

## **Reporte 6.**

### **Planeación.**

Durante la planeación aprendí a seccionar por pasos lo que tenía que hacer durante un programa, a planear que es lo que tenía y que me podría ser útil durante la realización de un programa. En mis programas mantuve un porcentaje de reutilización de más de 20%.

### **Estimación.**

Mejoré mi capacidad de estimar el tiempo y el tamaño de mis programas considerablemente, además aprendí a que si estimo estos dos parámetros mis programas se vuelven mas predecibles y tiendo a cometer menor errores. En la estimación de tiempo del primer programa tuve un error de estimación del -9.72% que no es mucho, pero el programa 2 y 3 mi error en la estimación aumento, después del tercer programa la estimación volvió a bajar mas consistentemente hasta el programa 6 que tuve un error en la estimación del tiempo de 8.81%. Para la estimación de tamaño no pude alcanzar un porcentaje de error menor al 10% por lo que aun me falta practicar, pero si mejoré estimando ya que el error en la estimación fue bajando del programa 2 hasta el 4 donde obtuve el porcentaje de error más chico (-15.9%).

### **Calidad de software.**

Cuando se sigue un proceso para realizar software se tiende a tener un producto de mejor calidad ya que para todas las partes del proceso le dedicas un tiempo determinado con el cual puedes analizar mejor cada parte del producto de software que se quiere realizar.

### **Revisiones.**

Las revisiones son muy útiles para revisar y comprobar que todo lo que hiciste durante una fase sea correcto de acuerdo a checklist que contiene todos los puntos a revisar los cuales representan los problemas comunes de la persona que está realizando un programa. En el programa 6 en las fases de revisión de diseño y código se resolvieron el 100% de los defectos.

### **Diseño.**

El diseño me ayudo a mejorar los programas que realizaba, así como reducir en gran medida el tiempo y el tamaño de la codificación de mi programa, además de evitarme resolver una gran cantidad de errores que podría haber tenido si solo codificaba y no diseñaba. Para mi programa 6 el tiempo de diseño (32%) de mi programa es mucho mayor que el tiempo de codificación (13%) y el tiempo de testing (14%).

### **Métricas.**

Las métricas son importantes para saber cuantitativamente que tan bien o mal se esta realizando un proceso. Si las métricas salen fuera de lo esperado significa que hay cosas a mejorar con respecto a esa métrica, si los datos son congruentes con lo que se espera y con lo que se realiza, entonces el proceso se está siguiendo correctamente. Las métricas de tiempo por fase me ayudaron a darme cuenta en que fases me tomaba mas tiempo y con esto darles más tiempo a las fases anteriores y mejorar el resultado y el tiempo de la fase en cuestión. Los porcentajes de error en la estimación de tiempo y tamaño me ayudaron a darme cuenta que necesitaba planear mejor lo que me tardaría en realizar un programa.

### Medición del reporte.

	Hora de comienzo	Hora de finalizacion
<b>Planeacion</b>	10:20:00 a. m.	10:30:00 a. m.
<b>Analisis</b>	10:35:00 a. m.	11:00:00 a. m.
<b>Desarrollo</b>	11:05:00 a. m.	11:40:00 a. m.
<b>Postmortem</b>	11:43:00 a. m.	11:55:00 a. m.

<b>Total:</b>	1:22:00
<b>Estimado:</b>	1:30:00

	Porcentaje
<b>Planeacion</b>	15%
<b>Analisis</b>	35%
<b>Desarrollo</b>	35%
<b>Postmortem</b>	15%
<b>Total:</b>	100%