la docencia o el derecho existen los códigos éticos que pueden confundirse con los morales que se corresponden con el código de conducta del profesional.

En algunas profesiones estos códigos contemplan reglas de cumplimiento obligatorio, para regular alguna determinada actividad profesional por ejemplo sería que los profesores/as no se vinculen románticamente con sus alumnos.

**Consistencia**

La ética tiende a ser teórica y personal. Continuamente se pregunta por la naturaleza y el bien en las actividades humanas y sobre la validez de la moralidad en cada sociedad confrontada con valores que hemos dado por buenos y justos, como el derecho a la vida a la igualdad y a la libertad. Una postura ética puede entrar en conflicto. Una postura ética puede entrar en conflicto con códigos y valore morales de determinadas sociedades.

En este sentido podría decirse que la ética es constante, no suele cambiar con el tiempo regula el comportamiento humano desde cada persona y desde su propia conciencia.

**Ejemplos de ética**

Declaración de derechos humanos. Proclamada por las naciones unidas 1948 y traducida a más de 500 idiomas, que en su artículo primero afirma: “todos los seres humanos nacemos libre e iguales en dignidad y derechos, y dotados como están de razón y conciencia deben comportarse fraternalmente los unos para con los otros “.

**Juramento Hipocrático** Un ejemplo también clásico de código ético laboral es el juramento Hipocrático que realizan los profesionales de la medicina al concluir sus estudios en casi todo el mundo, y que aunque actualizado, ha estado vigente por más de dos mil años.

**Ejemplos de Influencias** Ejemplo diario de conducta ética influenciada por el Judeocristianismo es que consideremos natural amar al prójimo y considerar malo el asesinato y el robo. Y de influencia grecolatina la defensa de la igualdad de derechos y la libertad en todas su formas

**Ejemplo de conflicto entre moral y ética** Están aquellos que se dan en torno en temas como el aborto, la pena de muerte, la sexualidad, la libertad de expresión etc.

**Qué es la moral**

Se refiere a los códigos de normas impuestos por una sociedad para regular la conducta o el comportamiento de los individuos. Abarca reglas tradiciones y valores compartidas por una sociedad en determinado momento de su historia. Se dice que una persona es moral cuando actúa de acuerdo a las normas costumbres y valores de su sociedad. Aunque estos pueden estar en contra de la ética de esa misma persona.

**Etimología de moral** Proviene del latín moralis que a su vez deriva del termino mos, moris (costumbre, principios y reglas que rigen en la vida social) .

**Campo de uso**

Los códigos morales se aplican a la vida cotidiana y a la conducta humana, nos dice como es el modo correcto de actuar en cada situación y lugar. En este sentido es la sociedad inmersa en una cultura determinada la que establece el código moral, por el que debe regirse todos sus integrantes. Faltar al código moral suele acarrear sanciones sociales, y penales en algunos casos.

**¿Que la gobierna?**

La moral tiende a ser normativa a reflejarse en reglamentos y prohibiciones, como algunos preceptos recogidos en textos judeocristianos como los diez mandamientos. Forman parte de la moralidad las normas de comportamiento. Valores como la honradez, la discreción en el actuar y vestir, el respeto a la autoridad civil y el acatamiento de normas religiosas. La moralidad puede abarcar incluso aspectos de la dieta diaria y la higiene personal, como la prohibición total o parcial de consumir ciertos alimentos. En las culturas musulmanas y hebreas por ejemplo.

**Consistencia**

Los códigos morales pueden variar en el tiempo y en el espacio, en una misma cultura los valores se van modificando de acuerdo con las dinámicas y los distintos procesos sociales, religiosos, políticos económicos etc. Diferentes valores morales pueden convivir en una misma sociedad bajo una misma ética.

**Ejemplos de comportamiento moral** Cortesía verbal. No mentir. A menos que se trate de una mentira socialmente aceptada. Decir a un enfermo terminal que va a mejorarse. A una persona desanimada y desalineada que tiene buen aspecto que está bien aunque la persona se sienta mal.

**Coherencia en la vida pública y privada** Ser honrados tanto en el trabajo como en la vida diaria. Ejemplo a recibir cambio de más, devolver el sobrante, o si se encuentra una cartera devolverla al propietario. Si es funcionario promover la eficiencia y combatir la corrupción.

**Respetar normas** Acatar las normas de vestir y conducirse al encontrarse en un templo religioso. O al estar en un país con religión dominante distintas a las nuestras.

**EL DILEMA DE LAS REDES SOCIALES**

“Hay dos industrias que llaman a sus clientes usuarios: la de las drogas ilegales y la del software". La frase aparece en uno de los momentos más comentados del documental The Social Dilemma ("El dilema de las redes sociales"), que se estrenó en febrero de 2020 en el Festival de Cine de Sundance, en Estados Unidos, y que, siete meses después, se acaba de convertir en tema de conversación en el resto del mundo tras ser incluido en el catálogo de Netflix. Hilado con testimonios de algunos exejecutivos de las empresas más grandes de Silicon Valley y de académicos, el filme describe la adicción y los impactos negativos de las redes sociales en personas y comunidades, como resultado de las estrategias diseñadas para manipular emociones y comportamientos, y mantener conectados a los usuarios. Así, según "El dilema de las redes sociales", experiencias digitales aparentemente triviales, como recomendaciones automáticas, notificaciones y publicaciones sugeridas, funcionarían como un cebo que es lanzado miles de millones de veces al día por las aplicaciones más populares del planeta. El objetivo sería acaparar el tiempo de las personas, una moneda valiosa para empresas, políticos, organizaciones o países que quieran vender productos o ideas a audiencias vulnerables e hipersegmentadas. Clasificado como docudrama -una fórmula que mezcla testimonios del documental tradicional con escenas dramatizadas-, la película ha sido descrita como una ventana a las mesas de decisión de gigantes de Silicon Valley como Facebook, Twitter y Google. Elogiado por detallar herramientas y estrategias sofisticadas en un lenguaje accesible a grandes audiencias, recibe críticas por no ofrecer soluciones a los problemas enumerados y por ser presentada por personas con características similares: buena parte de los entrevistados son exejecutivos: hombres jóvenes, ricos y blancos. "Si no pagas por el producto, el producto eres tú" En agosto, según el Índice de Multimillonarios del medio especializado en noticias financieras Bloomberg, la fortuna de Mark Zuckerberg superó los US$100.000 millones. Solo durante la pandemia de coronavirus, el fundador de Facebook habría ganado más de US$30.000 millones. ¿Cómo habría podido Zuckerberg ofrecer servicios gratuitos y hacerse más rico cada día? Según los entrevistados de "El dilema de las redes sociales", el estadounidense y sus colegas jefes ejecutivos ganan dinero con el tiempo. Explican que cuantas más horas pasa un usuario conectado a sus redes sociales, más información detallada sobre hábitos, gustos y características de consumo acaba exponiendo. Qué es el "oscuro" capitalismo de la vigilancia de Facebook y Google y por qué lo comparan con la conquista española Cómo descubrir todo lo que Facebook sabe sobre ti en tres sencillos pasos Estos datos son recopilados y organizados mediante algoritmos que mapean los me gusta y los comentarios, analizan los tiempos de lectura y la exposición a las imágenes, y alimentan enormes servidores (algunos de ellos alojados en submarinos).