1. **Responda las preguntas a final del capítulo: (1-12), incluyendo las que se refieren a temas más amplios como la 9.4, donde debe diseñar una BD que le permita administrar los ECS.**

**9.1 ¿Por qué es cierta la primera ley de la ingeniería de sistemas? ¿Cómo afecta a nuestra percepción de los paradigmas de la ingeniería del software?**

Es totalmente cierta porque siempre existirán nuevos requerimientos o ,como se da muchas veces, los usuarios no tienen completamente claro que es lo que quieren o que necesitan realmente y a medida que lo van notando piden mas cosas.

Nosotros solemos creer que luego de que se hace entrega formal del software ya no hay que trabajar mas en el, que esta completamente cerrado, y es falso. Un software nunca estará terminado en su totalidad, como dije antes, siempre habrán mejoras, correcciones que se tendremos que hacer para satisfacer las necesidades del cliente.

**9.2 Exponga las razones de la existencia de líneas base con sus propias palabras.**

Para evitar realizar cambios que puedan dañar un proceso que ya fue revisado y aprobado de modo que todo cambio que se vaya a realizar deberá pasar por un proceso se revisión, pruebas y correcciones hasta estar en perfecto estado.

**9.3 Asuma que es el gestor de un pequeño proyecto. ¿Qué líneas base definiría para el proyecto y cómo las controlaría?**

**9.4 Diseñe un sistema de base de datos que permita a un ingeniero del software guardar, obtener referencias de forma cruzada, buscar, actualizar, cambiar, etc., todos los elementos de la configuración del software importantes.**

**- ¿Cómo manejaría la base de datos de diferentes versiones de un mismo programa?**

**- ¿Se manejaría de forma diferente el código fuente que la documentación?**

**- ¿Cómo se evitaría que dos programadores hicieran cambios diferentes sobre el mismo ECS al mismo tiempo?**

**9.5 Investigue un poco sobre bases de datos orientadas a objetos y escriba un artículo que describa cómo se podrían usar en el contexto de la GCS.**

Las bases de datos orientadas a objetos permite representar elementos vistos en la realidad y guardar configuraciones de un programa.

**9.6 Utilice un modelo E-R (Capítulo 12) para describir las interrelaciones entre los ECS (objetos) de la Sección 9.1.2.**

**9.7 Investigue sobre herramientas de GCS existentes y describa cómo implementan el control de versiones, de cambios de objetos de configuración de software**

Una herramienta muy utiliza es Git, la misma posee una interfaz gráfica fácil de usar. Ademas de que tiene varias características como:

-Cada versión registrada en el tiempo es una revisión.

-Cada operación se realiza en el repositorio local.

-Permite enviar cambios a repositorios remotos.

-Es multiplataforma.

-Permite crear varias ramas para evitar hacer cambios directos en la rama principal hasta estar totalmente probados.

**9.8 Las relaciones ≪parte-de≫ e ≪interrelacionado≫ representan relaciones sencillas entre los objetos de configuración. Describa cinco relaciones adicionales que pudieran ser útiles en el contexto de la base de datos del proyecto.**

- contenido-por

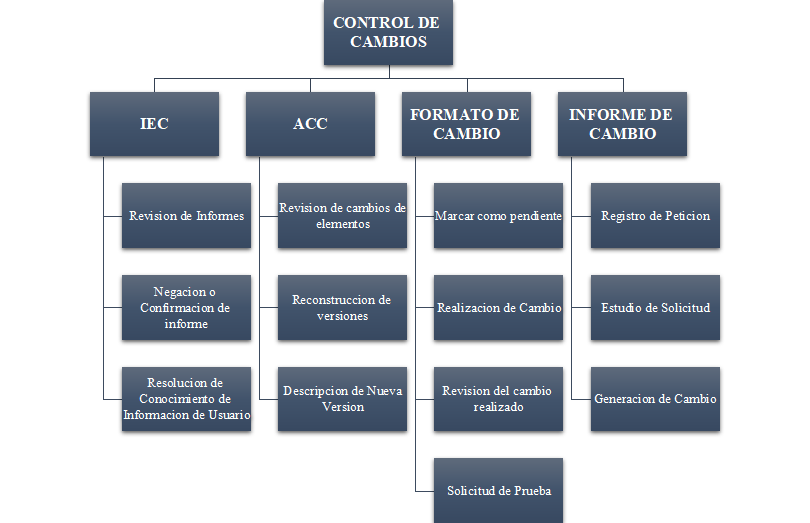
-hasta-de

-desde-de

-depende-de

-conjunto-a

**9.10 Utilizando la Figura 9.5 como guía, desarrolle un esquema de trabajo más detallado aún para el control de cambios. Describa el papel de la ACC y sugiera formatos para la petición de cambio, el informe de cambios e IEC.**

****

**9.11 Desarrolle una lista de comprobaciones que se pueda utilizar en las auditorías de configuración.**

**9.12 ¿Cuál es la diferencia entre una auditoría de GCS y una revisión técnica formal? ¿Se pueden juntar sus funciones en una sola revisión? ¿Cuáles son los pros y los contras?**