

Postmortem Menú Reportes Maestros.

27/04/17

0.1

|  |  |
| --- | --- |
| **Nombre.** | **Matricula** |
| **Jesús Alberto Goiz Barrales.** | **24400085** |
| **Juan Alberto Gutiérrez canto.** | **24400063** |
| **Brenda Robles Antonio.** | **24400073** |
| **Rene Moratilla Montes.** | **24400075** |
| **Guillermo Vivaldo Vazquez** | **24400093** |

**Control De Versiones**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nombre Del Archivo | Versión | Fecha | Autor | Comentarios |
| Csoft-PM\_Menú\_Reporte\_Maestros-270417.docx | 0.1 | 27/04/17 | JAGB | Creación del documento. |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# Índice

[**Índice**](#_kd5bfhkalq0n) **2**

[**I.- Forma PSP Project Plan Summary.**](#_jhy44tc5k06j) **3**

[**II.- Time Recording Log.**](#_7e50s7cf9wc3) **9**

[**III.- Defect Recording Log.**](#_3znysh7) **10**

[**IV.- Diseño.**](#_m4hw86ggwsmr) **13**

[**V.- Design CheckList.**](#_92tu2pewgguj) **18**

[**VI.- Programa Fuente.**](#_6bvnqqrwimi9) **20**

[**VII.- Code CheckList.**](#_87ns93g4124c) **24**

[**VIII.- Reporte de resultados.**](#_bo7s7l2hg3ra) **27**

[**Template de pruebas**](#_1fob9te) **27**

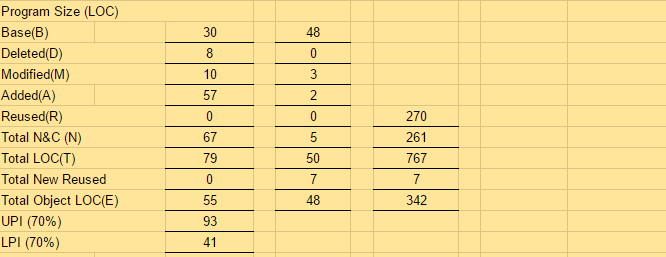
[**IX.- PIP**](#_4ocjjwajumc0) **28**

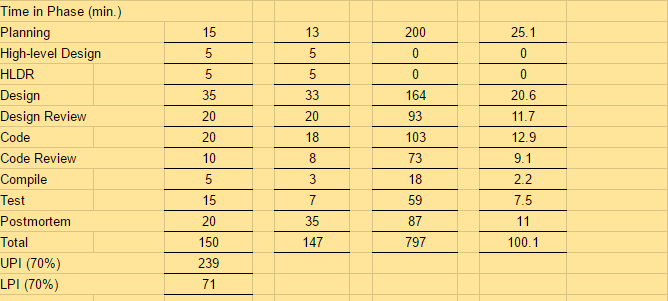
[**X.- Size Estimating Template**](#_w1ych8jaj7as) **29**

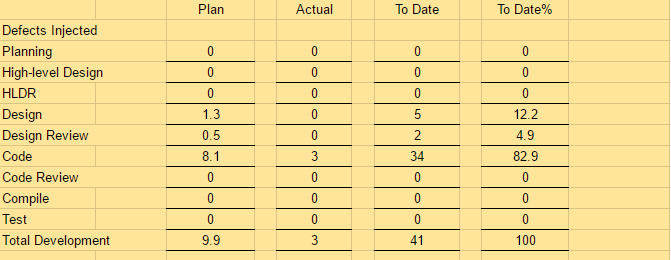
[**XI.- Task and Schedule**](#_195jonwtvlxs) **32**

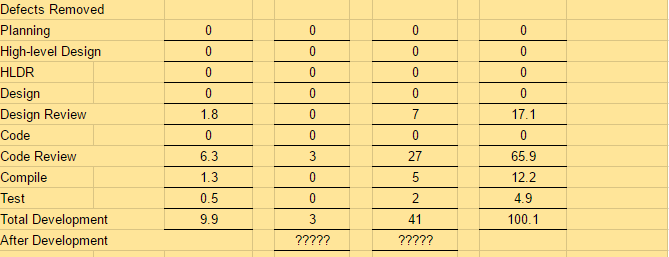
# I.- Forma PSP Project Plan Summary.

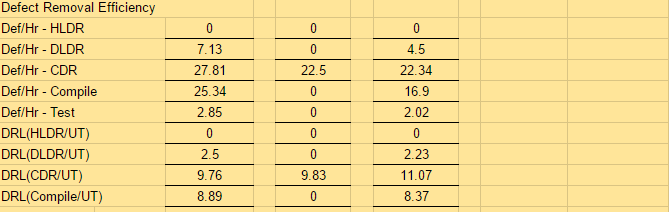
# 



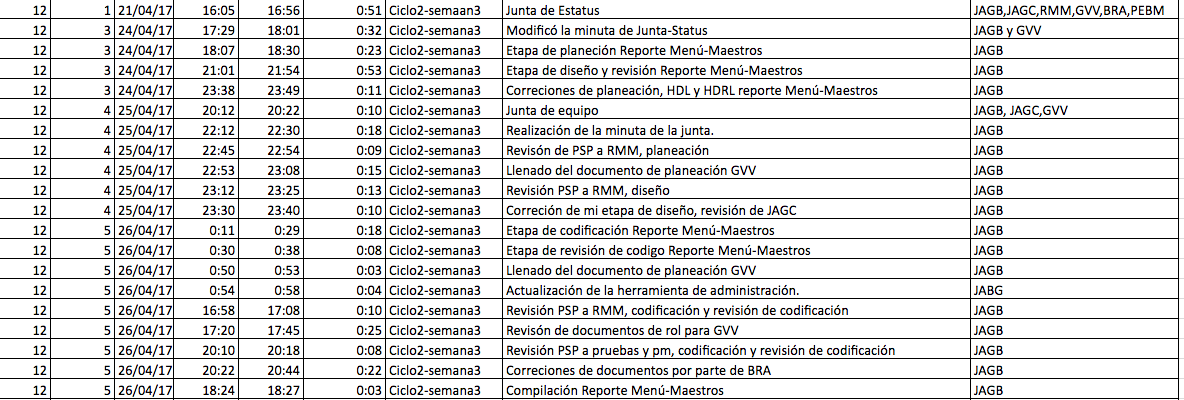


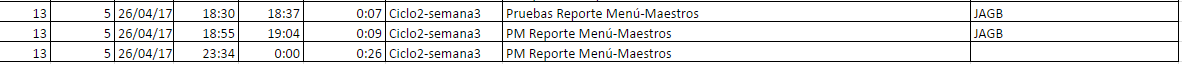




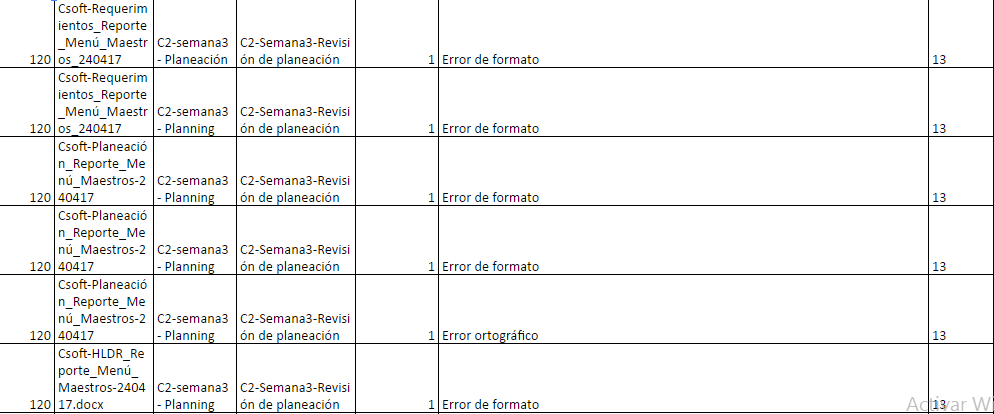


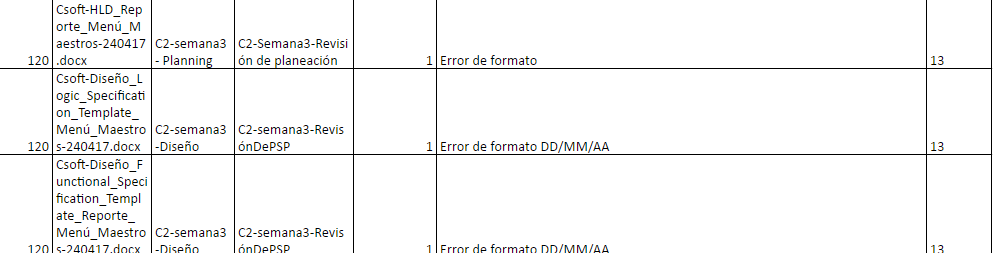
# II.- Time Recording Log.



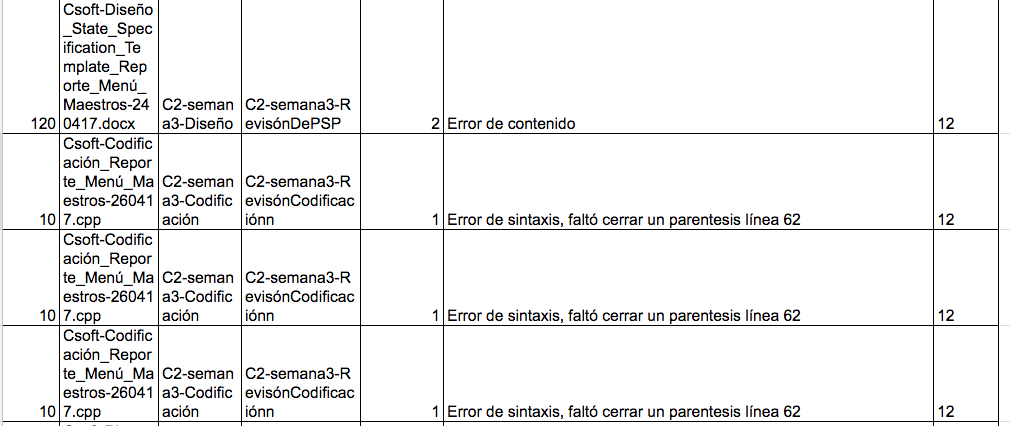


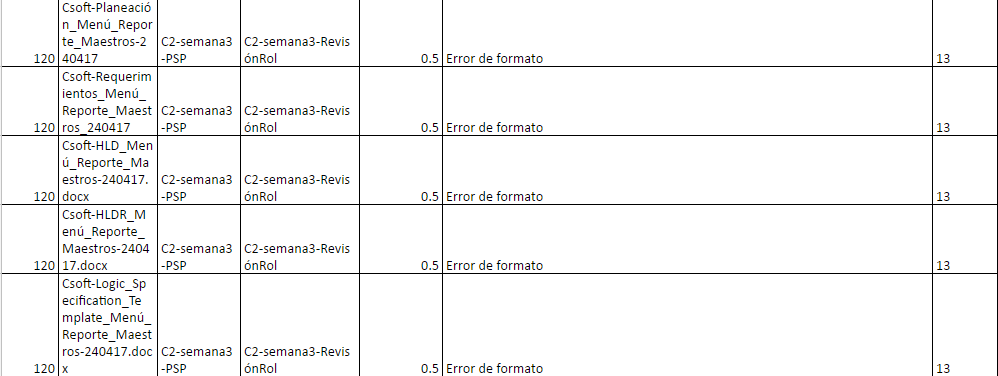
# III.- Defect Recording Log.

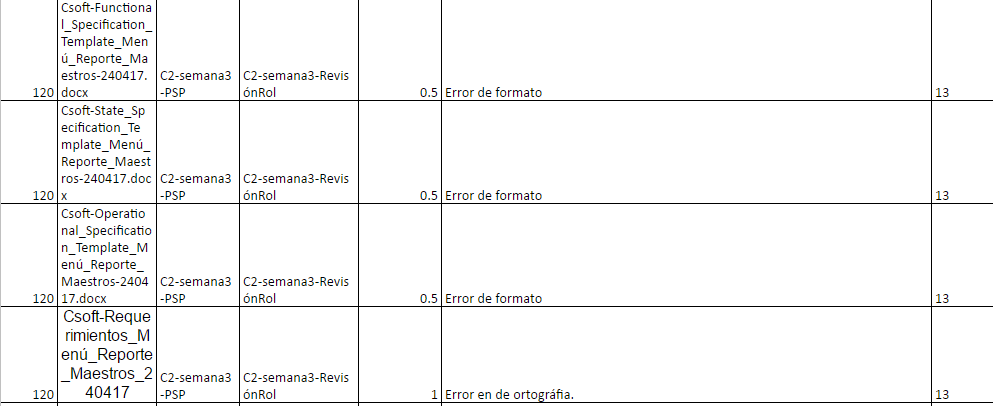












# IV.- Diseño.

**Table C72 Logic Specification Template**

|  |  |  |  |
| --- | --- | --- | --- |
| Student | JAGB | Date | 24/04/17 |
| Program | Menú Reporte Maestros | Program # |  |
| Instructor | PEBM | Language | C |
| Object | Reporte\_Maestros | Function | Reporte\_Maestros |

INCLUDES:

#include <cstdlib>

#include <iostream>

#include <stdlib>

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

Int opcion;

TYPE DEFINITIONS: Using namespace std;

|  |  |
| --- | --- |
| **Declaration:** |  |
| **Reference:** |  |
|  |  |
| **Logic reference numbers** | **Program logic, in pseudocode** |
|  |  |
|  | printf("o---------------------------------------------------------o\n") |
|  | printf("| ================ Reporte Maestros ==================== |\n") |
|  | printfi("o---------------------------------------------------------o\n") |
|  |  |
|  | printf(" | 1)Materias Impartidas 2)Materias disponibles | \n \n")  printf(" | 3)Calificación de Materias 4)Menú de reportes. | \n\n")  printf("o---------------------------------------------------------o \n\n\n") |
|  | Leer la opción de usuario:  Si es entre 1 a 4 |
|  | Opción 1: Manda a llamar la función Reporte\_Materias\_Impartidas |
|  | Opción 2: Manda a llamar la función Reporte\_Materias\_Disponibles |
|  | Opción 3: Manda a llamar la función Reporte\_Califación\_Materias |
|  | Opción 4: Manda a llamar la función Menú\_Reportes |
|  | En caso de ser otra opción se imprimirá un mensaje de error: |
|  |  |
|  | fin |
|  |  |
|  |  |

**Table C70 State Specification Template**

|  |  |  |  |
| --- | --- | --- | --- |
| Student | JAGB | Date | 24/06/17 |
| Program | Menú Reporte Maestros | Program # |  |
| Instructor | PEBM | Language | C |
| Object |  | Routine |  |

|  |  |
| --- | --- |
| **State name** | **Description** |
| Inicio | Inicia el programa. |
| Int Main | Se utiliza para llamar a nuestra función de reporte Menú-Maestros junto con toda su respectiva acción. |
| Reporte\_Maestros. | Se presenta el menú para agregar las opciones de reportes disponibles para los maestros. |
| Fuction/Parameter | Description |
| printf() | Imprimir los datos en pantalla. |
| gets() | Leer datos |
| opción | Indica el tipo de acciones que elige el usuario. |
| Reporte\_Maestros | Ilustra el menú de lo reportes que tiene el maestro. |

# 

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **States/Next States** | | | | **Transition Condition** | | **Action** | |
| Inicio | | | |  | |  | |
|  | Main | | | Ninguna | | Se muestra el menú en pantalla. | |
| Reporte\_Maestros | | | |  | |  | |
|  |  | | | opción=1 | | Reporte\_Materias\_Impartidas() | |
|  |  | | | opción=2 | | Reporte\_Materias\_Disponibles() | |
|  |  | | | opción=3 | | Reporte\_Califación\_Materias() | |
|  |  | | | opción=4 | | Menu\_Reportesl() | |

**Table C66 Operational Scenario Template**

|  |  |  |  |
| --- | --- | --- | --- |
| Student | JAGB | Date | 24/04/17 |
| Program | Menú Reporte Maestros | Program # |  |
| Instructor | PEBM | Language | C |

Construct operational scenarios to cover the normal and abnormal program uses, including user errors.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Scenario Number:1** | | | Se muestra el menú para el Reporte\_Maestros | |
| **Scenario Objective**: Se ilustra el funcionamiento del programa en general | | | | |
| **Source** | **Step** | **Action** | | **Comments** |
| User | 1 | Se ejecuta el .exe | |  |
| System | 2 | Se visualiza el menú de reportes de maestros con 4 opciones. | |  |
| User | 3 | Si se selecciona la opción 1 se visualizará el reporte de materias impartidas. | | Se manda a llamar a la función Reporte\_Materias\_Impartidas. |
| System | 4 | Una vez terminada la rutina se volverá al menú del Reporte de Maestros. | | Se visualizará el menú de 4 opciones. |
| User | 5 | En caso de no haber finalizado con las actividades se escoge la opción 4 y se regresa al menú principal del programa. | | Se manda a llamar la función principal del programa para poder continuar con las actividades. |
| System | 6 | Fin de las rutinas del Reporte Menú-Maestros | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Scenario Number:2** | | | Se muestra el Reportes Menú-Maestros | |
| **Scenario Objective**: Se ilustra el funcionamiento del programa en general | | | | |
| **Source** | **Step** | **Action** | | **Comments** |
| User | 1 | Se ejecuta el .exe | |  |
| System | 2 | Se visualiza el menú de Reporte\_Maestros con 4 opciones. | |  |
| User | 3 | Si se selecciona la opción 2 manda a llamar una funciòn para generar un reporte. | | Se manda a llamar a la función Reporte\_Materias\_Disponibles. |
| System | 4 | Una vez terminada la rutina se volverá al Reportes\_Maestros | | Se visualizará el menú de 4 opciones. |
| User | 5 | En caso de no haber finalizado con las actividades se escoge la opción 4 y se regresa al menú principal del programa. | | Se manda a llamar la función principal del programa para poder continuar con las actividades. |
| System | 6 | Fin de las rutinas del Reporte Menú-Maestros | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Scenario Number:3** | | | Se muestra el menú para el Reporte\_Maestros | |
| **Scenario Objective**: Se ilustra el funcionamiento del programa en general | | | | |
| **Source** | **Step** | **Action** | | **Comments** |
| User | 1 | Se ejecuta el .exe | |  |
| System | 2 | Se muestra el menú para el Reporte\_Maestros | |  |
| User | 3 | Si se selecciona la opción 3 manda a llamar una funciòn para generar un reporte. | | Se manda a llamar a la función Reporte\_Califación\_Materias. |
| System | 4 | Una vez terminada la rutina se volverá al Reportes\_Maestros | | Se visualizará el menú de 4 opciones. |
| User | 5 | En caso de no haber finalizado con las actividades se escoge la opción 4 y se regresa al menú principal del programa. | | Se manda a llamar la función principal del programa para poder continuar con las actividades. |
| System | 6 | Fin de las rutinas del Reporte\_Maestros | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Scenario Number:1** | | | Se muestra el menú para el Reporte\_Maestros | |
| **Scenario Objective**: Se ilustra el funcionamiento del programa en general | | | | |
| **Source** | **Step** | **Action** | | **Comments** |
| User | 1 | Se ejecuta el .exe | |  |
| System | 2 | Se visualiza el menú Reporte\_Maestros | |  |
| User | 3 | Si se selecciona la opción 4 se visualizará el menú principal de los reportes. | | Se manda a llamar a la función del menú principal para ir a el |
| System | 4 | Fin de las rutinas del Reporte\_Maestros | |  |

**Table C68 Functional Specification Template**

|  |  |  |  |
| --- | --- | --- | --- |
| Student | JAGB | Date | 24/04/17 |
| Program | Menú Reporte Maestros | Program # |  |
| Instructor | PEBM | Language | C |

|  |  |  |  |
| --- | --- | --- | --- |
| **Object/Class Name:**  Reporte\_Maestros | | **Parent Classes:**  Ninguna | **Attributes:**  Int opción; |
|  | Method Declaration | Method External Specification | |
|  | Void Reporte\_Maestros | Mostrará el menú para acceder a las funciones de para generar los reportes disponibles de los maestros. | |
|  | Main. | Se manda a llamar la función de Reporte\_Maestros para su visualización. | |
|  | | | |

# V.- Design CheckList.

**PSP2 Design Review Checklist**

|  |  |  |  |
| --- | --- | --- | --- |
| Student | JAGB | Date | 24/04/2017 |
| Program | Menú Reporte Maestros | Program # | Reporte Menú-Maestros |
| Instructor | PEBM | Language | C |

|  |  |
| --- | --- |
| **Purpose** | To guide you in conducting an effective design review |
| **General** | * Review the entire program for each checklist category; do not attempt to review for more than one category at a time! * As you complete each review step, check off that item in the box at the right. * Complete the checklist for one program or program unit before reviewing the next. |

|  |  |  |  |
| --- | --- | --- | --- |
| Complete | Verify that the design covers all of the applicable requirements.   * All specified outputs are produced. * All needed inputs are furnished. * All required includes are stated. |  | Revisado, Se Hicieron correcciones ( JAGB). |
| External Limits | Where the design assumes or relies upon external limits, determine if behavior is correct at nominal values, at limits, and beyond limits. |  |  |
| Logic | * Verify that program sequencing is proper.   Stacks, lists, and so on are in the proper order.  Recursion unwinds properly.   * Verify that all loops are properly initiated, incremented, and terminated. * Examine each conditional statement and verify all cases. |  |  |
| Internal Limits | Where the design assumes or relies upon internal limits, determine if behavior is correct at nominal values, at limits, and beyond limits. |  |  |
| Special Cases | * Check all special cases. * Ensure proper operation with empty, full, minimum, maximum, negative, and cero values for all variables. * Protect against out-of-limits, overflow, and underflow conditions. * Ensure “impossible” conditions are absolutely impossible. * Handle all possible incorrect or error conditions. |  |  |
| Functional Use | * Verify that all functions, procedures, or methods are fully understood and properly used. * Verify that all externally referenced abstractions are precisely defined. |  |  |
| System Considerations | * Verify that the program does not cause system limits to be exceeded. * Verify that all security-sensitive data are from trusted sources. * Verify that all safety conditions conform to the safety specifications. |  |  |
| Names | Verify that   * all special names are clear, defined, and authenticated * the scopes of all variables and parameters are self-evident or defined * all named items are used within their declared scopes |  |  |
| Standards | Ensure that the design conforms to all applicable design standards. |  |  |

# VI.- Programa Fuente.

#include <cstdlib>

#include <iostream>

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

/\*=========================================================\*/

/\*Autor: Jesus Alberto Goiz Barrales \*/

/\*Componente: Reporte\_Maestros \*/

/\*Fecha: 26/04/2017 \*/

/\*Descripción: Menú para realizar los reportes \*/

/\*de los maestros \*/

/\*Versión: 0.1 \*/

/\*=========================================================\*/

/\*==========================================================\*/

/\* El programa contiene lo siguiente: \*/

/\* void Reporte\_Maestros() \*/

/\*==========================================================\*/

/\*Cascarón de las opciones que tendrá el menú Reportes

Maestros \*/

void Reporte\_Maestros()

{

int opcion;

printf("\to---------------------------------------------------------o\n");

printf("\t|================= Reporte Maestros ====================|\n");

printf("\to---------------------------------------------------------o\n");

printf("\to---------------------------------------------------------o\n");

printf("\t| 1)Materias Impartidas 2)Materias disponibles |\n");

printf("\t| 3)Calificaci%cn de Materias 4)Men%c de reportes |\n",162,163);

printf("\to---------------------------------------------------------o\n");

/\*Parte del código que se utiliza para poder validar el valor de entrada que recibe

hasta que este sea el correcto\*/

r:printf("\t Ingrese una opci%Cn valida ---> ",162);

scanf("%d",&opcion);

if ((opcion) <= 0 || (opcion) >4)

{

printf("\t ---------------------------------------------------------------\n");

printf("\t| \*\*\*\* ERROR: Ingrese un n%cmero valido para continuar \*\*\*\* |\n",163);

printf("\t ---------------------------------------------------------------\n\n");

//Limpia el buffer(Evita el ciclo infinito)

while (getchar() != '\n'); goto r;

}

switch(opcion)

{

case 1:

system("cls");

printf("\t Sustituir por Reporte\_Materias\_Impartidas\n");

break;

case 2:

system("cls");

printf("\t Sustituir por Reporte\_Materias\_Disponibles\n");

break;

case 3:

system("cls");

printf("\t Sustituir por Reporte\_Califaci%cn\_Materias\n",162);

break;

case 4:

system("cls");

printf("\t Sustituir por Men%c\_Reportes\n",163);

break;

}

}//39 lineas

//6 lineas

int main()

{

Reporte\_Maestros();

system("PAUSE");

return 0;

}

# VII.- Code CheckList.

**Code Review Checklist**

|  |  |  |  |
| --- | --- | --- | --- |
| Student | JAGB | Date | 26/04/2017 |
| Program | Menú Reporte Maestros | Program # | Reporte Menú-Maestros |
| Instructor | PEBM | Language | C |

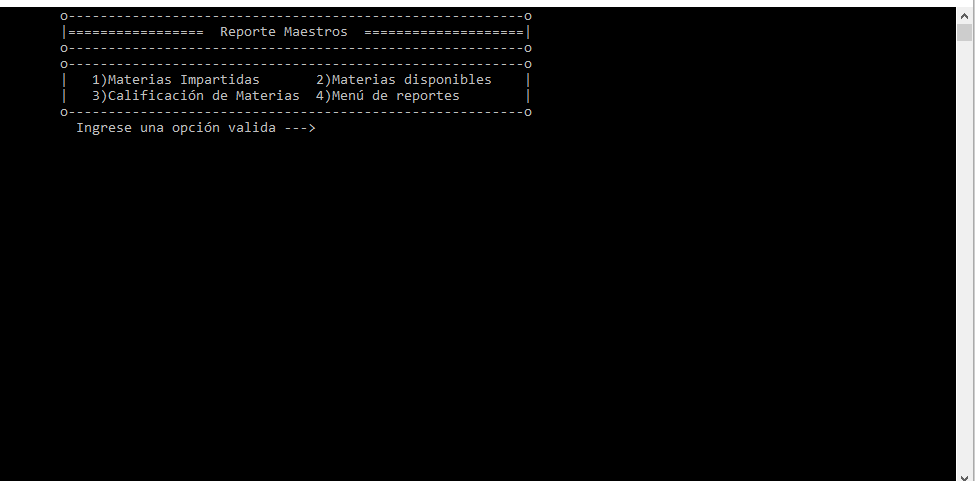
|  |  |
| --- | --- |
| Purpose | To guide you in conducting an effective code review |
| General | - Review the entire program for each checklist category; do not attempt to review for more than one category at a time!  - As you complete each review step, check off that item in the box at the right.  - Complete the checklist for one program or program unit before reviewing the next. |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Complete | Verify that the code covers all of the design. | MB | B | R | M |
| Includes | Verify that the includes are complete. | X |  |  |  |
| Initialization | Check variable and parameter initialization.  - at program initiation  - at start of every loop  - at class/function/procedure entry | x  X  X |  |  |  |
| Calls | Check function call formats.  - pointers  - parameters  - use of ‘&’ | X  X  X |  |  |  |
| Names | Check name spelling and use.  - Is it consistent?  - Is it within the declared scope?  - Do all structures and classes use ‘.’ reference? | x  X |  |  |  |
| Strings | Check that all strings are  - identified by pointers  - terminated by NULL | X  X |  |  |  |
| Pointers | Check that all  - pointers are initialized NULL  - pointers are deleted only after new  - new pointers are always deleted after use | X  X  X |  |  |  |
| Output Format | Check the output format.  - Line stepping is proper.  - Spacing is proper. | X  X |  |  |  |
| () Pairs | Ensure that () are proper and matched. |  | x |  |  |
| Logic Operators | - Verify the proper use of ==, =, ||, and so on.  - Check every logic function for (). | X  X |  |  |  |
| Line-by-line check | Check every line of code for  - instruction syntax  - proper punctuation |  | X  X |  |  |
| Standards | Ensure that the code conforms to the coding standards. | x |  |  |  |
| File Open and Close | Verify that all files are  - properly declared  - opened  - closed | X  X  X |  |  |  |

# VIII.- Reporte de resultados.

# Template de pruebas

Para este Menú no existen pruebas relacionadas debido a que solo es el cascarón, Se incluye la visualización que debe de tener el respectivo menú.

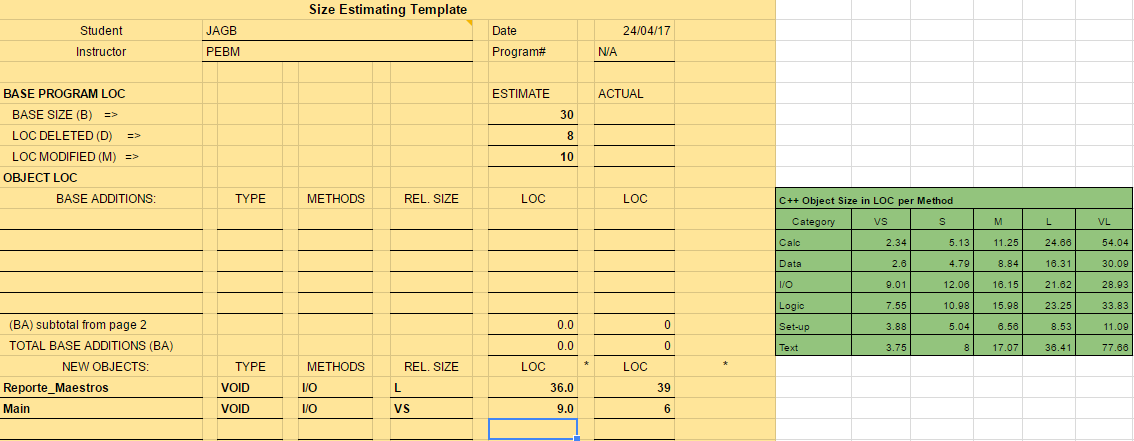


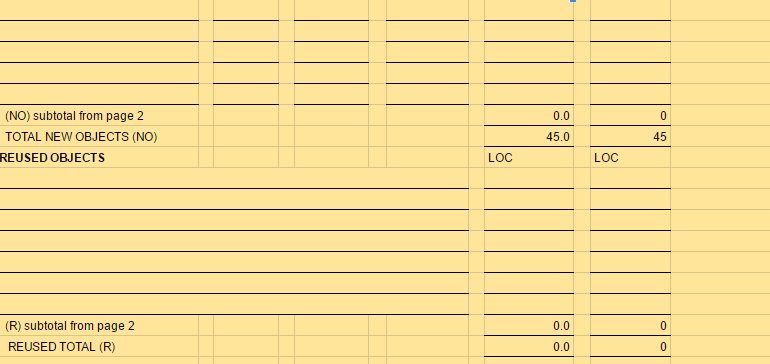
|  |  |
| --- | --- |
| Nombre de la Prueba |  |
| Caso de Uso |  |
| Objetivo de la Prueba |  |
| Descripción de la Prueba |  |
| Test |  |
| Resultados Esperados |  |
| Capturas de pantalla |  |
| Estado caso de prueba | Posibles estados:   * Ejecutado * Exitoso * Fallido * Frenado * Pendiente de ejecución * En construcción |
| Errores obtenidos |  |
| Responsable de prueba |  |
| Comentarios |  |

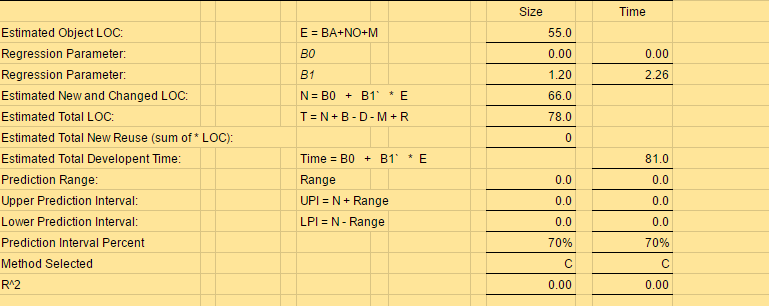
# IX.- PIP

|  |  |  |
| --- | --- | --- |
| Menú-Alumnos | | |
| No. PIP | Descripción del Problema | Descripción de la propuesta. |
| 1 | Mala organición de ideas | Tener mejor panorama al momento de hacer la planeación de nuestro PSP |
| 2 | Confusión de nombres de módulos | Tener una mejor organización en cuestión de análisis del problema. |

# X.- Size Estimating Template







# XI.- Task and Schedule

