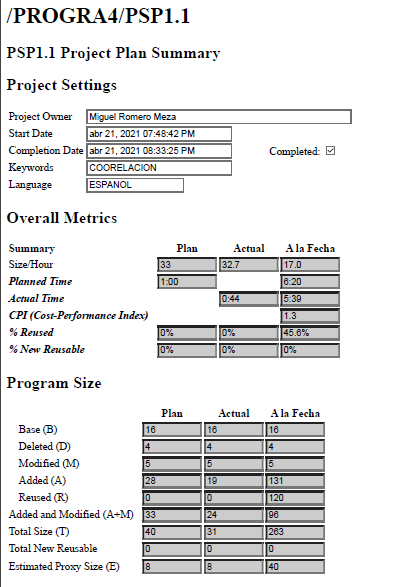
# REPORTE INTERMEDIO PSP

## Proceso de análisis de reportes e información del PSP DASHBOARD

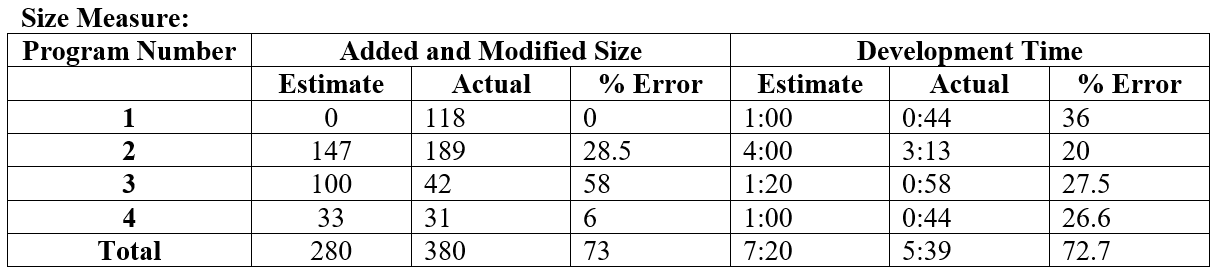
PSP Interim ReportProcess Script Template

|  |  |  |
| --- | --- | --- |
| Pase # | Purpose | Guía para el análisis y la redacción del informe provisional del PSP |
|  | Entry Criteria | * Finalización de los 4 primero programas y sus reportes * Tener los requerimientos a la mano * Tener los reportes del formulario plan summary y el registro de tiempo a la mano * Ejecutar el apartado del Interim Report en el dashboard |
|  | Planning | * Durante la planeación, habrá que reunir todos los reportes de los programas 1 al 4 * Plan summay * Estimación de tamaño * Reporte de tiempo * Reporte de defectos * Reporte r3 y r4 * Reporte PROBE * Así como las gráficas de:  |  | | --- | | • Análisis de Defectos  • Análisis de Planeación  • Análisis de Proceso  • Análisis de Calidad | |
|  | Development | Para el análisis de los datos se realizó lo siguiente:   * 1. 1. en la primera parte a través de una regla de 3 se estimo el porcentaje de error del calculo de tamaño y de tiempo. Ejemplo de formula:   𝑙𝑝∗ (100/𝑙𝑎)=%de error  De los cuales:  Lp: líneas planeas  La: líneas actuales  Una formula similar se uso para el calculo del tiempo  t𝑝∗ (100/t𝑎)=%de error  de los cuales:  tp: tiempo planeado  ta: tiempo actual  nota: para el primer programa, no existe un dato de planeación, además de que se hará un calculo para la sumatoria total de líneas planeadas y reales.  2. calculo de densidad de defecto, en esta parte lo que hacemos es obtener el porcentaje de error comparado con la extensión de tamaño de nuestro programa. La fórmula es la siguiente:  Numero de defectos\*(100/tamaño de programa)=%de error  Del mismo modo se deja un espacio para la el cálculo de la sumatoria d ellos datos anteriores.  3. calculo del porcentaje de defectos encontrados durante la compilación y el testeo. Esto a través de la fórmula:  Numero de defectos\*(100/tamaño de programa)=%de error  En mi caso, la detección y corrección de mis errores estuvo en el tiempo de compilación, por lo que la detección de errores en tiempo de testeo es 0.  4. en una tabla mas se separan los defectos encontrados separados por sus fases de manera de que se pueda encontrar en que parte de insertan mas errores durante el desarrollo del proyecto.  5. es una tablas mas se categorizan los errores de manera que se identifiquen sus tipos e identificar aquellos que sean recurrentes.  6. Dar respuesta a las preguntas:  • Para cada uno de los programas 3 y 4, ¿cómo se compara el tamaño aproximado estimado con el tamaño real añadido y modificado?  • Para cada uno de los programas 3 y 4, si el tamaño estimado del proxy no se acerca al tamaño real añadido y modificado, determine por qué.  • Basándose en el análisis anterior, ¿qué es lo que podría hacer para mejorar la precisión de sus estimaciones?  • Para los cuatro primeros programas, ¿cuál fue la estabilidad de su productividad?  • En los cuatro primeros programas, si su productividad varió, ¿están los cambios en la productividad relacionados con la cantidad de tiempo que pasó corrigiendo defectos en las fases de compilación y prueba?  • ¿Cómo se relaciona la precisión de sus estimaciones de tiempo con la precisión de sus estimaciones de tamaño?  • ¿Qué tipos de defectos inyectó con más frecuencia en la fase de diseño?  • ¿Qué tipos de defectos inyectó con más frecuencia en la fase de código?  • ¿Qué tipos de defectos eliminó principalmente en la fase de compilación?  • ¿Qué tipos de defectos eliminó principalmente en la fase de prueba?  • ¿Qué categoría de defectos inyectados tuvo el mayor tiempo medio de reparación?  • ¿Qué categoría de defectos inyectados tuvo el mayor tiempo total de reparación?  • ¿Encontró el compilador todos sus defectos de tipo 20 (sintaxis/tipología)? |
|  | Postmortem | * Redacción de conluciones tomando en cuenta: * Los tipos de defectos inyectados y eliminados por fase para los programas desarrollados hasta la fecha   • Los tipos de defectos encontrados por el compilador  • Los tipos de defectos encontrados durante las pruebas  • Los tipos de defectos inyectados y encontrados por fase  • Los defectos inyectados por hora por fase  • Los tiempos de reparación de los defectos en cada fase  • La densidad de defectos por unidad de tamaño por fase  • Tipos de defectos que se inyectaron en el diseño y los tipos que inyectó durante la codificación. |
|  | Exit Criteria | Adjuntar en un directorio los siguientes archivos:   * • Análisis del Reporte 4 del programa cuatro. * • Informe del Reporte intermedio. * • Obtener el Generic Project Planning Summary. * • Generar la Size Estimating Template. |

## Proceso de planeación:

Durante el proceso de planeación se recurre a los scripts descripción del proyecto, de manera que leerlo pueda ser legible y de una comprensión fácil de los requerimientos del programa, la mayoría de esos si bien recurre a operaciones matemáticas complejas también traen una descripción paso a paso de cómo ejecutar este problema, tanto un ejemplo como una descripción de los elementos que conforman el algoritmo o la fórmula matemática. Para poder entender bien este problema se recurre a realizar ejercicios de manera manual usando herramientas como papel y lápiz o Microsoft Excel, una vez llegado a este punto se comiencen a desarrollar las diferentes fases de descomposición del algoritmo de manera que se puedan extraer las partes más sencillas en algunos métodos.

Este proceso de planeación no solamente nos ha llevado utilizar este método sino también hacer un poco más metódico a la hora de estimar tanto tiempo como tamaño del proyecto.

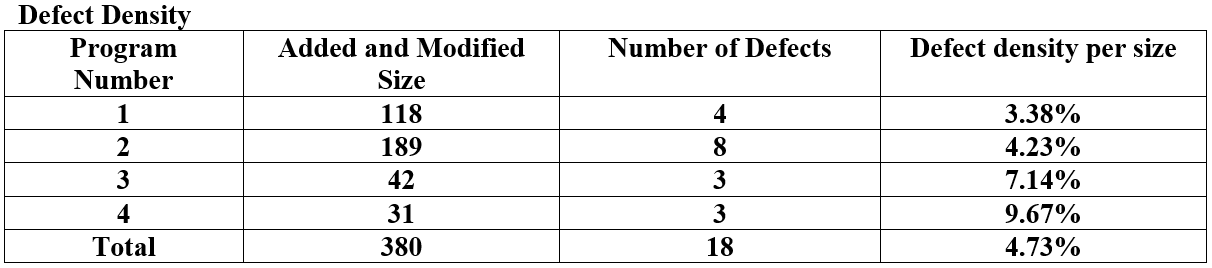


Si bien hay proyectos que hasta este momento se habían realizado por primera vez la estimación de dichos códigos o proyectos ha sido mayor de la Real No obstante cuando trata de elementos que son muy parecidos se ha podido observar que la estimación se acerca mucho a lo real en cuanto al tamaño. En la tabla anterior se puede observar la estimación de estos tamaños comparado con el tamaño actual y el porcentaje de error además del tiempo de desarrollo. Hablando de la estimación del tiempo la estimación siempre ha sido un poco debajo de lo estimado con un rango de error de un 20% promedio.

## 

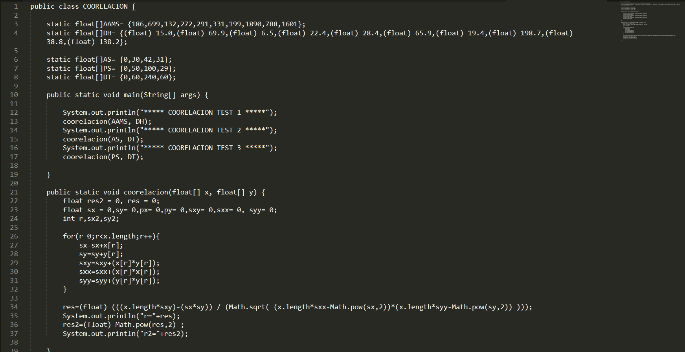
No obstante, la gráfica de defectos totales, muestra como los errores se incrementan mientras en promedio el numero de defectos es mayor con relación al tamaño. Es decir a pesar que el programa 3 y 4 recibieron el mismo numero de defectos, el programa 4 es mas pequeño, lo que hace que su porcentaje de error sea mas grande que el programa 3.

Como muestra de lo anterior se muestra la siguiente tabla:



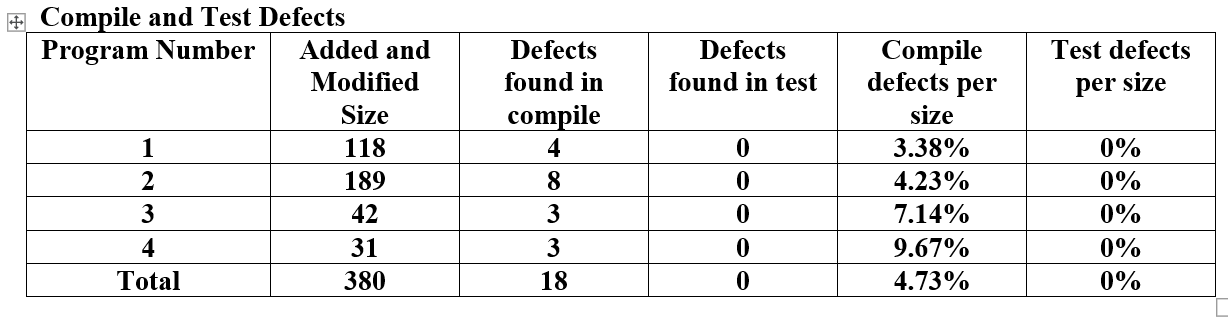
En la imagen anterior podemos observar en comparación con el tamaño el porcentaje de error y número de errores líneas de código con un promedio de error del 5.96%

## Proceso de desarrollo:

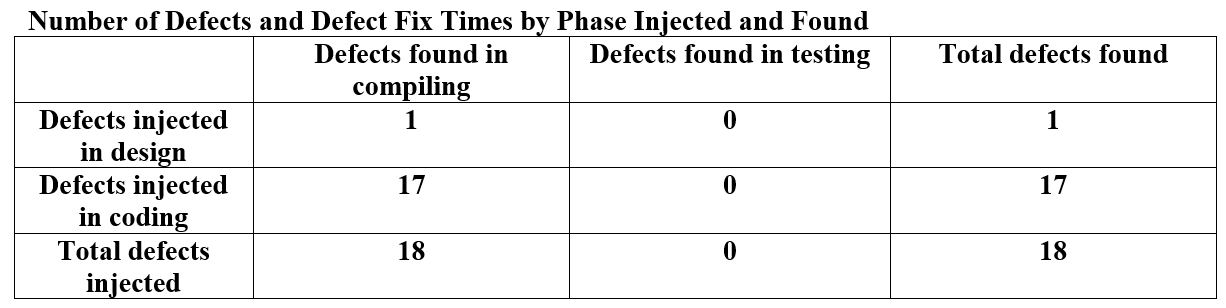
En cuanto a la fase de desarrollo el no tener una guía para poder visualizar algunos comandos el código del color de las palabras reservadas ha llegado si bien hace al principio un poco incómodo, también ayuda a poder tener un mejor entendimiento de algunas instrucciones, además de poder memorizar y tener mucho más cuidado a la hora del compilado.

Es en la fase de compilado donde yo me puedo dar cuenta que la mayoría de mis errores pueden estar en el periodo de desarrollo y que algunos de estos son muy recurrentes y se podría decir hasta básicos, también es el punto donde me he podido ver errores que se han ingresado en otros momentos del desarrollo como la planeación o diseño cómo fue el ejemplo de mi programa 3.

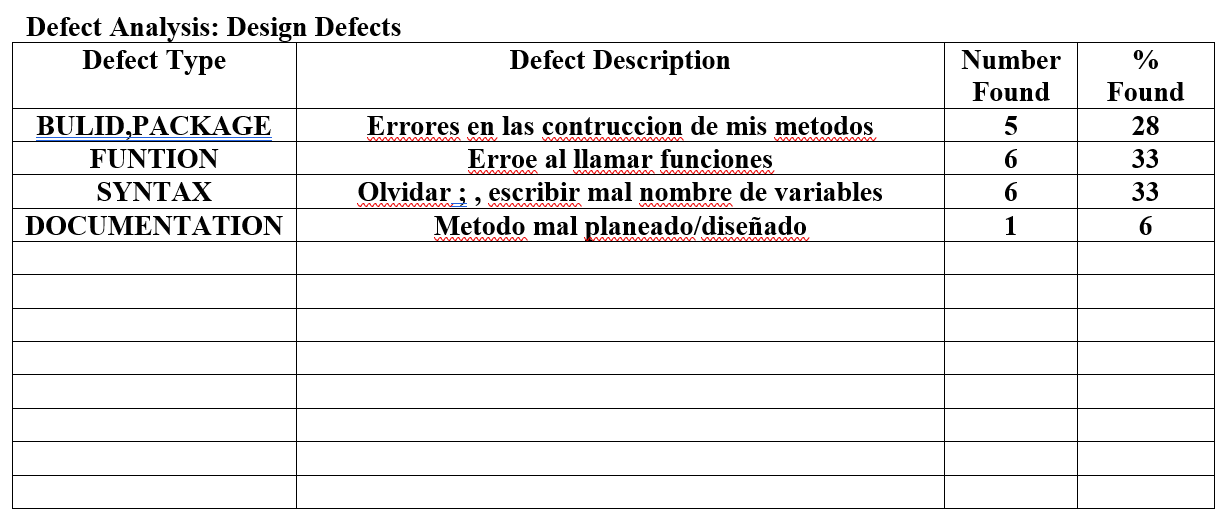
Siguiente pregunta es ¿en qué momento se identificaron los defectos dentro de la compilación o dentro del tiempo del test? en el caso personal la mayoría de estos fueron encontrados durante el tiempo de configuración si no es que todos esto para dejar muy en claro que el tiempo del test es sólo para hacer pruebas en donde ningún error puedes de quedar sin resolverse.



Y ¿en qué momentos fueron inyectados estos errores? en donde se puede observar los tipos de errores y ejemplos de ellos en la siguiente tabla.



I ademas ¿de que tipo son estos errores detectados?, la sigueinte tabla nos da la respuesta



## Proceso de posmortem:

Durante el desarrollo del proceso del post mortem más allá de la recolección de archivos ha sido mente difícil poder identificar de dónde obtienen los archivos en algunos casos hasta ahora lo ha sido fácil identificar de manera muy intuitiva fácilmente este encontrarlos, pero la mayoría han sido muy difíciles de encontrar.

# Resolución de preguntas:

### Para cada uno de los programas 3 y 4, ¿cómo se compara el tamaño aproximado estimado con el tamaño real añadido y modificado?

Los programas 3 y 4 son realmente parecidos en la estimación para el tamaño 3 fue demasiado grande comparado con el programa 4, al hacer el programa 4 una variante del programa 3 esto permitió hacer una estimación de tamaño y tiempo mucho más certera.

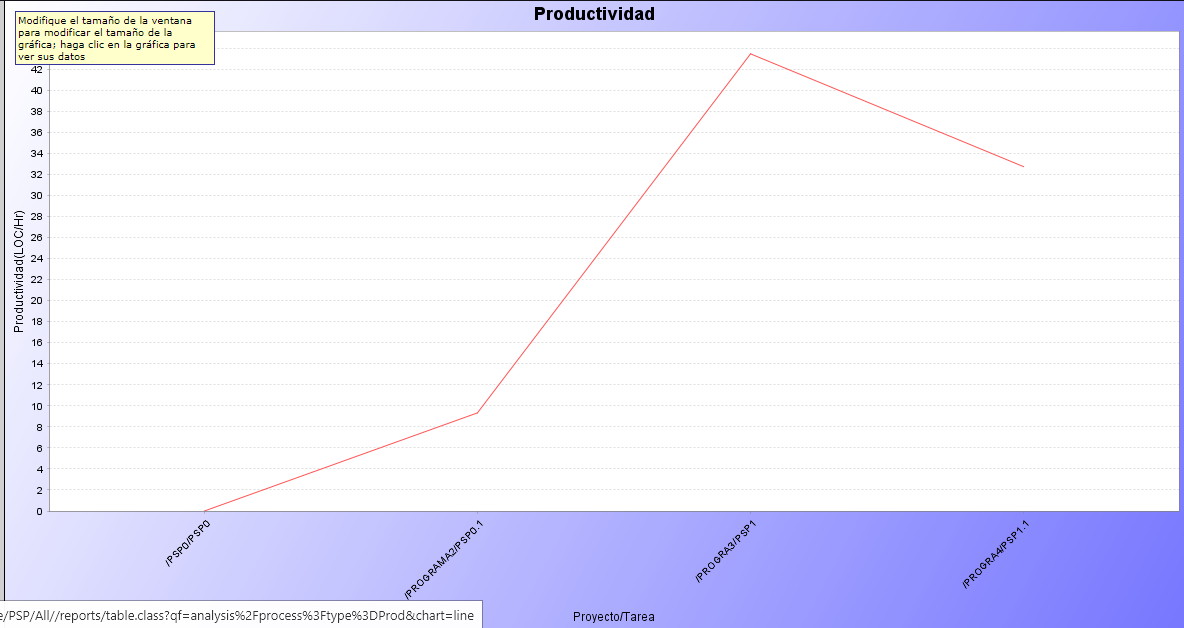
### Para cada uno de los programas 3 y 4, si el tamaño estimado del proxy no se acerca al tamaño real añadido y modificado, determine por qué.

El programa 3 tiene un margen de error en el cálculo del tamaño mucho más grande que el programa 4 una significable mejoría en el cálculo del tamaño para este último programa teniendo un margen de error mucho más pequeño.

### Basándose en el análisis anterior, ¿qué es lo que podría hacer para mejorar la precisión de sus estimaciones?

Las estimaciones pueden mejorar siempre y cuando se haya trabajado con alguna versión ligeramente diferente o con un proceso similar a la hora de calcular el tamaño de un proyecto.

### Para los cuatro primeros programas, ¿cuál fue la estabilidad de su productividad?



Según la gráfica anterior se puede observar un incremento en cuanto a la productividad y una ligera disminución en el último programa esto debido a que el cálculo entre el tamaño y el número de defectos tiene una correlación respecto al programa anterior que es un poco más grande pero con el mismo número de defectos.

### En los cuatro primeros programas, si su productividad varió, ¿están los cambios en la productividad relacionados con la cantidad de tiempo que pasó corrigiendo defectos en las fases de compilación y prueba?

Sí ha habido un avance Ana producir en los nuevos proyectos sin embargo aparecen en algunos casos nuevos tipos de errores No obstante la verificación de los anteriores errores ha sido un paso adelante para poder aumentar la productividad, pero si bien es cierto la mayor parte del tiempo en el periodo de compilación se han corregido todos los errores de manera que durante el tiempo de pruebas no se encuentra ningún error

### ¿Cómo se relaciona la precisión de sus estimaciones de tiempo con la precisión de sus estimaciones de tamaño?

Estimación del tiempo es bastante justa bastante precisa con un margen de error bastante pequeña en coordinación con las estimaciones de tamaño, pero la relación que tienen esos es que a mayor tiempo el tamaño del proyecto también es más grande.

### ¿Qué tipos de defectos inyectó con más frecuencia en la fase de diseño?

La planeación de un método que quizás requería modificaciones en sus parámetros o en la forma en cómo se debería mandar a llamar o ejecutar

### ¿Qué tipos de defectos inyectó con más frecuencia en la fase de código?

La finalización de alguna sentencia con; así como errores de sintaxis con alguna librería que se estuviera utilizando para dicho proyecto

### ¿Qué tipos de defectos eliminó principalmente en la fase de compilación?

El 100% de los errores fueron erradicados durante la fase de compilación

### ¿Qué tipos de defectos eliminó principalmente en la fase de prueba?

Ninguno porque todos fueron eliminados en la fase anterior

### ¿Qué categoría de defectos inyectados tuvo el mayor tiempo medio de reparación?

Los errores de sintaxis, de función y documentación

### ¿Qué categoría de defectos inyectados tuvo el mayor tiempo total de reparación?

Sintaxis

### ¿Encontró el compilador todos sus defectos de tipo 20 (sintaxis/tipología)?

Durante el tiempo de compilación estos fueron debidamente señalizados

## Conclusiones:

Todo este proceso ha llevado a una inminente mejora en cuanto a el tiempo d desarrollo y estimación del tamaño de los proyectos eso sin duda será una mayor ayuda cuando se quiere estimar costos de algún tipo de proyecto, además de que la planeación nos ayuda a definir mucho mejor en algunos casos la forma en cómo subdividir los procesos los métodos y por supuesto planificar mejor la estructura de nuestro proyecto.

