

Descubriendo Definiciones de Ingeniería

Definición del DRAE (Diccionario de la Real Academia Española de la Lengua):

Conjunto de conocimientos y técnicas que permiten aplicar el saber científico a la utilización de la materia y de las fuentes de energía.

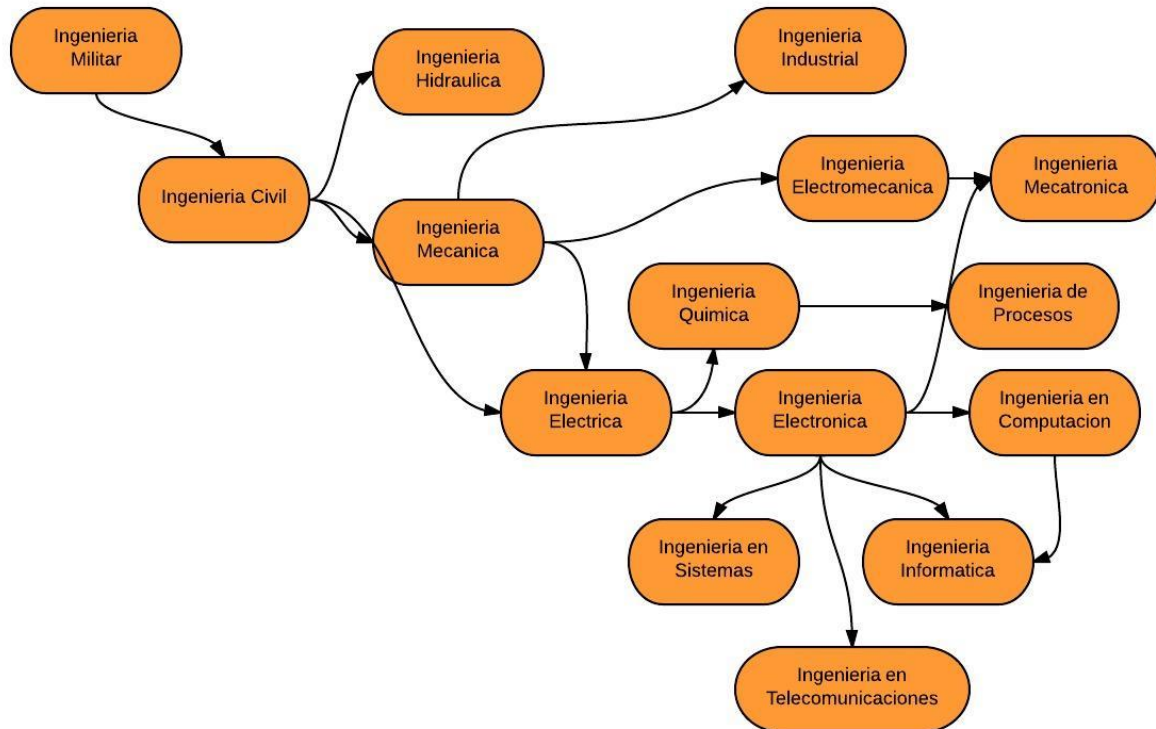
Descubriendo Definiciones de Software

Software (IEEE 610.12-1990): colección de programas, procedimientos, la documentación y datos asociados que determinan la operación de un sistema de computación. (<http://www.ieee.org/>).

El Software es un producto que diseñan y construyen los ingenieros de software.

Esto abarca **programas** que se ejecutan dentro de una computadora de cualquier tamaño y arquitectura, **documentos que comprenden:** formularios virtuales e impresos, datos que combinan números y textos y también incluye representaciones de información en audio video e imagen. (**Roger S. Pressman**).

Descubriendo Relaciones:



Descubriendo Definiciones de Ingeniería de Software

Ingeniería del Software, es el término que utilizó **Fritz Bauer** en la primera conferencia sobre desarrollo de software patrocinada por el Comité de Ciencia de la OTAN celebrada en Garmisch (Alemania), en octubre de 1968, previamente había sido utilizado por el holandés **Edsger Dijkstra** en su obra *The Humble Programmer*.

Al software sufría de ...

- Retrasos considerables en la planificación.
- Poca productividad.
- Elevadas cargas de mantenimiento.
- Demandas cada vez más desfasadas frente a las ofertas.

- Baja calidad y fiabilidad del producto.
- Dependencia de los realizadores.

Ingeniería del Software: Disciplina de la ingeniería que comprende todos los aspectos relacionados con la producción del software. Comprende todas las prácticas necesarias para desarrollar y entregar un software útil. *[Ian Sommerville]*.

Ingeniería de Software es el estudio de los principios y metodologías para desarrollo y mantenimiento de sistemas de software *[Zelkovitz, 1978]*.

Ingeniería del Software es la aplicación práctica del conocimiento científico en el diseño y construcción de programas de computadora y la documentación asociada requerida para desarrollar, operar (funcionar) y mantenerlos. Se conoce también como desarrollo de software o producción de software *[Bohem, 1976]*.

Ingeniería del Software es la aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación (funcionamiento) y mantenimiento del software; es decir, la aplicación de ingeniería al software. *[IEEE, 1993]*.

Descubriendo la Crisis del Software

Nos planteamos ¿Cómo fueron las etapas?

Primera Fase. Los albores (1945-1955)

Programar no es una tarea diferenciada del diseño de una máquina
Uso de lenguaje máquina y ensamblador.

Segunda Fase. El florecimiento (1955-1965)

Aparecen multitud de lenguajes
Se pensaba que era posible hacer casi todo.

Tercera Fase. La crisis (1965-1970)

Desarrollo inacabable de grandes programas
Ineficiencia, errores, coste impredecible
Nada es posible.

Cuarta Fase. Innovación conceptual (1970-1980)

Fundamentos de programación
Verificación de programas
Metodologías de diseño.

Quinta Fase. El diseño es el problema (1980-?)

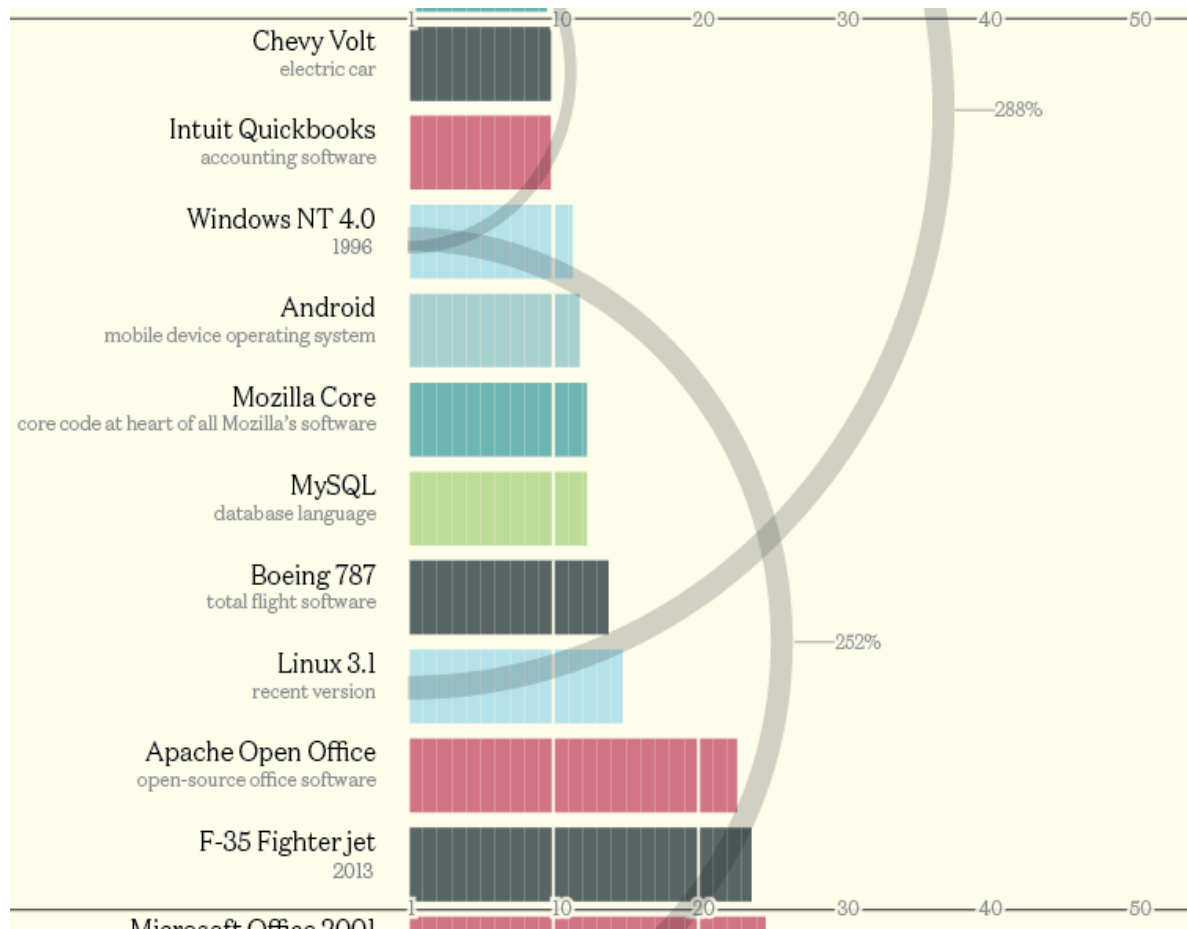
Entornos de programación
Especificación formal
Programación automática

Tópicos de Ingeniería del Software

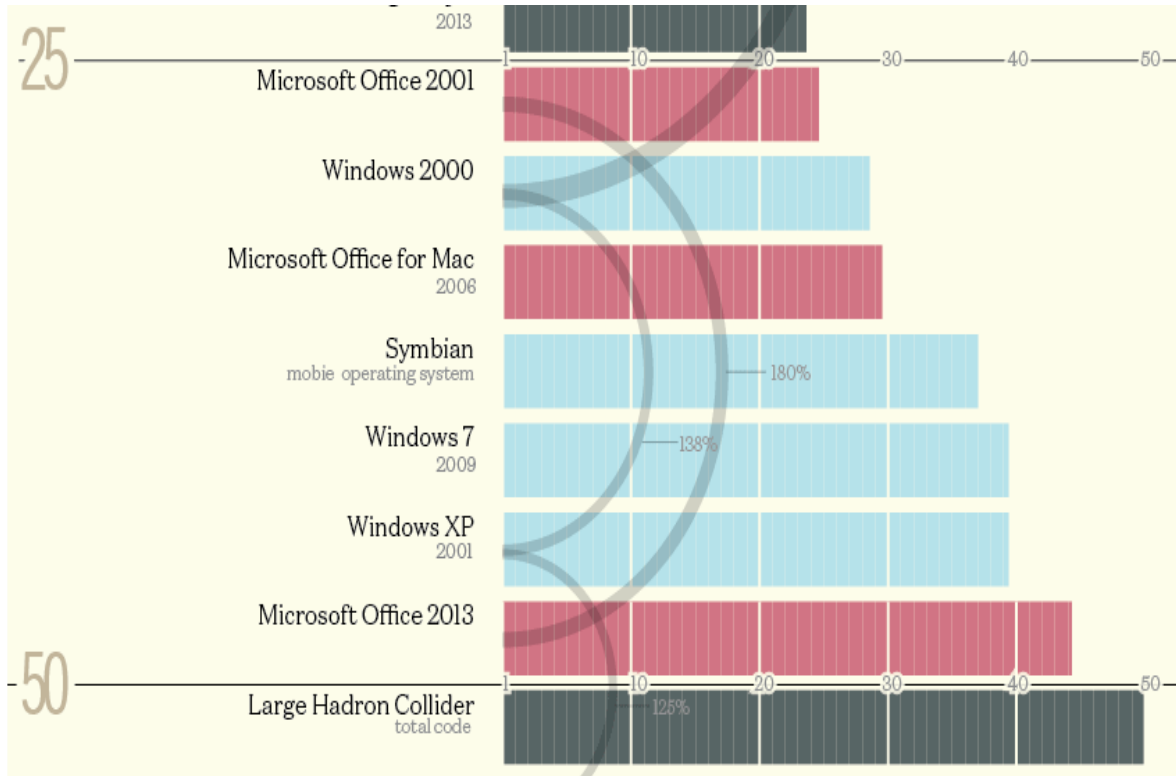
1. Conceptos fundamentales de resolución de problemas.
2. Proceso de desarrollo de software.
3. Especificaciones y requisitos de software.
4. Diseño e implementación de software.
5. Verificación y validación.

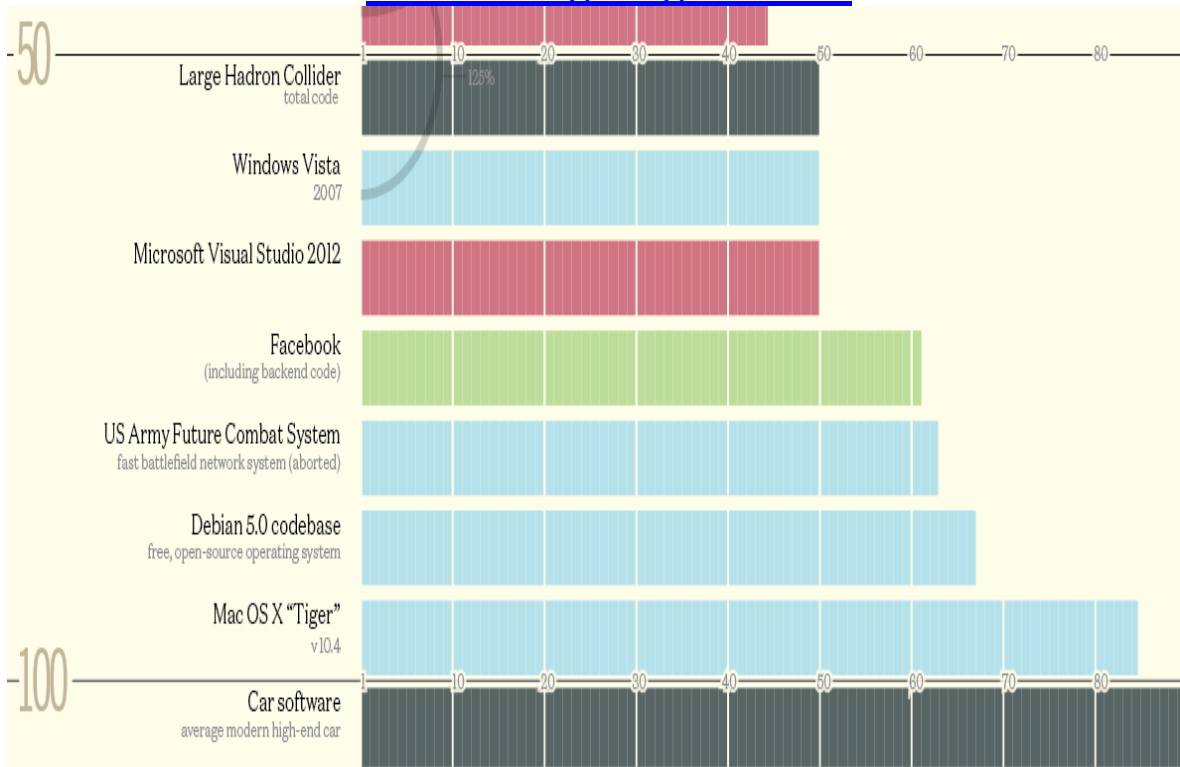
(Tomado de Roger S. Pressman).

Descubriendo una Reseña Histórica.



Ingeniería de Software I
Clase Número 1. Conceptos Generales.
Dr. Sergio D. Conde
sconde@unaj.edu.ar
drcondesergio@gmail.com





La Ingeniería del Software en la Web

La **ingeniería del Software en la web** es la aplicación de metodologías sistemáticas, disciplinadas y cuantificables al desarrollo eficiente, operación y evolución de aplicaciones de alta calidad en la WWW

La Ingeniería del Software en la web se debe al crecimiento desenfrenado que está teniendo la Web está ocasionando un impacto en la sociedad y el nuevo manejo que se le está dando a la información en las diferentes áreas en que se presenta ha hecho que las personas tiendan a realizar todas sus actividades por esta vía.

Uno de los aspectos más tenidos en cuenta, en el desarrollo de es sin duda alguna el Diseño Gráfico en la organización estructural del contenido. En la actualidad la web está sufriendo grandes cambios, que han obligado a expertos en el tema a utilizar herramientas y técnicas basadas en la Ingeniería del Software para poder garantizar el buen funcionamiento y administración de los sitios web.

Para garantizar el buen funcionamiento y mantenimiento de los sitios web, este debe contar con ciertos atributos y características que en conjunto forman un concepto muy importante, para alcanzar el éxito en cualquier organización, herramienta, y todo aquello que se pueda considerar como servicio. Dicho concepto es la calidad, que con atributos como, ***usabilidad, navegabilidad, seguridad, mantenibilidad***, entre otros, hace posible por un lado la eficiencia del artefacto web y por ende la satisfacción del usuario final.

Pero para tener artefactos de calidad, a esa misma se le debe planificar, programar y controlar, es decir la calidad no podrá ser agregada a un artefacto web o a cualquier otro producto, al final del proceso de desarrollo, si no que se deberá implementar durante todo el ciclo de vida del desarrollo.

Para finalizar el resultado de un proceso de calidad, podría arrojar recomendaciones para introducir mejoras, y la decisión final podría consistir en lanzar una nueva versión del sitio web o en modificar algunos atributos ausentes o pobremente diseñados.

Cabe destacar que la ingeniería de la web hace una diferencia entre un sitio web y un aplicativo, ya que ***la ingeniería de la web no se dedica a la construcción de sitios web si no a la construcción de aplicativos web***, la principal característica que los distingue (aplicativos de sitios web) es que los sitios web son sitios en la web en donde se publica contenido generalmente estático o un muy bajo nivel de interactividad con el **usuario**, mientras que ***los aplicativos son lugares con alto contenido de interactividad y funcionalidades que bien podrían ser de un software convencional, el aplicativo web más sencillo seria uno que contenga formularios y subiendo de nivel encontramos los que realizan conexión con bases de datos remotas, y administradores de contenidos entre otras.***

La ingeniería del software en la Web es la aplicación de metodologías sistemáticas, disciplinadas y cuantificables al desarrollo eficiente, operación y evolución de aplicaciones de alta calidad en la **World Wide Web**. En este sentido,

la ingeniería de la Web hace referencia a las metodologías, técnicas y herramientas que se utilizan en el desarrollo de **aplicaciones Web** complejas y de gran dimensión en las que se apoya la evaluación, diseño, desarrollo, implementación y evolución de dichas aplicaciones.

Los principales aspectos de la ingeniería del Software en la Web incluyen, entre otros, los siguientes temas:

- Diseño de procesos de negocio para aplicaciones web.
- Herramientas CASE para aplicaciones web.
- Generación de código para aplicaciones web.
- Desarrollo web colaborativo.
- Modelado conceptual de aplicaciones web.
- Diseño de Modelos de datos para sistemas de información web.
- Ingeniería web empírica.
- Entornos de desarrollo de aplicaciones web integrados.
- Herramientas de autor para contenido multimedia.
- Pruebas de rendimiento de aplicaciones basadas en web.
- Personalización y adaptación de aplicaciones web.
- Herramientas y métodos de prototipado.
- Control de calidad y pruebas de sistemas.
- Ingeniería de requisitos para aplicaciones web.
- Aplicaciones para la Web Semántica.
- Factorías de software para la web.
- Métodos, herramientas y automatización de pruebas para aplicaciones web.
- Aplicaciones web móviles y ubícuas.
- Usabilidad de aplicaciones web.
- Accesibilidad para la web.
- Metodologías de diseño web.

- Formación en ingeniería de la web.
- Diseño de interfaces de usuario.
- Métricas para la web, estimación de costes y medición.
- Gestión de proyectos web y gestión de riesgos.
- Desarrollo y despliegue de servicios web.

Se debe tener en cuenta que para la efectiva comunicación en la web, se tienen protocolos que es como el lenguaje para que se haga efectiva el intercambio de comunicación, vale la pena preguntarse, así para poder acceder a toda la información que nos puede suministrar Internet sólo debes poseer un servicio de algún proveedor de Internet un **navegador** como elemento fundamental.

La ingeniería del software, incluye nuevas metodologías de desarrollo esenciales para la administración de proyectos. Actualmente la ingeniería web ha adoptado también metodologías de la ingeniería del software y ha creado muchas nuevas. Debido a que la información es publicada para conocimiento de todo el mundo, hay que tener muy en cuenta aspectos sociales, jurídicos y **éticos** que pueden influir a la hora de la publicación.

Usualmente, las aplicaciones web son multidisciplinarias, ya que son construidas en un medio constantemente cambiante, donde los requerimientos son inestables, los equipos de desarrollo generalmente son pequeños, las comunidades de usuarios son más amplias que antes y la competencia ahora es a nivel mundial. En general, las **aplicaciones web**, necesitan ser funcionales, mantenibles, escalables y seguras.

Descubriendo la Industria del Software

En la euforia creada por la promesa de un paradigma económico nuevo, los inversores de Wall Street dieron a las pequeñas **empresas «punto-com»** estimaciones en billones de dólares antes de que éstas comenzasen a producir un dólar en ventas. (Tomado de Roger S. Pressman).

Han surgido nuevas industrias dirigidas por software y las antiguas que no se han adaptado a esta nueva tendencia están ahora amenazadas de extinción.

El gobierno de Estados Unidos ha mantenido un contencioso frente a la mayor compañía de la industria del software, como lo mantuvo hace poco tiempo cuando se movilizó para detener las actividades monopolísticas en las industrias del acero y del aceite. **(Tomado de Roger S. Pressman).**

Descubriendo algunos Números

Con ventas estimadas en 2972 millones de dólares, la Cámara de Empresas de Software y Servicios Informáticos (CESSI) publicó las cifras del sector, que ya cuenta con un mercado laboral compuesto por 64.300 trabajadores"

Ref : <http://www.lanacion.com.ar/1489817-la-industria-delsoftware-nacional-mantiene-su-crecimiento>

Con la venta de Whatsapp por US\$19.000 millones, la pareja de amigos supera a otros jóvenes emprendedores convertidos en millonarios com Janus Friis (38), fundador de Skype, quien vendió Skype a Microsoft por US\$6.000 millones, o Kevin Systrom (31) y Mike Krieger (28) que recibieron US\$1.000 de Facebook por Instagram.

http://www.bbc.co.uk/mundo/noticias/2014/02/140220_tecnologia_fundadores_whatsapp_aa.shtml

La Ley de Economía del Conocimiento

Las nuevas modificaciones de Ley de Economía del Conocimiento serán debatidas en la Cámara de Diputados de la Nación.

Luego de un arduo trabajo en conjunto con el Ministerio de Desarrollo Productivo, diversos funcionarios nacionales y representantes de la industria, CESSI, la

Cámara de la Industria Argentina del Software, celebra que el proyecto de Ley de la Economía del Conocimiento sea debatido por la Cámara de Diputados de la Nación en su próxima sesión.

En este último proyecto presentado al Congreso de La Nación, se pone principal foco en los beneficios fiscales, aplicando reducciones e incentivos a estos, como 60% de desgravación en Ganancias y un 70% de las contribuciones patronales como bono fiscal. Por su parte, los beneficios proponen duplicar las exportaciones, como también, el empleo en el sector, para todas aquellas empresas que incrementen su nómina de empleados de manera general y con beneficios adicionales basados en una perspectiva federal, inclusiva, de género y de alta capacitación.

Como parámetros fundamentales, la nueva Ley de la Economía del Conocimiento propone establecer requisitos de ingreso, evaluación y permanencia exigentes, diferenciados según el tamaño de cada empresa, sin que esta modificación implique dejar de promover el desarrollo de grandes empresas esenciales para el crecimiento de nuestro país. Cuando se trate del acceso al Régimen por parte de Micro-Empresas, con una antigüedad menor a 3 años o que no cuenten con facturación alguna, según el artículo 2° de la Ley N° 24.467, deberán acreditar el desarrollo por cuenta propia de alguna de las actividades mencionadas en el artículo 2° de la presente Ley.

Estamos convencidos de que la reglamentación de esta nueva ley impulsará y fomentará la competitividad, exportaciones, apertura de nuevos mercados, la generación de empleo y la transformación productiva, de todas las empresas del sector, startups, PyMEs y grandes compañías.

De esta manera, seguimos apostando al desarrollo de un país inclusivo y federal, defendiendo esta industria que posee un amplio potencial para nuestros trabajadores y la tecnología local, como herramienta que ayuda a combatir la pobreza,

Ingeniería de Software I

Clase Número 1. Conceptos Generales.

Dr. Sergio D. Conde

sconde@unaj.edu.ar

drcondesergio@gmail.com

indigencia y hambre existentes en la nación. Para lograrlo, CESSI se encuentra a plena disposición de los legisladores para discutir todos los detalles del impacto de esta nueva versión de la ley en nuestra industria. (<https://cessi.org.ar/ver-noticias-las-nuevas-modificaciones-de-ley-de-economia-del-conocimiento-seran-debatidas-en-la-camara-de-diputados-de-la-nacion-2412>)

Código de Ética de Ingeniería de Software.

Los ingenieros de software debieran obligarse a hacer del análisis, especificación, diseño, desarrollo, pruebas y mantenimiento del software una profesión respetada y beneficiosa. En concordancia con la obligación con el bienestar, salud y seguridad de la sociedad, los ingenieros del software debieran adherirse a los Ocho Principios siguientes:

1. **Sociedad.** Los ingenieros de software actuarán en forma congruente con el interés social.
2. **Cliente. y empresario.** Los ingenieros de software actuarán de manera que se concilien los mejores intereses de sus clientes y empresarios, congruentemente con el interés social.
3. **Producto.** Los ingenieros de software asegurarán que sus productos y modificaciones correspondientes cumplen los estándares profesionales más altos posibles.
4. **Juicio.** Los ingenieros de software mantendrán integridad e independencia en su juicio profesional.
5. **Administración.** Los ingenieros de software gerentes y líderes promoverán y se suscribirán a un enfoque ético en la administración del desarrollo y mantenimiento de software.
6. **Profesión.** Los ingenieros de software incrementarán la integridad y reputación de la profesión congruentemente con el interés social.
7. **Colegas.** Los ingenieros de software apoyarán y serán justos con sus colegas.
8. **Personal.** Los ingenieros de software participarán toda su vida en el aprendizaje relacionado con la práctica de su profesión y promoverán un enfoque ético en la práctica de la profesión.

Las computadoras tienen un papel central cada vez mayor en el comercio, industria, gobierno, medicina, educación, entretenimiento, y sociedad. Los ingenieros de software son aquellos que contribuyen, mediante la participación directa o enseñanza, al análisis, especificación, diseño, desarrollo, certificación, mantenimiento y pruebas de sistemas de software. Debido a sus funciones en el desarrollo de sistemas de software, los ingenieros de software tienen suficientes oportunidades para causar beneficio o generar daño y para habilitar o influenciar a otros a causar daño o beneficio. Para asegurar, en la medida de lo posible, que sus esfuerzos se utilizarán para hacer el bien, los ingenieros de software deben comprometerse a hacer de la ingeniería del software una profesión benéfica y respetada. De acuerdo con tal compromiso, los ingenieros de software deberán adherirse al siguiente Código De Ética Y Práctica Profesional. El Código contiene ocho Principios relacionados con la conducta y las decisiones tomadas por los ingenieros de software profesionales, bien sean profesionales en ejercicio, educadores, administradores, supervisores y directivos, así como educandos y estudiantes de la profesión. Los Principios identifican las relaciones éticamente responsables en las que los individuos, grupos y organizaciones participan, y las principales obligaciones de tales relaciones. Las Cláusulas de cada Principio son ejemplos de algunas de las obligaciones incluidas en estas relaciones. Estas obligaciones se fundamentan en las características humanas del ingeniero de software en los deberes hacia las personas que se ven afectadas por el trabajo del ingeniero de software, y en los elementos peculiares de la práctica de la ingeniería del software. El Código prescribe éstas como obligaciones de cualquiera que se identifique como ingeniero de software o que aspire a serlo. No se pretende que se utilicen partes individuales del Código aisladamente, para justificar errores por omisión o comisión. La lista de Principios y Cláusulas no es exhaustiva. Las Cláusulas no deben leerse como la frontera entre lo aceptable y lo inaceptable en todas las situaciones prácticas de la conducta profesional. El Có-

El código no es un simple algoritmo ético que genera decisiones éticas. En algunas situaciones los estándares pueden entrar en conflicto entre sí o con estándares de otras fuentes. Estas situaciones requieren que el ingeniero de software utilice su juicio ético para actuar en la manera más congruente con el espíritu del Código de Ética y Práctica Profesional, teniendo en cuenta las circunstancias. Los conflictos éticos pueden manejarse mediante una consideración cuidadosa de los principios fundamentales, más bien que apoyándose ciegamente en reglamentos detallados. Estos Principios deberían influenciar a los ingenieros de software a considerar ampliamente a quién se ve afectado por su trabajo; a examinar si ellos o sus colegas tratan al resto de las personas con el debido respeto; a reflexionar en cómo la sociedad vería sus decisiones si estuviera bien informada; a analizar cómo el menos favorecido será afectado por su decisión; y a considerar si sus actos lo juzgarían como un valioso profesional ideal que trabaja como ingeniero de software. En todas estas valoraciones la preocupación por la salud, seguridad y bienestar público es primordial; esto es, el "Interés Social" es central en este Código.

El contexto dinámico y exigente de la ingeniería de software requiere un código que sea adaptable y relevante a las nuevas situaciones a medida que ocurran. Sin embargo, incluso en esta generalidad, el Código proporciona apoyo a los ingenieros del software y administradores que necesitan actuar positivamente en un caso específico documentando la postura ética de la profesión. El Código proporciona un fundamento ético al cual los individuos de un equipo o el propio equipo pueden acudir.

El Código también ayuda a definir aquellas cuestiones que son éticamente impropias de solicitar a un ingeniero de software o equipo de ingenieros de software.

El Código no está simplemente orientado a identificar la naturaleza de los actos cuestionables, sino que también tiene una función educativa importante. Puesto

que este código representa el consenso de la profesión en cuestiones éticas, es un medio para educar tanto a la sociedad como a los futuros profesionales acerca de las obligaciones éticas de todos los ingenieros de software.

Principios:

Principio 1. Sociedad.

- 1.1. Los ingenieros de software actuarán de forma congruente con el interés social. Particularmente, los ingenieros de software deberán, cuando sea apropiado:
- 1.2. Aceptar la responsabilidad total de su trabajo.
- 1.3. Moderar los intereses del ingeniero de software, el empresario, el cliente y los usuarios con el bienestar social.
- 1.4. Aprobar software sólo si se tiene una creencia bien fundamentada de que es seguro, cumple las especificaciones, pasa las pruebas apropiadas y no reduce la calidad de vida, la privacidad o daña el medio ambiente. El efecto último del trabajo deberá ser el bien social.
- 1.5. Exponer a las personas o autoridades apropiadas cualquier daño real o potencial al usuario, a la sociedad o el medio ambiente, que razonablemente se cree que está asociado con el software o documentos relacionados.
- 1.6. Cooperar en los esfuerzos para solucionar asuntos importantes de interés social causados por el software su instalación, mantenimiento, soporte o documentación.
- 1.7. Ser justo y veraz en todas las afirmaciones, particularmente las públicas, relativas al software o documentos asociados, métodos y herramientas.

Considerar incapacidad física, distribución de recursos, desventajas económicas y otros factores que pueden reducir el acceso a los beneficios del software.

- 1.8. Estar motivado a ofrecer voluntariamente asistencia técnica a buenas causas y contribuir a la educación pública relacionada con esta profesión.

Principio 2. Cliente y empresario.

Los ingenieros de software actuarán de manera que se concilien lo mejores intereses de sus clientes y empresarios, congruentemente con el interés social. Particularmente, los ingenieros de software deberán, cuando sea apropiado:

2.01 Prestar servicios en sus áreas de competencia, siendo honestos y francos acerca de sus limitaciones en su experiencia y educación.

2.02 No usar conscientemente software que se obtiene o retiene ya sea ilegalmente o sin ética.

2.03 Usar la propiedad de un cliente o empresario sólo en forma propiamente autorizada y con el conocimiento y consentimiento del cliente o empresario.

2.04 Cuando se requiera, asegurar que cualquier documento en el que se confía ha sido aprobado por alguien autorizado para aprobarlo.

2.05 Mantener secreta cualquier información confidencial obtenida en su labor profesional, donde tal confidencialidad es congruente con el interés social y congruente con la ley.

2.06 Identificar, documentar, reunir evidencia y reportar oportunamente al cliente o al empresario si, en su opinión, un proyecto tiene probabilidades de fracasar, de ser muy costoso, de violar la ley de propiedad intelectual o ser problemático

de cualquier otro modo.

2.07 Identificar, documentar y reportar al cliente o empresario asuntos significativos de interés social, de los cuales se tiene conocimiento, acerca del software o documentos relacionados.

2.08 Rechazar trabajos externos que vayan en detrimento del trabajo que se realiza para su patrón primario.

2.09 No promover intereses adversos a su empresario o cliente, a menos que se comprometa un interés ético más alto; en ese caso, informar al empresario u otra autoridad apropiada del interés ético en cuestión.

Principio 3. Producto.

Los ingenieros de software asegurarán que sus productos y sus modificaciones correspondientes cumplen los estándares profesionales más altos posibles. Particularmente, los ingenieros de software deberán, cuando sea apropiado:

3.01 Procurar la alta calidad, costos aceptables y una agenda razonable asegurando que los costos y beneficios significativos son claros y aceptados por el empresario y el cliente, y están disponibles para consideración del usuario y de la sociedad.

3.02 Asegurar que las metas y objetivos para cualquier proyecto que se propone o en el que se trabaja sean adecuados y alcanzables.

3.03 Identificar, definir y atender asuntos éticos, económicos, culturales, legales y ambientales relacionados a los proyectos de trabajo.

3.04 Asegurar que se está calificado, con una combinación apropiada de educación, adiestramiento y experiencia para cualquier proyecto en que se trabaje o que se proponga trabajar.

3.05 Asegurar que se usan los métodos apropiados en cualquier proyecto en el que se trabaja o se propone trabajar.

3.06 Tratar de seguir los estándares profesionales más adecuados, siempre que estén disponibles, para el proyecto en que se trabaja. Sólo en caso de que hubiera una justificación ética o técnica mayor, se permitiría alterar dichos estándares.

3.07 Esforzarse por entender completamente las especificaciones del software en el que se trabaja.

3.08 Asegurar que las especificaciones del software en el que se trabaja están bien documentadas, satisfacen los requerimientos del usuario y cuentan con las aprobaciones adecuadas.

3.09 Asegurar estimaciones cuantitativas realistas de costos, agenda, personal, calidad y resultados de cualquier proyecto en el que se trabaja o se propone trabajar, proporcionando una evaluación de la incertidumbre de esas estimaciones.

3.10 Asegurar que las pruebas, depuración, revisión del software y documentos relacionados con los que se trabaja sean adecuados

3.11 Asegurar que la documentación sea adecuada, incluyendo problemas significativos encontrados y soluciones adoptadas, para cualquier proyecto en el que se trabaja.

3.12. Trabajar para desarrollar software y documentos relacionados que respeten

la privacidad de aquellos a quienes está dirigido este software.

3.13 Ser cuidadoso para usar sólo datos precisos derivados por medios éticos y legales, y usarlos sólo de las maneras propiamente autorizadas.

3.14 Mantener la integridad de los datos siendo sensible a aquellos inexactos u obsoletos.

3.15 Tratar todas las formas de mantenimiento de software con el mismo profesionalismo que los desarrollos nuevos.

Principio 4. Juicio.

Los ingenieros de software mantendrán integridad e independencia en su juicio profesional. Particularmente, los ingenieros de software deberán, cuando sea apropiado:

4.01 Moderar todos los juicios técnicos por la necesidad de apoyar y mantener los valores humanos.

4.02 Endosar documentos únicamente cuando han sido preparados bajo su supervisión o dentro de sus áreas de competencia y con los cuales se está de acuerdo.

4.03 Mantener objetividad profesional con respecto a cualquier software o documento relacionado del cual se le pidió una evaluación.

4.04 No involucrarse en prácticas financieras fraudulentas tal como corrupción, facturación doble u otras prácticas financieras impropias.

4.05 Exponer a todas las partes involucradas aquellos conflictos de interés que no puedan evitarse o evadirse razonablemente.

4.06 Negarse a participar como miembro o asesor en organismos profesionales, privados o gubernamentales vinculados en asuntos relacionados con software donde sus empresarios o clientes pudieran tener conflictos de intereses no declarados todavía.

Principio 5. Administración.

Los ingenieros de software gerentes y líderes promoverán y se suscribirán a un enfoque ético en la administración del desarrollo y mantenimiento de software. Particularmente, los ingenieros de software administrando o dirigiendo deberán, cuando sea apropiado:

5.01 Asegurar una buena administración para cualquier proyecto en el cual trabaje, incluyendo procedimientos efectivos para promover la calidad y reducir riesgos.

5.02 Asegurar que los ingenieros de software estén informados de los estándares antes de sujetarse a ellos.

5.03 Asegurar que los ingenieros de software conozcan las políticas y procedimientos del empresario para proteger las contraseñas, archivos e información que es confidencial al empresario o confidencial a otros.

5.04 Asignar trabajo sólo después de tomar en cuenta contribuciones adecuadas de educación y experiencia moderadas con un deseo de continuar esa educación y experiencia.

5.05 Asegurar estimaciones de costos, agendas, personal, calidad y resultados cuantitativamente realistas en cualquier proyecto que trabaje o se propone trabajar, proporcionando una evaluación de la incertidumbre de esas estimaciones.

5.06 Atraer ingenieros de software potenciales sólo bajo una descripción completa y precisa de las condiciones del empleo.

5.07 Ofrecer una remuneración justa y equitativa.

5.08 No impedir injustamente que alguna persona ocupe una posición para la cual está perfectamente calificada.

5.09 Asegurar que exista un acuerdo equitativo en lo referente a la propiedad de cualquier software, proceso, investigación, documentación u otra propiedad intelectual a la cual el ingeniero de software ha contribuido.

5.10 Tomar medidas prudentes en procesos legales vinculados a la violación de la política de un empresario o de este código.

5.11 No pedir a un ingeniero de software hacer algo incongruente con este código.

5.12 No castigar a nadie por expresar temas éticos relativos a cualquier proyecto.

Principio 6. Profesión.

Los ingenieros de software incrementaran la integridad y reputación de la profesión congruentemente con el interés social. Particularmente, los ingenieros de software deberán, cuando sea apropiado:

6.01 Ayudar a desarrollar un ambiente organizacional favorable para actuar éticamente.

6.02 Promover el conocimiento público de la ingeniería de software.

6.03 Extender el conocimiento de la ingeniería de software participando apropiadamente en organizaciones, reuniones y publicaciones profesionales.

6.04 Apoyar, como miembros de una profesión, a otros ingenieros de software que se esfuercen por seguir este código.

6.05 No promover el interés propio a costa de la profesión, cliente o empresario.

6.06 Obedecer todas las leyes que gobiernan su trabajo, salvo en circunstancias excepcionales, donde tal obediencia es incongruente con el interés social.

6.07 Ser preciso en la descripción de las características del software en el que trabaja, evitando no sólo declaraciones falsas, sino también declaraciones que podrían ser razonablemente asumidas como especulativas, vacías, fraudulentas, engañosas o dudosas.

6.08 Tomar la responsabilidad de detectar, corregir y reportar errores en el software y documentos asociados en los que se trabaja.

6.09 Asegurar que los clientes, empresarios y supervisores conozcan el compromiso de los ingenieros de software con este código de ética, y las subsecuentes ramificaciones de tal compromiso.

6.10 Evitar asociaciones con negocios y organizaciones que estén en conflicto con este código.

6.11 Reconocer que las violaciones de este código son incongruentes con ser un ingeniero de software profesional.

6.12 Hablar seriamente con la gente involucrada cuando se detecten violaciones significativas de este código, a menos que sea imposible, contraproductivo o peligroso.

6.13 Reportar las violaciones significativas de este código a las autoridades correspondientes cuando está claro que consultar con la gente involucrada en estas violaciones es imposible, contra productivo o peligroso.

Principio 7. Colegas.

Los ingenieros de software apoyarán y serán justos con sus colegas. Particularmente, los ingenieros de software deberán, cuando sea apropiado:

7.01 Motivar a sus colegas a sujetarse a este código.

7.02 Ayudar a sus colegas en el desarrollo profesional.

7.03 Reconocer completamente el trabajo de otros y abstenerse de atribuirse méritos indebidos.

7.04 Revisar el trabajo de otros en forma objetiva, sincera y propiamente documentada

7.05 Escuchar equitativamente las opiniones, preocupaciones y quejas de un colega.

7.06 Ayudar a sus colegas a que estén totalmente alertas a los actuales estándares incluyendo políticas y procedimientos de protección de contraseñas, archivos, información confidencial y las medidas de seguridad en general.

7.07 No intervenir injustamente en la carrera de algún colega; sin embargo, el interés del empresario, del cliente o el interés social puede conducir a ingenieros de software, de buena fe, a cuestionar la competencia de un colega.

7.08 En situaciones fuera de sus propias áreas de competencia, solicitar las opiniones de otros profesionales que tengan competencia en esa área.

Principio 8. Personal.

Los ingenieros de software participarán toda su vida en el aprendizaje relacionado con la práctica de su profesión y promoverán un enfoque ético en la práctica de la profesión. Particularmente, los ingenieros de software deberán, cuando sea apropiado:

8.01 Mejorar su conocimiento de los avances en el análisis, especificación, diseño, desarrollo, mantenimiento, pruebas del software y documentos relacionados, junto con la administración del proceso de desarrollo.

8.02 Mejorar su habilidad para crear software seguro, confiable, útil y de calidad a costos razonables y en un tiempo razonable.

8.03 Mejorar su habilidad para producir documentación precisa, informativa y bien redactada.

8.04 Mejorar su comprensión del software de los documentos con que se trabaja y del medio ambiente donde serán usados.

8.05 Mejorar su conocimiento de los estándares relevantes y de las leyes que gobiernan el software y los documentos con que se trabaja.

8.06 Mejorar su conocimiento de este código, su interpretación y su aplicación al trabajo.

8.07 No tratar injustamente a nadie debido a prejuicios irrelevantes.

8.08 No influenciar a otros a emprender alguna acción que involucre una violación de este código.

8.09 Reconocer que las violaciones personales de este código son incongruentes con ser un ingeniero de software profesional.

Para terminar desarrollaremos conceptos muy importantes para tener en cuenta en un Software:

¿Qué significa construir software?

Cómo se construiría cualquier producto donde **el cliente quede satisfecho**, aplicando un proceso que necesariamente tiene que llevar a un resultado de alta calidad y además debe satisfacer las necesidades de la gente que usa este producto, todo esto aplicando un enfoque de la ingeniería de software.

Por lo tanto podemos determinar que el **el cliente quede satisfecho**.

El programador solitario de antaño ha sido reemplazado por un equipo de especialistas del software, cada uno centrado en una parte de la tecnología requerida para entregar una aplicación concreta. Y de este modo, las cuestiones que se preguntaba el programador solitario son las mismas cuestiones que nos preguntamos cuando construimos sistemas modernos basados en computadoras
(Tomado de Roger S. Pressman)

¿Que se obtiene de la construcción?

- **Según Ingeniero de software:** Programas, documentos, datos, que configuran el software.
- **Según el usuario final o el cliente:** Información resultante que hace mucho más fácil la vida.

Descubriendo Características del Software

1: Mantenibilidad.

Debemos escribir software de tal forma que pueda evolucionar para cumplir las necesidades del cliente.

El cambio en el software es una consecuencia inevitable.



¿Intentarías agregar una conexión nueva?

2. Confiabilidad

La confiabilidad del software se transmite en fiabilidad, protección y seguridad. El software no debe causar daños físicos ni económicos en caso de una falla del sistema.

3. Eficiencia

EL software debe ser diseñado de manera que **no desgaste los recursos del sistema como memoria o procesador**. (Tiempos de respuesta, tiempo de procesamiento, etc).



4. Usabilidad

El software que diseñamos debe ser de fácil uso para el usuario final que fue diseñado, no debería generar esfuerzo adicional.(Interfaz y documentación simples).