

Ingeniería del Software I

2º

TEMA 1

Principios básicos. Sentido y finalidad de la ingeniería del software. Sistemas sociotécnicos



- «La vida es demasiado corta para **construir algo que nadie quiera**». Ash Manrya
- «Las especies que sobreviven no son las más fuertes sino las que **mejor se adaptan al cambio**». Darwin
- **Esta asignatura es muy importante para trabajar y para vivir.**
- Estad muy atentos a **clase para comprender + complementar con libro Sommerville.**
- **Evaluación:** 1 parcial + 1 final. Junio.
- Asignatura **columna vertebral**: se extiende 2º y 3º

Objetivos

Introducir en la ingeniería del software y ofrecer un **marco conceptual** para entender la Ingeniería de Software.

- Qué es la Ingeniería del Software y por qué es importante
- El desarrollo de distintos sistemas de software puede requerir distintas técnicas de ingeniería de software.
- Existen conflictos éticos y profesionales para los ingenieros del software

Índice

1. Introducción
2. Desarrollo de software profesional
 - Ingeniería del Software
 - Diversidad de la ingeniería del software
 - Ingeniería del software y la Web
3. Ética en la ingeniería del software

1. Introducción

- Imposible operar sin software
 - Infraestructuras del mundo – software de sistemas
 - Fabricación industrial
 - Sistema financiero
 - Móviles, relojes, pulseras de actividad
 - Domótica, productos de compras – software de control
 - Televisión, cine, el mundo del juego
- La ingeniería del software es esencial para el funcionamiento de las sociedades a nivel internacional.

- **Los sistemas de software son abstractos e intangibles**
 - No están restringidos por las propiedades de los materiales
 - No están regidos por las leyes físicas ni por procesos de fabricación.
 - No existen límites naturales a su potencial.
 - Y justamente, debido a la **falta de restricciones físicas**, los sistemas pueden volverse rápidamente:
 - Muy complejos
 - Difíciles de entender
 - Costosos de cambiar
- Hay muchos tipos de sistemas de software desde los simples sistemas embebidos hasta los complejos sistemas de información mundial.
 - **No tiene sentido buscar notaciones, métodos, técnicas universales para la ingeniería del software ya que diferentes tipos de software requieren distintos enfoques.**
 - Ejemplo: ¿desarrollar un sistema organizacional de información es completamente diferente a un controlador para un instrumento científico!!
 - **Pero todos necesitan INGENIERÍA DEL SOFTWARE aunque con distintas técnicas.**

- Por experiencia, muchos proyectos de software salen mal, y contienen errores de software. Hay que tener en cuenta que muchos de los errores de software son consecuencia de dos factores:
 - **Demandas crecientes:**
 - La demanda cambia.
 - Los sistemas han de construirse y distribuirse más rápidamente.
 - Se requieren sistemas más grandes y complejos.
 - Nuevas capacidades que antes se consideraban imposibles.
 - → Los métodos existentes de ISW no pueden enfrentar la situación, han de desarrollar nuevas técnicas.
 - **Expectativas bajas:**
 - No se han utilizado técnicas de ISW para programas de cómputo pequeños. Muchas veces no se han utilizado técnicas siquiera.
 - Conclusión: software más costoso y menos confiable.
 - Hay que educar y capacitar en ISW para resolver este problema.
- Sin ISW no se habría explorado el espacio ni se tendría Internet o las telecomunicaciones modernas.

1968. El concepto de Ingeniería del Software se propuso originalmente en una conferencia realizada para discutir lo que entonces se llamaba “la crisis del software”.

- Quedó claro que los **enfoques individuales** al desarrollo de programas no escalaba hacia los grandes y complejos sistemas de software. No eran confiables, más costosos en esfuerzo y en tiempo.

2. Desarrollo de software profesional

- Se desarrollan muchos programas como hobbies pero **la gran mayoría del desarrollo de software es una actividad profesional**
 - Donde el software se realiza para propósitos de negocios específicos para su inclusión en otros dispositivos o como productos de software.
 - El software profesional destinado a usarse por alguien más que por el desarrollador, se lleva a cabo por equipos en vez de individualmente. Se mantiene y cambia a lo largo de su vida.
- **La ISW busca apoyar el desarrollo del software profesional, en lugar de la programación individual.**
 - Incluye técnicas que apoyan la especificación, el diseño, y la evolución del programa. Normalmente no relevantes para la programación personal.

- Cuando se habla de ISW esto **no sólo se refiere a los programas en sí**, sino también a toda la **documentación asociada y los datos de configuración requeridos** para hacer que estos programas **operen** de manera correcta.
 - Un sistema de software desarrollado profesionalmente suele incluir varios programas, archivos de configuración, documentación del programa, de usuario, webs para descargar últimas versiones del producto.

PREGUNTAS FRECUENTES

Pregunta	Respuesta
¿Qué es software?	Programas de cómputo y documentación asociada. Los productos de software se desarrollan para un cliente en particular o para un mercado en general
¿Cuáles son los atributos del buen software?	El buen software debe entregar al usuario la funcionalidad y el desempeño requeridos; y debe ser mantenible, eficiente, confiable y aceptable.

T1. 2. Desarrollo de software profesional

¿Qué es Ingeniería del Software?	La Ingeniería del Software es una disciplina de la Ingeniería que se interesa por todos los aspectos de la producción del software
¿Cuáles son las actividades fundamentales de la Ingeniería del Software?	Especificación, desarrollo y diseño, validación e implantación y evolución del software
¿Cuál es la diferencia entre Ingeniería del Software y Ciencias de la Computación?	Las Ciencias de la Computación se enfocan en teoría y en fundamentos, mientras que la ISW se enfoca en el sentido práctico del desarrollo y en la distribución de software.
¿Cuál es la diferencia entre Ingeniería del Software y Ingeniería de Sistemas?	La Ingeniería de sistemas se interesa por todos los aspectos del desarrollo de sistemas basados en computadoras , incluidos hardware, software e ingeniería de procesos. La ISW es parte de este proceso en general.
¿Cuáles son los mejores métodos y técnicas de la Ingeniería del Software?	Aun cuando todos los proyectos de software deben gestionarse y desarrollarse de manera profesional, existen diferentes técnicas adecuadas para distintos tipos de sistema. No puede decirse que un método sea mejor que otro.
¿Qué diferencias ha marcado la Web a la Ingeniería del Software?	La Web ha llevado a la disponibilidad de servicios de software y a la posibilidad de desarrollar sistemas basados en servicios distribuidos . El desarrollo basado en web ha conllevado bastantes avances y también a reutilización .

- Existen dos tipos de productos de software:
 - **Productos genéricos**. Consisten en sistemas independientes que se producen en una organización de desarrollo.
 - Se venden en mercado abierto a cualquier cliente que desee comprarlo.
 - La empresa controla la especificación del producto, lo desarrolla y lo vende.
 - **Productos personalizados** (o a la medida). Son sistemas que están destinados para un cliente en particular.
 - El cliente controla la especificación. El contratista desarrolla el programa específico para dicho cliente.
 - **Opción mixta**. Cada vez es más difícil distinguirlos, surgen productos como SAP, gestores documentales, que se compran y luego se especifican para el cliente en cuestión.
- La calidad no tiene que ver sólo con lo que hace el software, con los atributos funcionales sino también con el tiempo de respuesta y la comprensibilidad.

T1. 2. Desarrollo de software profesional

- Atributos esenciales de buen software

Características del producto	Descripción
Mantenimiento	El software debe escribirse de tal forma que pueda evolucionar para satisfacer las necesidades cambiantes de los clientes. Atributo crítico. Empresas varían constantemente
Confiabilidad y seguridad	Incluye un rango de características que abarcan fiabilidad, seguridad y protección. El software confiable no tiene que causar daño físico ni económico en caso de error del sistema. Un usuario malintencionado no ha de tener posible acceso al sistema
Eficiencia	El SW no tiene que desperdiciar recursos del sistema, como memoria y ciclos del procesador. Incluye capacidad de respuesta, tiempo de procesamiento, utilización de memoria, etc.
Aceptabilidad	El SW debe ser aceptable al tipo de usuarios que quienes se diseña. Esto significa que necesita ser comprensible, utilizable y compatible con otros sistemas que ellos usan.

2.1 Ingeniería del Software

- La Ingeniería del Software es una **disciplina de la ingeniería** que se interesa por **todos los aspectos de la producción de software**, desde las primeras etapas de la especificación del sistema hasta el mantenimiento del sistema después de que se pone en operación.
- En esta definición se presentan dos frases clave:
 - **Disciplina de ingeniería**: para hacer que las cosas funcionen, los ingenieros aplicamos teorías, métodos y herramientas donde es adecuado. Incluso se buscan soluciones dentro de las limitaciones organizacionales y financieras.
 - **Todos los aspectos de la producción del software**: la ISW no sólo se interesa por los procesos técnicos del desarrollo de software sino también incluye actividades como administración del proyecto y el desarrollo de herramientas, métodos y teorías para apoyar la producción de SW.
- Los ingenieros no deben ser perfeccionistas. La ingeniería busca obtener resultados de calidad requerida dentro de la fecha y del presupuesto.

- La Ingeniería del Software es **importante** por dos razones:
 - Cada vez la **sociedad se apoya más en sistemas de software**. Se requiere producir sistemas confiables de forma rápida y económica.
 - **Resulta más barato a largo plazo usar métodos y técnicas** de ingeniería del software formales.
- El enfoque sistemático que se usa en la ingeniería del software se conoce como proceso del software. Es consecuencia de actividades que conducen a la elaboración de un producto software.
 - **Existen cuatro actividades fundamentales que son comunes a todos los procesos de software.**
 - **Especificación del software.** Clientes e ingenieros definen el SW que se producirá y las **restricciones** en su operación.
 - **Diseño y desarrollo del software**, donde se **diseña** y **programa** el software.
 - **Validación e implantación del software**, donde se **verifica** el software para asegurar que sea lo que el cliente quiere y se implanta.
 - **Evolución del software** donde se modifica el software para reflejar los requerimientos cambiantes del cliente y del mercado.



- La ISW se **relaciona con las ciencias de la computación y la ingeniería de sistemas**:
 - Las ciencias de la computación se interesan por las teorías y los métodos que subyacen en las computadoras y los sistemas de software, en tanto que la ingeniería de software se preocupa por los asuntos prácticos de la **producción del software**.
 - La ingeniería de sistemas se interesa por **todos los aspectos del desarrollo y la evolución de complejos sistemas**, donde el software tiene un papel principal. Por lo tanto, la ingeniería de sistemas se preocupa por el desarrollo del hardware, el diseño de políticas y procesos, la implementación del sistema, **así como por la ISW**.

- Existen **tres problemas generales** que afectan a muy diversos tipos de software:
 - **Heterogeneidad**: será cada vez más frecuente **integrar nuevo software con muchos sistemas legados más viejos** o nuevos escritos en otros lenguajes. El reto aquí es desarrollar técnicas para construir software confiable que sea suficientemente flexible para poderse integrar con facilidad.
 - **Cambio empresarial y social**: la **sociedad y los negocios cambian** de forma increíblemente rápida y necesitan nuevas técnicas de ISW que **se adapten y no consuman tiempo (con una buena gestión del cambio)**. La entrega de los nuevos sistemas **tarda más de lo planeado**. Requieren evolucionar de modo que se reduzca el tiempo necesario para que el software dé valor a sus clientes.
 - **Seguridad y confianza**: el software está vinculado a todos los aspectos de la vida, es **esencial confiar** en dicho software y aún más en experiencias previas malas. Especialmente importante en sistemas remotos o web. Es necesario **asegurarse que usuarios malintencionados** no puedan atacar el software y que se conserve la seguridad en el acceso a la información.

Desafíos del SW:

Plazo (en tiempo), coste y calidad.



T1. 2.2 Diversidad de la Ingeniería del Software

- Factor más significativo en la determinación de la ingeniería del software – tipo de aplicación:

- **Aplicaciones independientes.** Corren en un ordenador local, incluyen toda la funcionalidad y no requieren conectarse a una red. Ejemplos: programas de CAD, manipulación de fotografías, etc.
- **Aplicaciones interactivas basadas en transacción.** Aplicaciones que se ejecutan en un ordenador remoto y al que los usuarios acceden desde sus propios ordenadores. En ellas se incluyen aplicaciones Web como comercio electrónico. Este tipo de aplicación también suelen ser sistemas empresariales donde se proporcionan programas cliente o navegadores web para acceder a la información de la empresa situada en la nube.
- **Sistemas de control embebido.** Sistemas de control de software que regulan y gestionan dispositivos hardware. Numéricamente, quizás existan más sistemas de control embebido que cualquier otro tipo de sistema. Ejemplo: software que controla frenos antibloqueo, horno que controla proceso cocinado.
- **Sistema de procesamiento en lotes.** Sistemas empresariales que se diseñan para procesar datos en grandes lotes (batch) Procesan gran cantidad de entradas individuales para crear las salidas correspondientes. Ejemplo: facturación telefónica, pago de salario.
- **Sistemas de entretenimiento.** Sistemas de uso personal con el fin de entretener al usuario. La calidad de interacción ofrecida al usuario es la característica más importante de los sistemas de entretenimiento.
- **Sistema para modelado y simulación.** Sistemas que desarrollan científicos e ingenieros para modelar procesos o situaciones físicas, que incluyen muchos objetos separados interactuantes. Suelen ser computacionales intensivos y para su ejecución requieren sistemas paralelos para su alto desempeño.
- **Sistemas de adquisición de datos.** Son sistemas que desde su entorno recopilan datos usando un conjunto de sensores y envían dichos datos para su procesamiento a otros sistemas. El SW tiene que interactuar con los sensores y se suele instalar en un ambiente hostil o en un lugar remoto.
- **Sistemas de sistemas.** Sistemas compuestos de un cierto número de sistemas SW. Algunos son producto de SW genérico como un programa de hoja de cálculo. Otros sistemas en el ensamble pueden estar especialmente escritos para ese entorno.

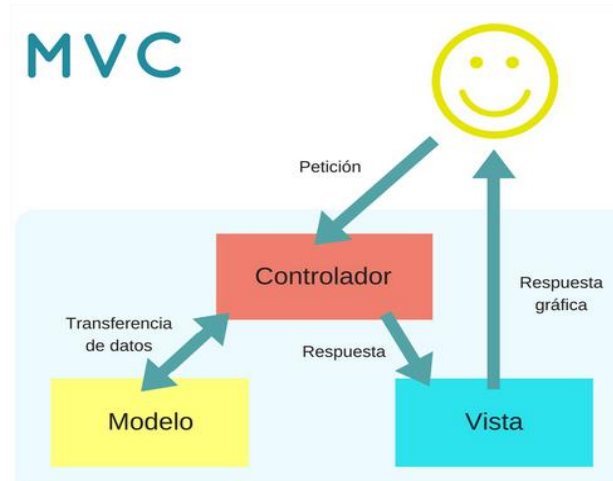
T1. 2.2 Diversidad de la Ingeniería del Software

¿Cuáles son los atributos del buen software?	El buen software debe entregar al usuario la funcionalidad y el desempeño requeridos; y debe ser mantenible, eficiente, confiable y aceptable.
--	--

- Para cada tipo de sistema se usan distintas técnicas de ingeniería del software porque el software tiene características muy diferentes.
- Existen **fundamentos de ingeniería del software que se aplican a todos los tipos de sistema de software**:
 - Deben llevarse a cabo usando un **proceso de desarrollo administrado y comprendido**.
 - La organización que diseña el software necesita planear el proceso de desarrollo, como tener ideas claras sobre lo que producirá y el tiempo en que estará completado.
 - La **confiabilidad y el desempeño** son importantes para todos los tipos de sistemas.
 - El SW tiene que comportarse como se espera sin errores y, cuando se requiera, estar disponible.
 - Es importante comprender y gestionar la **especificación y los requerimientos del software** (lo que el software debe hacer).
 - Debe conocerse qué esperan de él los diferentes clientes y usuarios del sistema y gestionar sus expectativas para entregar un sistema útil en fecha y presupuesto.
 - Tiene que **usar de manera tan efectiva como sea posible los recursos existentes**.
 - Esto significa que, donde sea adecuado, hay que reutilizar el SW que se haya desarrollado, en vez de diseñar uno nuevo.
- Estos fundamentos no cubren la implementación ni la programación. La programación varía totalmente de un sistema a otro.

T1. 2.3. Ingeniería del software y la Web

- Desde el año 2000 la Web comenzó a evolucionar, y a los navegadores se les agregaron cada vez más funcionalidades. Los sistemas basados en web pueden desarrollarse donde se tiene acceso a dichos sistemas, **usando un navegador Web, en lugar de una interfaz de usuario de propósito específico.**
- Así se produjo el **nacimiento** de una gran variedad de nuevos productos de sistemas a los cuales se ingresaba desde Web. Evolución de negocios y software organizacional.
- En lugar de implantar software en PCs se hacía en un Servidor Web, era menos costoso porque el diseño de interfaces resultaba bastante caro.
- **La siguiente etapa en el desarrollo de los sistemas basados en la Web fue la noción de los servicios Web.**
 - Componentes de SW que entregan funcionalidad específica y útil, a los que se accede desde la Web.



T1. 2.3. Ingeniería del software y la Web

- En los últimos años se desarrolló la noción de **“software como servicio”**. Se propuso que el software no correría usualmente en ordenadores locales sino como **“servicios en la nube”** a los que se accede a través de Internet. **SaaS**
 - Los usuarios no compran SW sino que **pagan según el tiempo de software que se utiliza**, o también se les otorga acceso gratuito a cambio de ver anuncios publicitarios que se despliegan en sus pantallas.
 - ¿De quién son los datos? **Serie interesante: “The Great Hack”**
 - La **reutilización** de software se ha convertido en el enfoque dominante para construir sistemas basados en la Web.
 - **Especificación progresiva**. Ahora se conoce en general que no es práctico especificar por adelantado todos los requerimientos para tales sistemas. Los sistemas basados en la Web deben desarrollarse y entregarse de manera progresiva.
 - **Interfaces restringidos**. Las interfaces de usuario están restringidas por las capacidades de los navegadores Web.
- Las ideas fundamentales de la ingeniería de software se aplican en el software basado en la Web de la misma forma que en otros tipos de sistemas de software.

T1. 3. Ética en la Ingeniería del Software

- Como otras disciplinas de ingeniería, la ingeniería del software se realiza dentro de un **marco social y legal** que limita el comportamiento de las personas que trabajan en dicho área.
- Un ingeniero debe mantener **estándares normales de honestidad e integridad**. No debe usar sus habilidades y experiencia para comportarse de forma deshonesta o de un modo que desacredite la profesión de la ingeniería del software.
- Existen **áreas donde los estándares de comportamiento aceptable no están acotados** por la **legislación** sino por **la responsabilidad profesional**
 - **Confidencialidad**. Por lo general debe respetar la confidencialidad de sus empleados o clientes sin importar si se firmó o no un acuerdo formal sobre la misma.
 - **Competencia**. No debe desvirtuar su nivel de competencia. Es decir, no hay que aceptar de manera intencional trabajo que esté fuera de su competencia.
 - **Derechos de propiedad intelectual**. Tiene que conocer las leyes locales que rigen el uso de la propiedad intelectual, como las patentes y el copyright, Cuidadoso con propiedad intelectual de empleadores y clientes
 - **Mal uso de ordenadores**. No debe emplear sus habilidades técnicas para usar incorrectamente los ordenadores de otros individuos. El mal uso de ordenadores varía desde lo trivial hasta lo extremadamente serio.