**Sitúa las tres capas de ingeniería del software encima de la capa titulada ≪enfoque de calidad≫.**

**≪Proceso ≫**

**≪enfoque de calidad≫**

**≪Metodos ≫**

**≪Herramientas≫**

**Esto implica un programa de calidad tal como Gestión de Calidad Total. Investigue y desarrolle un esquema de los principios clave de un programa de Gestión de Calidad Total**

Imagen que contiene Texto

Descripción generada automáticamente

(1)Imagen

(ingenieria)

2. Hay algún caso en que no se apliquen fases genéricas del proceso de ingeniería del software?

Si hay varios caso en los que no se aplican las fases genéricas uno de ellos son :

(Aguilar)

Descríbalo

1. Desarrollo ágil: El enfoque ágil de desarrollo de software se basa en principios como la colaboración cercana con los clientes, la entrega incremental y la respuesta rápida a los cambios. En este enfoque, las fases tradicionales del proceso de ingeniería del software pueden fusionarse o eliminarse por completo. Por ejemplo, en lugar de tener una fase separada de diseño, implementación y prueba, se pueden realizar estas actividades de manera iterativa y continua a lo largo del proyecto.

2. Desarrollo rápido de aplicaciones (RAD): El RAD es un enfoque que se centra en la entrega rápida de un producto funcional mediante el uso de prototipos y ciclos de desarrollo acelerados, la fases tradicionales del proceso de ingeniería del software pueden simplificarse o acelerarse para adaptarse a los plazos ajustados.

3. Mantenimiento de software: En el caso del mantenimiento de software, las fases iniciales del proceso de ingeniería del software, como la planificación y el análisis de requisitos, pueden no ser necesarias. En su lugar, el enfoque se centra en la corrección de errores, la mejora de la funcionalidad existente y la adaptación a los cambios en el entorno operativo.

4. Desarrollo de software de investigación: En el ámbito de la investigación científica, el desarrollo de software puede requerir un enfoque más flexible y experimental. En este caso, las fases genéricas del proceso de ingeniería del software pueden adaptarse o modificarse para permitir la experimentación y la iteración continua.

5. Proyectos pequeños o individuales: En proyectos pequeños o individuales, puede no ser necesario seguir todas las fases genéricas del proceso de ingeniería del software debido a limitaciones de tiempo o recursos. En estos casos, es posible que se omitan algunas fases o que se realicen de manera más informal y simplificada.

Pagina 24, Pressman, R.S. (2010). Software Engineering: A Practitioner's Approach (7th ed.). McGraw-Hill Education.

<https://books.google.com.mx/books?id=bL7QZHtWvaUC&lpg=PR26&ots=O8Ce8SzP6k&dq=Pressman%2C%20R.S.%20(2014).%20Software%20Engineering%3A%20A%20Practitioner's%20Approach%20(8th%20ed.).%20pdf&lr&hl=es&pg=PA11#v=onepage&q&f=false>

El modelo del caos sugiere que un bucle de resolución de problemas se pueda aplicar en cualquier grado de resolución, investigue y de un ejemplo.

4. Que paradigmas de ingeniería del software de los presentados en esta clase piensa que sería el más eficaz. ¿Por qué?

5. Proporcione cinco ejemplos de proyectos de desarrollo del software que sean adecuados para construir prototipos Nombre dos o tres aplicaciones que fueran más difíciles para construir prototipos.

6. El modelo DRA a menudo se une a herramientas CASE. Investigue la literatura y proporcione un resumen de una herramienta típica CASE que soporte DRA.

7. Proponga un proyecto específico de software que sea adecuado para el modelo incremental. Presente un escenario para aplicar el modelo al software.

8. ¿A medida que vaya hacia afuera por el modelo en espiral, que puede decir del software que se está desarrollando o manteniendo?

9. Muchas personas creen que la única forma en la que se van a lograr mejoras de magnitud en la calidad del software y en su productividad es a través del desarrollo basado en componentes. Encuentre tres o cuatro artículos recientes sobre el asunto y resúmalos.

10. Describa el modelo de desarrollo concurrente con sus propias palabras.

11. ¿Qué es más importante, el producto o el proceso