

TEAM#5 : Code Noob 8/10

Topic: DEAD CODE

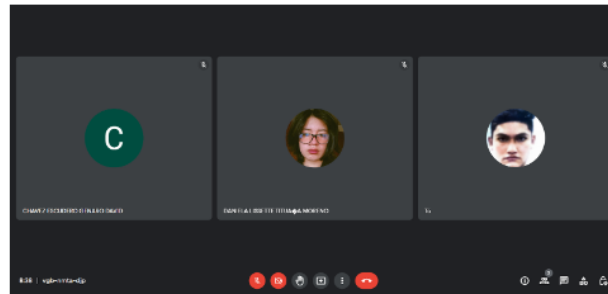
Chavez Escudero Genaro David 5

Imbaquinga Guña Jose Ricardo 5

Ponce Arguello Diego Armando 5

Tituaña Moreno Daniela Lissette 5

Meet:



Descripción del Problema: Código Muerto en programación significa, que una parte de código es ejecutable pero no se vuelve a reutilizar en ningún otro cálculo, lo cual también es un desperdicio de tiempo y memoria.

Consecuencias:

- Desperdicio de memoria
- Desperdicio de tiempo de ejecución
- Agrega complejidad al entendimiento del código.
- El Código muerto es engañoso si se lo comenta.
- Alguien puede volver a llamar al código muerto lo cual puede generar errores desastrosos.

Causas:

- En la mayoría de los casos donde existe Dead Code es debido al temor a perder código o malograr el sistema construido.
- Esto tiene relación directa con el desconocimiento de que los gestores de versiones permiten volver al pasado si es que se necesita ver algo que ha sido borrado.
- Otras causa muy común es el envejecimiento del código que debido a características obsoletas es una razón común a producir un Dead Code.

Cómo evitarlo:

- Inspecciones visuales
- Uso de un buen IDE
- Inspeccionar el código que no se usa durante meses o años.
- Si es un código heredado ejemplo de otra empresa, estar pendiente a comentarios de los anteriores desarrolladores

Como reconocerlo:

a comentarios de los anteriores desarrolladores

Como reconocerlo:

Instalar el sistema en ejecución para mantener registros del uso del código

Resultados jamás se miran o se utilizan

El código muerto hace que la base de código sea sustancialmente más difícil de mantener a escala

Código:

```

13
14 public static void main(String[] args) {
15     int addend1=10;
16     int addend2=20;
17     Variable restart1 is never read
18     (Alt-Enter shows hints)
19
20     int restart1=12;
21     int restart2=20;
22     int sumResult;
23     int valor1=10;
24     int result;

```

Última modificación de JORGE EDISON LASCANO el 30 de junio

```

13
14 public static void main(String[] args) {
15     int addend1=10;
16     int addend2=20;
17     Variable restart1 is never read
18     (Alt-Enter shows hints)
19
20     int restart1=12;
21     int restart2=20;
22     int sumResult;
23     int valor1=10;
24     int result;
25
26     sumResult=addend1+addend2;
27     result=valor1*sumResult;
28
29 }

```

Código arreglado

```

public class WS21Pitfalls {

    public static void main(String[] args) {
        int addend1=10;
        int addend2=20;
        int sumResult;
        int valor1=10;
        int result;
        sumResult=addend1+addend2;
        result=valor1*sumResult;
        System.out.println("The result is"+result);
    }
}

```