



---

**Universidad de las Fuerzas Armadas ESPE**

**Departamento: Ciencias de la computacion**

**Carrera : Ingeniria en Tecnologias de la Informacion**

**Taller académico N°: 2**

---

**1. Información General**

- **Asignatura: Metodologia de Desarrollo de Software**
  - **Apellidos y nombres de los estudiantes: Peñaherrera Rojas Jhaldry Santiago, Sandoval Espinoza José María y Pinto Segovia Diego Xavier**
  - **NRC: 20967**
  - **Fecha de realización: 04/05/2025**
- 

**2. Objetivo del Taller y Desarrollo**

**Objetivo del Taller:**

Analizar y responder de manera clara y fundamentada una serie de preguntas relacionadas con los principios, metodologías, procesos y buenas prácticas de la Ingeniería de Software, con el fin de reforzar la comprensión teórica.



**Desarrollo:**

Responder las siguientes preguntas

1.1. Explique por qué el software profesional no sólo son programas que se desarrollan para un cliente

El software profesional se caracteriza no solo por cumplir una función específica, sino por el enfoque integral con el que es concebido, diseñado y desarrollado. A diferencia de los programas creados de forma improvisada o con fines meramente personales, el software profesional se distingue por la aplicación rigurosa de metodologías de desarrollo, el cumplimiento de estándares de calidad, la documentación adecuada, y su orientación hacia la solución de problemas reales dentro de un entorno determinado (Anónimo, illusionstudio, 2023).

1.2. ¿Cuál es la principal diferencia entre desarrollo de productos de software genéricos y desarrollo de software personalizado? ¿Qué significa esto en la práctica para los usuarios de productos de software genérico?

La principal diferencia entre el desarrollo de software genérico y software personalizado radica en quién define los requisitos y para quién se construye el software:

El software genérico se desarrolla para ser vendido a muchos usuarios. Se produce una organización en base a lo que se cree que será útil para un amplio mercado, sin conocer las necesidades de cada cliente individual. Ejemplos: Microsoft Word o navegadores web.



El software personalizado, por otro lado, desarrolla por encargo específico de un cliente, el cual define los requisitos y características que necesita. Se ajusta a técnicas, reglas o necesidades únicas de esa organización.

1.3. ¿Cuáles son los cuatro atributos importantes que debe tener todo software profesional?

Sugiera otros cuatro atributos que en ocasiones sean significativos.

Uno de los atributos más importantes que debe tener un software profesional es que este sea *funcional*, debe hacer lo que se le pide, cumpliendo correctamente con los requisitos planteados. Así mismo este debe ser *fiable*, funcionar de manera estable, sin fallos inesperados. Como tercer atributo se tiene la *eficiencia*, es decir hacer un uso adecuado de los recursos del sistema, y finalmente la *mantenibilidad*, este termino se refiere a que el producto debe ser fácil de modificar, adaptable a cambios repentinos sin afectar su funcionamiento (Guerrero, 2023).

Como una sugerencia de atributos igualmente importantes se tiene la Escalabilidad, seguridad, intuitivo y finalmente la portabilidad.

1.4. Además de los retos de la heterogeneidad, cambio empresarial y social, y confianza y seguridad, identifique otros problemas y retos que sea probable que enfrente la ingeniería de software en el siglo XXI. (Sugerencia: piense en el ambiente).

Se han considerado algunas problemáticas o retos con los que se enfrentará la ingeniería de Software. El primero se enfoca en la sostenibilidad y eficiencia energética, el software debe ser diseñado para consumir menos recursos, especialmente en centros de datos y dispositivos móviles. Otro reto que se podría considerar se relaciona con la



inteligencia artificial (IA) y su uso ético, el desarrollo de software basado en IA plantea dilemas éticos, como la transparencia en algoritmos, sesgos en datos y decisiones automatizadas. Y finalmente un reto de gran relevancia en el mundo tiene que ver con la privacidad de los datos personales, especialmente con la recopilación, almacenamiento y distribución de datos personales o información sensible (Summerville, 2011).

1.5. Con base en su conocimiento de algunos tipos de aplicación estudiados en la sección 1.1.2, explique, con ejemplos, por qué diferentes tipos de aplicación requieren técnicas especializadas de ingeniería de software, para apoyar su diseño y desarrollo.

Los sistemas de entretenimiento, como los videojuegos, suelen poseer un ciclo de vida útil más larga que algunos sistemas, es por eso que no podríamos comparar sus requerimientos con un sistema de control embebido, ya que estos últimos requieren más tiempo de desarrollo y validación ya que tienen un hardware integrado, como por ejemplo los frenos de un automóvil y el programa que los activa.

Además, podríamos hablar de los sistemas de transacción donde se necesita confiabilidad y seguridad como por ejemplo las transacciones de la Banca Móvil de Pichincha es por eso que las técnicas de desarrollo de ingeniería de software son necesarias en aplicaciones que intervengan con dinero, o seguridad. Mientras que los sistemas de entretenimiento como los videojuegos al final solo necesitan un diseño funcional y priorizan la experiencia del usuario, gráficos avanzados y respuesta rápida. Es por eso que no todos los sistemas van a requerir el mismo, es decir se desarrollaran con distintas técnicas de Ingeniería de Software (Summerville, 2011).



- 1.6. Explique por qué existen ideas fundamentales de la ingeniería de software que se aplican a todos los tipos de sistemas de software.

La principal razón por la que existen ideas fundamentales en la ingeniería de software que se aplican a todos los tipos de sistemas es porque todos los proyectos comparten principios básicos o metodologías sin importar el dominio, la tecnología o el tipo de aplicación. Estas ideas garantizan que el software sea confiable, mantenible y funcional. Algunas de estas ideas son: Verificación y validación, mantenimiento continuo y el trabajo en equipo (UNIR, 2024).

- 1.7. Explique cómo el uso universal de la Web cambió los sistemas de software. El software ya no necesitaba ser instalado ya que se podía visualizar en un navegador, empezó a utilizarse el software como servicio Los sistemas web ahora se desarrollan de manera progresiva, es decir no hay una versión final instalada en una computadora.

El avance hacia el uso universal de la Web permitió el ahorro de espacios en los computadores ya que el uso de interfaces para usuario era muy caro y pesado. La web daba acceso a paginas donde se podrían simular acciones antes realizadas por aplicaciones. Es así que gracias a estas facilidades brindadas por la Web se generaron aún más aplicaciones para satisfacer las necesidades de la comunidad que usaba la Web (Summerville, 2011).



1.8. Analice el hecho de si los ingenieros profesionales deben ser certificados en la misma forma que los médicos o abogados.

Para poder responder a esta pregunta es necesario hacer un análisis a las diferentes posturas que presentan varios expertos, ya que muchos plantean que sí, los ingenieros de software deberían estar certificados profesionalmente, al igual que los médicos o abogados, especialmente porque su trabajo puede tener un gran impacto en la seguridad y la privacidad. Por ejemplo, un error médico puede costar vidas, un fallo en software crítico (como en aviones, hospitales o bancos) puede tener consecuencias graves.

Por otro lado, un gran grupo plantean que no se los debería considerar de igual forma, ya que estas exigencias podrían frenar la innovación, especialmente en startups o desarrolladores independientes. También el campo del software es muy amplio, lo que hace difícil crear una única certificación válida para todos los contextos.

Una alternativa podría ser exigir la certificación únicamente a los profesionales que operan sistemas de alta importancia, como los relacionados con la salud, la defensa o la infraestructura esencial. Para otros sectores, se podrían impulsar certificaciones voluntarias o enfocadas en áreas específicas, incentivando la calidad sin que la certificación sea un requisito obligatorio para ejercer la profesión (Anónimo, Quora, 2024).



# ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS  
INNOVACIÓN PARA LA EXCELENCIA



1.9. Para cada una de las cláusulas del Código de ética ACM/IEEE que se muestra en la figura 1.3, sugiera un ejemplo adecuado que ilustre dicha cláusula.

## Público

Un ingeniero detecta que una aplicación médica tiene un error que podría provocar diagnósticos incorrectos. Aunque la empresa decide no corregirlo de inmediato, el ingeniero informa a las autoridades pertinentes para proteger a los pacientes.

## Cliente y empleador

Un desarrollador se niega a modificar un sistema de control de nóminas para que oculte pagos irregulares solicitados por su empleador, porque afectaría la transparencia hacia los trabajadores.

## Producto

Antes de liberar una nueva versión de una app bancaria, un equipo exige más pruebas de seguridad, aunque esto retrase el lanzamiento, priorizando la calidad y confianza del producto.

## Juicio

Un ingeniero recomienda no usar un componente de software de código abierto que no cumple con los requisitos legales, a pesar de que otros miembros del equipo lo prefieren por facilidad.



#### Gestión:

Un gerente de proyecto establece políticas internas para registrar con transparencia todos los errores encontrados, sin permitir que se oculten para cumplir plazos.

#### Profesión:

Un profesional participa en eventos académicos para compartir buenas prácticas de desarrollo seguro y responsable, reforzando la imagen positiva de la ingeniería de software.

#### Colegas:

Una ingeniera orienta a un colega nuevo ayudándole a comprender los estándares de codificación y ética del equipo, promoviendo colaboración y respeto.

#### Uno mismo:

Un profesional asiste a cursos sobre ciberseguridad y nuevas regulaciones para asegurar que su trabajo esté alineado con las exigencias actuales y los principios éticos.





1.10. Para ayudar a contrarrestar el terrorismo, muchos países planean o desarrollaron sistemas de cómputo que siguen la pista a gran cantidad de sus ciudadanos y sus acciones. Claramente esto tiene implicaciones en cuanto a la privacidad. Discuta la ética de trabajar en el desarrollo de este tipo de sistema.

El desarrollo de sistemas de vigilancia que rastrean a gran escala a los ciudadanos plantea un dilema ético profundo: por un lado, pueden ayudar a prevenir actos de terrorismo; por otro, representan una amenaza significativa a la privacidad y las libertades civiles.

La postura a favor de estos sistemas sostiene que su aplicación sería con un fin enfocado a la protección, ya que el sistema ayudaría a prevenir ataques, lo que salvaría más vidas, siendo esto una responsabilidad ética importante.

Por otro lado, la protura en contra menciona que los sistemas de vigilancia estarían violando la privacidad al momento de monitorear sin consentimiento. Esto aparte generaliza la sospecha sobre toda la población, incluso inocentes.

Ahora bien, cuáles son las implicaciones éticas que engloban al ingeniero en software, según los códigos de ética, el ingeniero debe actuar en favor del interés público y con respeto por los derechos humanos.

Es por esto que trabajar en este tipo de sistemas no es éticamente incorrecto por sí solo, pero depende de cómo y para qué se usen. Los ingenieros deben ejercer juicio crítico, y negarse a colaborar si el sistema viola derechos fundamentales (Summerville, 2011).



---

### 3. Referencias (Norma APA 7.0)

- Anónimo. (08 de Septiembre de 2023). *illusionstudio*. Obtenido de <https://www.illusionstudio.es/importancia-uso-de-software-empresa#:~:text=El%20software%20cuenta%20con%20caracter%C3%ADsticas,decisiones%20instant%C3%A1neas%20para%20cambiar%20estrategias>.
- Anónimo. (12 de Septiembre de 2024). *Quora*. Obtenido de <https://www.quora.com/Should-a-professional-engineer-be-certificated-the-same-as-doctors-or-lawyer>
- Guerrero, P. (5 de Abril de 2023). *pragma*. Obtenido de <https://www.pragma.co/es/blog/como-ser-arquitecto-de-software-aprende-sobre-atributos-de-calidad>
- Summerville, I. (2011). *Ingeniería de Software*. Mexico: PEARSON EDUCACIÓN.
- UNIR. (10 de Mayo de 2024). *UNIR*. Obtenido de <https://colombia.unir.net/actualidad-unir/ingenieria-de-software-que-es-objetivos/>
-