



Universidad de las Fuerzas Armadas ESPE

Departamento: Ciencias de la computación

Carrera : Ingeniería en Tecnologías de la Información

Deber académico N°: 1

1. Información General

- **Asignatura: Metodología de Desarrollo de Software**
 - **Apellidos y nombres de los estudiantes: Gallardo Vega Santiago José, Pérez Díaz David Ismael, Zambrano Cajas Isabela Valentina**
 - **NRC: 20967**
 - **Fecha de realización: 30/04/2025**
-

2. Objetivo del Taller y Desarrollo

Objetivo del Taller:

[Escribir aquí el objetivo principal del taller, por ejemplo: "Aplicar conceptos fundamentales del desarrollo ágil de software mediante la simulación de una planificación de sprint utilizando Scrum."]

Desarrollo:

[Describir aquí las actividades realizadas en el taller. Puedes incluir secciones como: Introducción teórica breve, Actividades prácticas, Discusión de resultados, Conclusiones.]

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

El desarrollo de software no solo se basa en crear un producto para vender, también se basa en el levantamiento de requisitos, testeo, depuración, despliegue y mantenimiento del software, por tanto el trabajo del desarrollador no termina cuando el producto se entrega, este también debe mantenerlo y documentarlo correctamente.

1.3. ¿Cuáles son los cuatro atributos importantes que debe tener todo software profesional? Sugiera otros cuatro atributos que en ocasiones sean significativos.

- **Funcional:** El programa debe cumplir una función o tarea.



- Acorde a requisitos del cliente: El software profesional debe tener todas las funciones y requisitos especificados por el cliente.
- Sustentable: El software debe ser expandible, mantenible y versátil al cambio, en el caso de que el cliente requiera expandir el enfoque del software.
- Confiable: El cliente debe poder confiar en el programa y en lo que hace para su trabajo.

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 a todos los tipos de sistemas de software.

En las aplicaciones independientes se corre la aplicación en una computadora local, esta no necesita conectarse a una red. Un ejemplo de esto es Photoshop, un software de edición fotográfica que se enfoca en el rendimiento gráfico y herramientas para la manipulación de imágenes, está para un correcto funcionamiento requiere de una interfaz optimizada y algoritmos para el proceso de elaboración gráfica, la técnica usada aquí es el desarrollo de interfaces intuitivas, rendimiento en sistemas locales y el manejo de memoria.

En contraste a las aplicaciones interactivas basadas en transacciones las cuales se enfocan en permitir transferencias seguras y rápidas, asegurando disponibilidad del servicio. Estas se enfocan más en seguridad robusta, manejo de datos, arquitectura distribuida y estrategias de escalabilidad.

Estas diferencias destacan que cada aplicación requiere técnicas especializadas de ingeniería de software porque sus características y necesidades se enfocan en distintas demandas o requisitos.

1.7. Explique cómo el uso universal de la Web cambió los sistemas de software.

El desarrollo de la World Wide Web (Web) en cuanto al uso universal tuvo un efecto radical en los sistemas de software haciendo que el mismo tenga que evolucionar en cuanto a su desarrollo y operaciones, el primer cambio es la accesibilidad y la distribución global antes de la web el software se ejecutaba en computadoras locales, con la expansión de la Web así los sistemas se volvieron accesibles desde cualquier lugar del mundo rompiendo la barrera geográfica.

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.

1. Público:

Si el ingeniero detecta un fallo en la seguridad de su aplicación desarrollada, aunque no afecta directamente el uso de la aplicación, se debe pensar en la protección de los datos del usuario.

2. Cliente:



El ingeniero propone una mejor opción que cumple de manera eficiente todos los requisitos del cliente a pesar de que esto signifique una menor ganancia.

3. Producto:

Enviar constantes actualizaciones de software o seguridad, para garantizar el correcto funcionamiento del producto sin importar el tiempo que pase.

4. Juicio:

Cuando el ingeniero se niega a utilizar programas no seguros o poco amigables para el cliente aunque esto incremente el costo de desarrollo del programa.

5. Gestión:

El jefe o líder de grupo propone una buena organización y administración de los recursos dentro de la toma de decisiones para el desarrollo de software.

6. Profesión:

7. Colegas:

Una empresa en la que existe un buen ambiente laboral donde todos se apoyan entre si y comparten recursos para mejorar sus conocimientos y habilidades.

8. Uno Mismo:

El ingeniero que se mantiene actualizado frente a las nuevas amenazas

3. Referencias (Norma APA 7.0)

- Sommerville, I. (2015). *Software Engineering* (9th ed.). Pearson Education.
-