

**UNIVERSIDAD TECNOLÓGICA DE**

**SAN LUIS RIO COLORADO**

**PRACTICE 7**

**MTRO. RICARDO ALEJANDRO SOTO MORALES**

**ALUMNO: VICTOR MANUEL GALVAN COVARRUBIAS**

**TECNOLOGÍAS DE LA INFORMACIÓN**

**ÁREA DESARROLLO DE SOFTWARE MULTIPLATAFORMA**

San Luis Rio Colorado, Sonora Abril, 2021

|  |  |
| --- | --- |
| **Guión del proceso PSP** | |
| Entradas requeridas | * La descripción del problema. * Tabla Resumen del Plan del Proyecto PSP. * Una copia de la lista de comprobación para la revisión de código. * Datos de tamaños y tiempos reales de programas anteriores. * Cuaderno de Registro de tiempos. * Cuaderno de Registro de Defectos |
| 1 Planificación | * Obtén una descripción de las funciones del programa. * Estima las LOC máx., mín., total requeridas. * Determina los minutos/LOC. * Calcula los tiempos de desarrollo máx., mín. y total. * Estima los defectos a introducir y eliminar en cada fase. * Escribe lso datos del plan en la tabla Resumen del Plan del Proyecto. * Anota el tiempo de planificación en el Cuaderno de Registro de Tiempos. |
| 2 Diseño | * Diseña el programa. * Anota el diseño en el formato especificado. * Anota el tiempo de diseño en el Cuaderno de Registro de Tiempos. |
| 3 Codificación | * Implementa el diseño. * Utiliza un formato estándar para introducir el código. * Anota el tiempo de codificación en el Cuadero de Registro de Tiempos. |
| 4 Revisión de código | * Revisar completamente el código fuente. * Seguir el guión de revisión de códig de la lista de comprobación. * Corregir y registrar todos los defectos encontrados. * Registrar el tiemop de revisión en el Cuaderno de Registro de Tiempos. |
| 5 Compilación | * Compila el programa. * Corrige y registra todos los errores encontrados. * Anota el tiempo de revisión en el Cuaderno de Registro de Tiempos. |
| 6 Pruebas | * Prueba el programa. * Corrige y registra todos los errores encontrados. * Anota el tiempo de revisión en el Cuaderno de Registro de Tiempos. |
| 7 Postmorten | * Corrige y registra todos los errores encontrados.Completa la tabla Resumen del Plan del Proyecto con los datos de tiempo, tamaño y defectos reales. * Revisa los datos de defectos y actualiza la lista de comprobación para la revisión de código. * Anota el tiempo postmortem en el Cuaderno de Registro de Tiempos. |
| Criterios de salida | * Programa probado a fondo. * Diseño adecuadamente documentado. * Lista de comprobación para la revisión de código completa. * Listao completo del programa. * Resumen del Plan del Proyecto completo. * Cuaderno de Registro de tiempos y defectos completos. |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **REGISTRO DE TIEMPOS** | | | | | | | | |
| **Fecha** | **Inicio** | **Fin** | **Interrupción** | **Tiempo** | **Actividad** | **Comentarios** | **C** | **U** |
| 15/03 | 09:00 | 12:00 | 30+5+2+3 | 140 | Planificación |  |  |  |
| 16/03 | 9:05 | 16:30 | 10+10+5+2+3 | 415 | Diseño | Diseño del CRUD de la agenda |  |  |
| 17/03 | 09:00 | 18:00 | 10+30 | 500 | Codificación | Codificación CREATE y REAS |  |  |
| 18/03 | 8:00 | 18:37 | 5+30 | 602 | Codificación | Codificación UDATE |  |  |
| 19/03 | 8:00 | 18:00 | 30 | 570 | Codificación | Codificación DELETE |  |  |
| 22/03 | 8:00 | 15:00 | 10 | 410 | Revisión de código | Revisión al CRUD |  |  |
| 23/03 | 8:00 | 15:30 | 8 | 442 | Revisión de código | Revisión al diseño |  |  |
| 24/03 | 10:00 | 13:00 |  | 180 | Compilación |  |  |  |
| 25/03 | 12:00 | 15:00 |  | 180 | Pruebas | Pruebas de funcionalidad |  |  |
| 26/03 | 12:00 | 15:00 | 5 | 175 | Pruebas | Pruebas de rendimiento |  |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **RESUMEN SEMANAL** | | | | | | | **Semana:** 15/03 | | |
| **Tarea** | **Planificación** | **Diseño** | **Codificación** | **Revisión de código** | **Compilación** | **Pruebas** |  |  | **Total** |
| **Fecha** |
| **Lunes** | 140 |  |  |  |  |  |  |  | 140 |
| **Martes** |  | 415 |  |  |  |  |  |  | 415 |
| **Miércoles** |  |  | 500 |  |  |  |  |  | 500 |
| **Jueves** |  |  | 602 |  |  |  |  |  | 602 |
| **Viernes** |  |  | 570 |  |  |  |  |  | 570 |
| **Sábado** |  |  |  |  |  |  |  |  |  |
| **Domingo** |  |  |  |  |  |  |  |  |  |
| **Totales** | 140 | 415 | 1,672 |  |  |  |  |  | 2,227 |
| Número de semanas (número anterior +1) : 1 | | | | | | | | | |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **RESUMEN SEMANAL** | | | | | | | **Semana:** 22/03 | | |
| **Tarea** | **Planificación** | **Diseño** | **Codificación** | **Revisión de código** | **Compilación** | **Pruebas** |  |  | **Total** |
| **Fecha** |
| **Lunes** |  |  |  | 410 |  |  |  |  | 410 |
| **Martes** |  |  |  | 442 |  |  |  |  | 442 |
| **Miércoles** |  |  |  |  | 180 |  |  |  | 180 |
| **Jueves** |  |  |  |  |  | 180 |  |  | 180 |
| **Viernes** |  |  |  |  |  | 175 |  |  | 175 |
| **Sábado** |  |  |  |  |  |  |  |  |  |
| **Domingo** |  |  |  |  |  |  |  |  |  |
| **Totales** |  |  |  | 852 | 180 | 355 |  |  | 1,387 |
| Número de semanas (número anterior +1) : 2 | | | | | | | | | |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **RESUMEN SEMANAS ANTERIORES** | | | | | | | | | |
| **Total** | 140 | 415 | 1,672 |  |  |  |  |  | 2,227 |
| **Media** | 140 | 415 | 1,672 |  |  |  |  |  | 2,227 |
| **Máximo** | 140 | 415 | 1,672 |  |  |  |  |  | 2,227 |
| **Mínimo** | 140 | 415 | 1,672 |  |  |  |  |  | 2,227 |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **RESUMEN INCLUYENDO LA ÚLTIMA SEMANA** | | | | | | | | | |
| **Total** | 140 | 415 | 1,672 | 852 | 180 | 355 |  |  | 3,614 |
| **Media** | 140 | 415 | 1,672 | 852 | 180 | 355 |  |  | 3,614 |
| **Máximo** | 140 | 415 | 1,672 | 852 | 180 | 355 |  |  | 3,614 |
| **Mínimo** | 140 | 415 | 1,672 | 852 | 180 | 355 |  |  | 3,614 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **CUADERNO DE TRABAJOS** | | | | | | | | | | | | |
| **Trabajo** | **Fecha** | **Proceso** | **Estimado** | | **Real** | | | **Hasta la fecha** | | | | |
|  |  |  | Tiempo | Unidades | Tiempo | Unidades | Velocidad | Tiempo | Unidades | Velocidad | Máx. | Mín. |
| 1 | 15/03 | **Planificación** | 200 | 1 | 140 | 1 | 140 | 140 | 1 | 140 | 140 | 140 |
| **Planificación** | | | | | | | | | | | |
| 2 | 16/03 | **Diseño** | 500 | 1 | 415 | 1 | 415 | 415 | 1 | 415 | 415 | 415 |
| **Diseño** | | | | | | | | | | | |
| 3 | 17/03 | **Codificación** | 500 | 1 | 500 | 1 | 500 | 500 | 1 | 500 | 500 | 500 |
| **Codificación** | | | | | | | | | | | |
| 4 | 18/03 | **Codificación** | 600 | 1 | 602 | 1 | 602 | 602 | 1 | 602 | 602 | 602 |
| **Codificación** | | | | | | | | | | | |
| 5 | 19/03 | **Codificación** | 600 | 1 | 570 | 1 | 570 | 570 | 1 | 570 | 570 | 570 |
| **Codificación** | | | | | | | | | | | |
| 6 | 22/03 | **Revisión de código** | 600 | 1 | 410 | 1 | 410 | 410 | 1 | 410 | 410 | 410 |
| **Revisión de código** | | | | | | | | | | | |
| 7 | 23/03 | **Revisión de código** | 500 | 1 | 442 | 1 | 442 | 442 | 1 | 442 | 442 | 442 |
| **Revisión de código** | | | | | | | | | | | |
| 8 | 24/03 | **Compilación** | 60 | 1 | 180 | 1 | 180 | 180 | 1 | 180 | 180 | 180 |
| **Compilación** | | | | | | | | | | | |
| 9 | 25/03 | **Pruebas** | 300 | 1 | 180 | 1 | 180 | 180 | 1 | 180 | 180 | 180 |
| **Pruebas** | | | | | | | | | | | |
| 10 | 26/03 | **Pruebas** | 200 | 1 | 175 | 1 | 175 | 175 | 1 | 175 | 175 | 175 |
| **Pruebas** | | | | | | | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **EJEMPLO DE ESTIMACIÓN DEL TAMAÑO** | | | | | |
| Programa | LOC | Func. estimadas | Mín. | Med. | Máx. |
| Bucles |  |  |  |  |  |
| 8 | 40 | Bucle while |  |  |  |
| Case |  |  |  |  |  |
| 16 | 40 | Sentencia case sencilla | 4 | 7 | 10 |
| 1 | 10 | Sentencia case grande |  |  |  |
| Calc. |  |  |  |  |  |
| 40 | 100 | Cálculo pequeño | 8 | 15 | 19 |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| Total | 150 |  | 12 | 22 | 29 |
| Este programa tiene ocho sentencias case sencillas, ocho bucles y cuarenta cálculos. Asumo que, como máximo, el tamaño se obtendrá sumando estos tamaños típicos, 40+40+10+100=150 LOC. Para el valor mínimo, asumo que estas funciones podrán combinarse más efectivamente que cuando están como elementos separados. Esto nos da 12 LOC como valor mínimo. 22 LOC es el punto medio entre los dos valores anteriores. | | | | | |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **PRESUPUESTO SEMANAL DE TIEMPO (1)** | | | | | | **Semana :15/03** | | |
| **Tarea** | **Planificación** | **Diseño** | **Codificación** | **Revisión de código** | **Compilación** | **Pruebas** |  |  | **Total** |
| **Fecha** |
| **Lunes** | 09:00-12:00 |  |  |  |  |  |  |  | 140 |
| **Martes** |  | 9:05-16:30 |  |  |  |  |  |  | 415 |
| **Miércoles** |  |  | 09:00-18:00 |  |  |  |  |  | 500 |
| **Jueves** |  |  | 8:00-18:37 |  |  |  |  |  | 602 |
| **Viernes** |  |  | 8:00-18:00 |  |  |  |  |  | 570 |
| **Sábado** |  |  |  |  |  |  |  |  |  |
| **Domingo** |  |  |  |  |  |  |  |  |  |
| **Totales** | 140 | 415 | 1,672 |  |  |  |  |  | 2,227 |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **PRESUPUESTO SEMANAL DE TIEMPO (2)** | | | | | | **Semana :22/03** | | |
| **Tarea** | **Planificación** | **Diseño** | **Codificación** | **Revisión de código** | **Compilación** | **Pruebas** |  |  | **Total** |
| **Fecha** |
| **Lunes** |  |  |  | 8:00-15:00 |  |  |  |  | 410 |
| **Martes** |  |  |  | 8:00-15:30 |  |  |  |  | 442 |
| **Miércoles** |  |  |  |  | 10:00-13:00 |  |  |  | 180 |
| **Jueves** |  |  |  |  | 12:00-15:00 |  |  |  | 180 |
| **Viernes** |  |  |  |  |  | 12:00-15:00 |  |  | 175 |
| **Sábado** |  |  |  |  |  |  |  |  |  |
| **Domingo** |  |  |  |  |  |  |  |  |  |
| **Totales** |  |  |  | 852 | 180 | 355 |  |  | 1,387 |

|  |  |  |
| --- | --- | --- |
| **EJEMPLO DE PRESUPUESTO SEMANAL DE TIEMPO (1)** | | **Semana :** 15/03 |
| **Actividad** | **Minutos estimados** | **Minutos reales** |
| **Planificación** | 200 | 140 |
| **Diseño** | 500 | 415 |
| **Codificación** | 1700 | 1672 |
| **Total** | 2400 | 2227 |

|  |  |  |
| --- | --- | --- |
| **EJEMPLO DE PRESUPUESTO SEMANAL DE TIEMPO (2)** | | **Semana :** 22/03 |
| **Actividad** | **Minutos estimados** | **Minutos reales** |
| **Revisión de código** | 1100 | 852 |
| **Compilación** | 60 | 180 |
| **Pruebas** | 500 | 355 |
| **Total** | 2200 | 1387 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Compromisos** | | | | |
| **Fecha comprometida** | **Compromiso** | **¿Con quién?** | **Horas** | **Consigo** |
| **Semanal** |  |  |  |  |
| V | Entregar avances | Cliente | 1 | Confirmar |
| L,M,M,J,V | Trabajar | Cliente | 10 | Avanzar |
| **Otros** |  |  |  |  |
|  | Entregar agenda | Cliente | 1 | Terminar |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Programa: Agenda telefonica** | | | | | | | | **Fecha: 26/03** | | |
| **Descripción:**  **Agenda telefónica donde se pueden registras datos importantes de personas .** | | | | | | | | | | |
| **Resumen** | **Plan** | | | **Real** | | | | **Hasta la fecha** | | |
| Minutos/LOC | 4000 | | | 3650 | | | | 3650 | | |
| LOC/Hora | 67 | | | 60 | | | | 60 | | |
| Defectos/KLOC | 10 | | | 4 | | | | 4 | | |
| Rendimiento |  | | |  | | | |  | | |
| Valoración/Fallo |  | | |  | | | |  | | |
| **Tamaño programa (LOC)** | **Plan**  **400** | | | **Real**  **500** | | | | **Hasta la fecha**  **500** | | |
| Total nuevo & cambiado | 400 | | | 500 | | | | 500 | | |
| Tamaño máximo | 600 | | |  | | | | | | |
| Tamaño mínimo | 300 | | |
| **Tiempo por Fase (min.)** | **Plan** | | **Real** | | | **Hasta la fecha** | | | **% Hasta la fecha** | |
| Planificación | 300 | | 140 | | | 140 | | | 3.76% | |
| Diseño | 300 | | 415 | | | 415 | | | 11.17% | |
| Codificación | 900 | | 1,672 | | | 1,672 | | | 45.01% | |
| Revisión del código | 500 | | 852 | | | 852 | | | 22.94% | |
| Compilación | 300 | | 180 | | | 180 | | | 4.84% | |
| Pruebas | 300 | | 355 | | | 355 | | | 9.55% | |
| Postmorten | 100 | | 100 | | | 100 | | | 2.69% | |
| Total | 2700 | | 3714 | | | 3714 | | | 100% | |
| Tiempo máximo | 4000 | |  | | | | | | | |
| Tiempo mínimo | 2000 | |
| **Defectos Introducidos** | **Plan** | **Actual** | | | **Hasta la fecha** | | **% Hasta la fecha** | | | **Def./Hora** |
| Planificación |  |  | | |  | |  | | |  |
| Diseño | 5 | 0 | | | 0 | |  | | |  |
| Codificación | 10 | 4 | | | 4 | | 100% | | |  |
| Revisión del código |  |  | | |  | |  | | |  |
| Compilación |  |  | | |  | |  | | |
| Pruebas |  |  | | |  | |  | | |
| Total |  |  | | |  | | 100% | | |
| Defectos eliminados | **Plan** | **Actual** | | | **Hasta la fecha** | | **% Hasta la fecha** | | | **Def./Hora** |
| Planificación |  |  | | |  | |  | | |  |
| Diseño |  |  | | |  | |  | | |
| Codificación |  |  | | |  | |  | | |
| Revisión del código | 5 | 0 | | | 0 | |  | | |  |
| Compilación | 10 | 4 | | | 4 | | 100% | | |  |
| Pruebas | 0 |  | | |  | |  | | |  |
| Total |  |  | | |  | | 100% | | |  |

|  |  |  |
| --- | --- | --- |
| Tipos de defectos | | |
| Nº de tipo | Nombre del tipo | Descripción |
| 10 | Documentación | Comentarios, mensajes |
| 20 | Sintaxis | Ortografía, puntuación, erratas, formato de las instrucciones |
| 30 | Construir, paquetes | Gestión del cambio, librerías, control de versión |
| 40 | Asignación | Declaración, nombres duplicados, ámbito, límites |
| 50 | Interfaz | Llamadas a procedimientos y referencias, E/S, formatos de usuario |
| 60 | Chequeo | Mensajes de error, chequeos inadecuados |
| 70 | Datos | Estructura, contenido |
| 80 | Función | Lógica, punteros, bucles, recursión, computación, defectos de la función |
| 90 | Sistema | Configuración, temporización, memoria |
| 100 | Entorno | Diseño, compilación, pruebas y otros problemas que soporta el sistema |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Registro de Defectos | | | | | | |
| Fecha | Número | Tipo | Introducido | Eliminado | Tiempo de corrección | Defecto corregido |
| 23/03/21 | 1 | 40 | Codificación | Compilación | 1 |  |
| Descripción: Omitido ; | | | | | | |
| 23/03/21 | 2 | 20 | Codificación | Compilación | 1 |  |
| Descripción: Omitido ; | | | | | | |
| 24/03/21 | 3 | 40 | Codificación | Compilación | 1 |  |
| Descripción: Instancia hecha incorrectamente; | | | | | | |
| 25/03/21 | 4 | 40 | Codificación | Compilación | 1 |  |
| Descripción: Declaración de variable incorrecta; | | | | | | |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Lista de comprobación** | | | | | | Fecha:19/03 | |
| Propósito | Guía | # | # | # | # | Hasta la fecha | % Hasta la fecha |
| Método | Cuando completes cada paso de la revisión, antoa el número de defectos que has encontrado de cada tipo en la casilla de la derecha. Si no hay ninguno, anota un control en la casilla de la derecha.  Completa la lista de comprobación para un programa, clase, objeto o método antes de comenzar a revisar la siguiente. | X |  |  |  |  | 50% |
| Completo | Verfica que todas las funciones del diseño están programadas | X |  |  |  |  | 50% |
| Includes | Verifica que las sentencias import están completas | X |  |  |  |  | 50% |
| Inicialización | Comprobar la inicialización de parámetros y variables:   * Al inicio del programa. * Al comenzar cada bucle. * En la entrada a un procedimiento o función. | X |  |  |  |  | 50% |
| Llamadas | Comprobar los formatos de las llamadas a los procedimientos:   * Signos de puntuación. * Parámetros. | X |  |  |  |  | 50% |
| Nombres | Comprobar la ortografía de los nombres y su utilización:   * ¿Es consistente? * ¿Está dentro del ámbito declarado? | X |  |  |  |  | 50% |
| Formato de salida | Comprobar el formato de salida:   * ¿Es adecuado el salto de línea? * ¿Es adecuado el espaciado? | X |  |  |  |  | 50% |
| Parejas de () {} [] | Asegúrate que los () {} [] son adecuados y están balanceados | X |  |  |  |  | 50% |
| Operadores lógicos | * Verfiicar la utilización correcta de todos los operadores lógicos. * Comprobar que cada función lógica tiene (). | X |  |  |  |  | 50% |
| Comprobación línea a línea | Comprobar cada línea de código.   * Sistaxis de la instrucción. * Utilización adecuada de los ; * Comprobar que los ; no se escriben como : * Otros signos de puntuación. | X |  |  |  |  | 50% |
| Estándares | Asegurarse que cada programa se adapta a los estándares de codificación | X |  |  |  |  | 50% |
| Apertura y cierre de ficheros | Verfica que todos los ficheros son:   * Declarados de forma adecuada. * Abiertos. * Cerrados. | X |  |  |  |  | 50% |
| Global | Hacer una revisión global al progara para comprobar los resultados del sistema y problemas inesperados. | X |  |  |  |  | 50% |
| Totales |  |  |  |  |  |  | 50% |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Lista de comprobación** | | | | | | Fecha:26/03 | |
| Propósito | Guía | # | # | # | # | Hasta la fecha | % Hasta la fecha |
| Método | Cuando completes cada paso de la revisión, antoa el número de defectos que has encontrado de cada tipo en la casilla de la derecha. Si no hay ninguno, anota un control en la casilla de la derecha.  Completa la lista de comprobación para un programa, clase, objeto o método antes de comenzar a revisar la siguiente. | X |  |  |  |  | 100% |
| Completo | Verfica que todas las funciones del diseño están programadas | X |  |  |  |  | 100% |
| Includes | Verifica que las sentencias import están completas | X |  |  |  |  | 100% |
| Inicialización | Comprobar la inicialización de parámetros y variables:   * Al inicio del programa. * Al comenzar cada bucle. * En la entrada a un procedimiento o función. | X |  |  |  |  | 100% |
| Llamadas | Comprobar los formatos de las llamadas a los procedimientos:   * Signos de puntuación. * Parámetros. | X |  |  |  |  | 100% |
| Nombres | Comprobar la ortografía de los nombres y su utilización:   * ¿Es consistente? * ¿Está dentro del ámbito declarado? | X |  |  |  |  | 100% |
| Formato de salida | Comprobar el formato de salida:   * ¿Es adecuado el salto de línea? * ¿Es adecuado el espaciado? | X |  |  |  |  | 100% |
| Parejas de () {} [] | Asegúrate que los () {} [] son adecuados y están balanceados | X |  |  |  |  | 100% |
| Operadores lógicos | * Verfiicar la utilización correcta de todos los operadores lógicos. * Comprobar que cada función lógica tiene (). | X |  |  |  |  | 100% |
| Comprobación línea a línea | Comprobar cada línea de código.   * Sistaxis de la instrucción. * Utilización adecuada de los ; * Comprobar que los ; no se escriben como : * Otros signos de puntuación. | X |  |  |  |  | 100% |
| Estándares | Asegurarse que cada programa se adapta a los estándares de codificación | X |  |  |  |  | 100% |
| Apertura y cierre de ficheros | Verfica que todos los ficheros son:   * Declarados de forma adecuada. * Abiertos. * Cerrados. | X |  |  |  |  | 100% |
| Global | Hacer una revisión global al progara para comprobar los resultados del sistema y problemas inesperados. | x |  |  |  |  | 100% |
| Totales |  |  |  |  |  |  | 100% |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Análisis de Errores | | | | | | Fecha: | |
| Tipo | Introducido | | | Eliminado | | | Omitido |
|  | Diseñar | Codificar | Otros | Revisar | Compilar | Pruebas | En revisión |
| 10 |  |  |  |  |  |  |  |
| 20 |  | 1 |  |  |  |  | 1 |
| 30 |  |  |  |  |  |  |  |
| 40 |  | 3 |  |  | 2 |  | 1 |
| 50 |  |  |  |  |  |  |  |
| 60 |  |  |  |  |  |  |  |
| 70 |  |  |  |  |  |  |  |
| 80 |  |  |  |  |  |  |  |
| 90 |  |  |  |  |  |  |  |
| 100 |  |  |  |  |  |  |  |
| Total |  | 4 |  |  | 2 |  | 4 |
| Programa |  | | | | | | |
| Agenda telefónica |  | 4 |  |  | 2 |  | 4 |

|  |  |  |
| --- | --- | --- |
| **Datos de Defectos** | | Fecha: 26/03 |
| Nº de programa | Defectos (D) | LOC |
| 1 | 4 | 20 |
| Total hasta la fecha | 4 | 20 |

The formats or forms used in PSP are very efficient when it comes to productivity. We cannot know what we are not measuring. PSP makes sure that we measure our actual working hours to ensure that we are productive and efficient when developing.

The forms are easy to understand and no advanced programming knowledge is required. They are simply a real record of our work done. The filling of these formats occurs simultaneously with the life cycle of the software. As we continue to advance in the construction of the program, we will also record our times and errors. Helping us in a very clear way by increasing the quality of the programs we make.