Universidad Tecnológica Metropolitana

**2017**

**INTEGRANTES: Luis Armando Quintal Castro Diosemir Isael Nah May**

30/03/2017

Metropolitana



Contenido

[Introducción 1](#_Toc475085560)

[Procedimientos (Scrips) 2](#_Toc475085561)

[Diagrama de Gantt 2](#_Toc475085562)

[Lista de requerimientos 3](#_Toc475085563)

[Funcionales: 3](#_Toc475085564)

[No funcionales: 4](#_Toc475085565)

[Diagrama de caso uso 4](#_Toc475085566)

[Diagrama de Clases 4](#_Toc475085567)

[Cuaderno de registro de Tiempos 5](#_Toc475085568)

[Cuaderno de registro de defectos (logs) 5](#_Toc475085569)

[Estándar tipos de defectos 7](#_Toc475085570)

[Resumen plan de proyecto 8](#_Toc475085571)

[Código fuente según estándar 9](#_Toc475085572)

[Aplicación ejecutable e instrucciones para ejecutar 9](#_Toc475085573)

[Link del repositorio 12](#_Toc475085574)

[Conclusión 12](#_Toc475085575)

# Introducción

Esta investigación refleja todos los procesos para la medición del PSP con un programa elaborado en Android y la importancia de conllevar todas las plantillas de manera correcta al momento de elaborar algún software o programa.

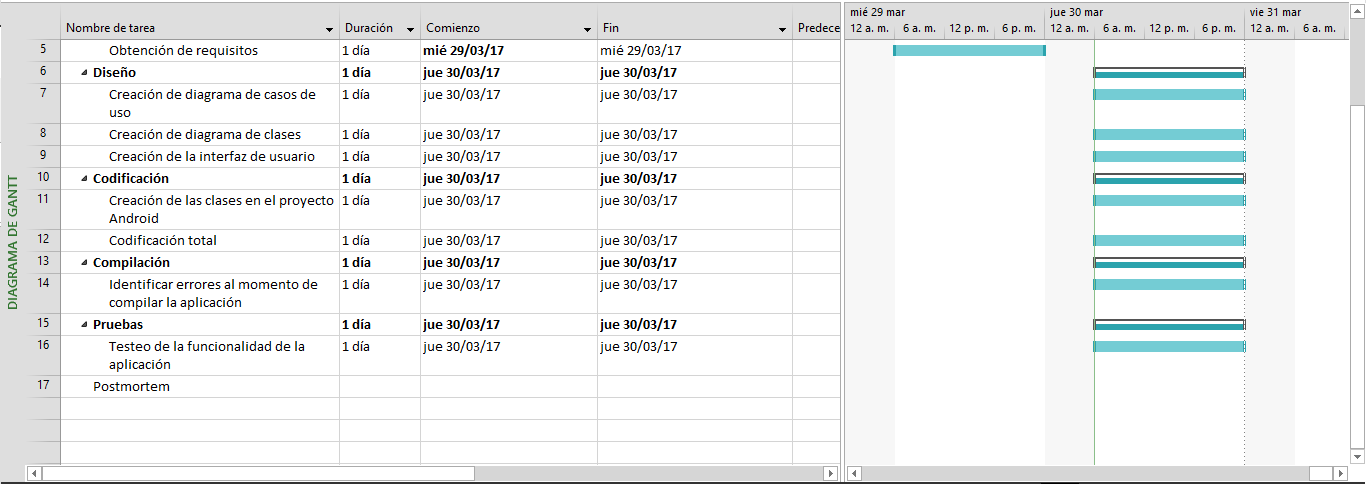
El PSP se caracteriza porque es de uso personal y se aplica a programas pequeños de menos de 10.000 líneas de código. Se centra en la administración del tiempo y en la administración de la calidad a través de la eliminación temprana de defectos. En el PSP se excluyen los siguientes temas: Trabajo en equipo, Administración de configuraciones y Administración de requerimientos, todos para lograr un buen trabajo.

# Procedimientos (Scrips)

En total se utilizaron 3 la cuales fueron

1. Registro de Tiempos
2. Registro de Defectos
3. Resumen Plan del Proyecto

# Diagrama de Gantt



# Lista de requerimientos

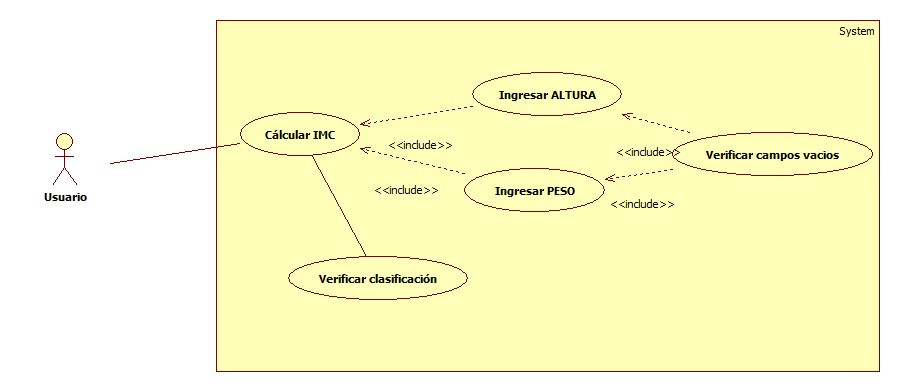
## Funcionales:

* RF001. Obtención del peso y altura por parte del usuario.
* RF002. Validar que los campos de peso y estatura no estén vacíos.
* RF003. Calcular el índice de masa corporal.
* RF004. Clasificación según el índice de masa corporal.

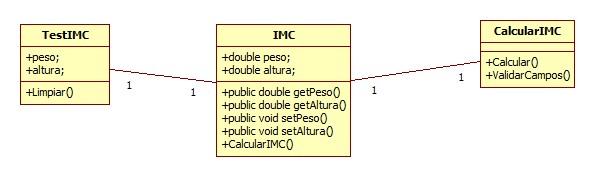
## No funcionales:

* RF002. Validar que se llenen todos los campos para poder calcular la operación.

# Diagrama de caso uso



# Diagrama de Clases



# Cuaderno de registro de Tiempos

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Responsable** | **Fecha** | **Inicio** | **Detención** | **Tiempo de Interrupción** | **Tiempo**  **Delta** | **Fase** | **Comentarios** |
| Luis Armando Quintal Castro | 28/03/17 | 3:00 am | 4:40  am | 20 Minutos | 80 Minutos | Análisis | * Obtención de los requerimientos. * Creación del diagrama de caso de uso. * Creación del diagrama de clases. |
| Luis Armando Quintal Castro | 30/03/17 | 12:37 am | 12:40  am | 0 Minutos | 4 Minutos | Diseño | * Creación de prototipos de la interfaz. * Creación de la interfaz en Android estudio. |
| Luis Armando Quintal Castro | 30/03/17 | 12:40  am | 12:47  am | 0 minutos | 7 minutos | Desarrollo | * Programación de los métodos y las variables en la aplicación. * Creación de las clases |
| Diosemir Isael Nah May | 30/03/17 | 5:51  am | 5:57 am | 0 minutos | 6 minutos | Prueba | * Testeo de los posibles fallos. * Corrección de fallos. |
| Diosemir Isael Nah May  Y  Luis Armando Quintal Castro | 30/03/17 | 5:57 am | 6:10 am | 0 minutos | 13 minutos | Mantenimiento | * Mantenimiento de toda la App y su código fuente. |

# Cuaderno de registro de defectos (logs)

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre | Diosemir Isael Nah May | Fecha | 29 de febrero del 2017 |
| Profesor | M.G.T.I Oscar Uh Pérez | Programa | IndiceMasaCorporal |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Fecha | Nº | | Tipo | | | Introducido | | Eliminado | | | Tiempo de corrección | | |
| 30-Marzo-2017 | 1 | | 70 | | | Datos | | Contenido | | | 19 minutos | | |
| descripción | | Se cambió el contenido de las clases por las que deberían ser para su correcto funcionamiento. | | | | | | | | | | | |
| 30-Marzo-2017 | | 3 | | 60 | | | Chequeo | | Mensajes correctos | | | 4 minutos | |
| descripción | | Se verifico los mensajes que sean correctos para el usuario. | | | | | | | | | | | |
| 30-Marzo-2017 | | 4 | | | 10 | | Documentación | | | Estructura | | | 10 minutos |
| descripción | | Modificación de la estructura del documento | | | | | | | | | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre | Luis Armando Quintal Castro | Fecha | 15 de febrero del 2017 |
| Profesor | M.G.T.I Oscar Uh Pérez | Programa | IndiceMasaCorporal |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Fecha | Nº | | | Tipo | Introducido | | Eliminado | | Tiempo de corrección |
| 30-Marzo-2017 | 5 | | | 70 | Datos | | Estructura Desordenada-Entorno | | 5 minutos |
| descripción | | Se cambió la estructura del programa. | | | | | | | |
| 30-Marzo-2017 | | 6 | 70 | | Datos | Metodos | | 4 minutos | |
| Descripción | | Se verifico el funcionamiento del método. | | | | | | | |

# 

# Estándar tipos de defectos

|  |  |  |
| --- | --- | --- |
| Nº Tipo | Nombre Tipo | Descripción |
| 10 | Documentación | Comentarios, Mensajes |
| 20 | Sintaxis | Ortografía, puntualidad, erratas, formato de instrucciones |
| 30 | Construir | Gestión del cambio ,librerías, control de versiones |
| 40 | Asignación | Declaración, Duplicidad de nombres, ámbito |
| 50 | Interfaz | Llamadas a procedimientos y referencias. E/S |
| 60 | Chequeo | Mensajes de error, chequeos inadecuados |
| 70 | Datos | Estructura, Contenido |
| 80 | Función | Lógica, bucles, recursión… |
| 90 | Sistema | Configuración, memoria, temporización |
| 100 | Entorno | Diseño, Compilación, pruebas |

# Resumen plan de proyecto

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Programa:** Área de un triángulo | | | | | | | **Fecha:** 15/02/2017 |
| **Descripción:** Software encargado de calcular el área de un triángulo realizando múltiples operaciones, tales como validar el triángulo, calcular el área en base a la función de una variable. Está desarrollado en el lenguaje Java en la plataforma Android Studio. | | | | | | | |
| **Resumen** | **Plan** | | **Real** | | | | **Hasta la fecha** |
| Minutos/LOC | 1.6 | | 1 | | | | 2.6 |
| LOC/Hora | 95 | | 60 | | | | 155 |
| Defectos/KLOC | - | | 13 | | | | 13 |
| Rendimiento | - | | - | | | |  |
| Valoración/Fallo | - | | - | | | |  |
| **Tamaño programa (LOC)** | **Plan** | | **Real** | | | | **Hasta la fecha** |
|  | |  | | | |  |
| Total nuevo & cambiado | 100 | | 150 | | | | 130 |
| Tamaño máximo | 300 | |  | | | | |
| Tamaño mínimo | 100 | |
| **Tiempo por Fase (min.)** | **Plan** | **Real** | | **Hasta la fecha** | **% Hasta la fecha** | | |
| Planificación | 50 | 120 | | 100 | 30% | | |
| Diseño | 80 | 90 | | 120 | 50% | | |
| Codificación | 180 | 200 | | 170 | 60% | | |
| **Revisión del código** |  | | | | | | |
| Compilación | 10 | 20 | | 30 | 15% | | |
| Pruebas | 50 | 60 | | 10 | 20% | | |
| Postmorten | 60 | 65 | | 100 | 5% | | |
| Total | 430 | 555 | | 530 |  | | |
| Tiempo máximo |  |  | | | | | |
| Tiempo mínimo |  |
| **Defectos Introducidos** | **Plan** | **Actual** | | **Hasta la fecha** | **% Hasta la fecha** | **Def./Hora** | |
| Planificación | - | 50 | | 50 | 10% | 0.2 | |
| Diseño | - | 10 | | 10 | 15% | 0.1 | |
| Codificación | - | 15 | | 15 | 20% |  | |
| Revisión del código | - | 30 | | 30 | 22% |  | |
| Compilación | - | 20 | | 20 | 25% |  | |
| Pruebas | - | 10 | | 10 | 27% |  | |
| Total | - | 135 | | 135 | 119% |  | |
| **Defectos eliminados** | **Plan** | **Actual** | | **Hasta la fecha** | **% Hasta la fecha** | **Def./Hora** | |
| Planificación | - | 45 | | 45 | 10% |  | |
| Diseño | - | 7 | | 7 | 17% |  | |
| Codificación | - | 12 | | 12 | 21% |  | |
| Revisión del código | - | 25 | | 25 | 23% | 1.5 | |
| Compilación | - | 15 | | 15 | 27% | 1.8 | |
| Pruebas | - | 8 | | 8 | 30% | 2.3 | |
| Total | - | 112 | | 112 | 128 |  | |

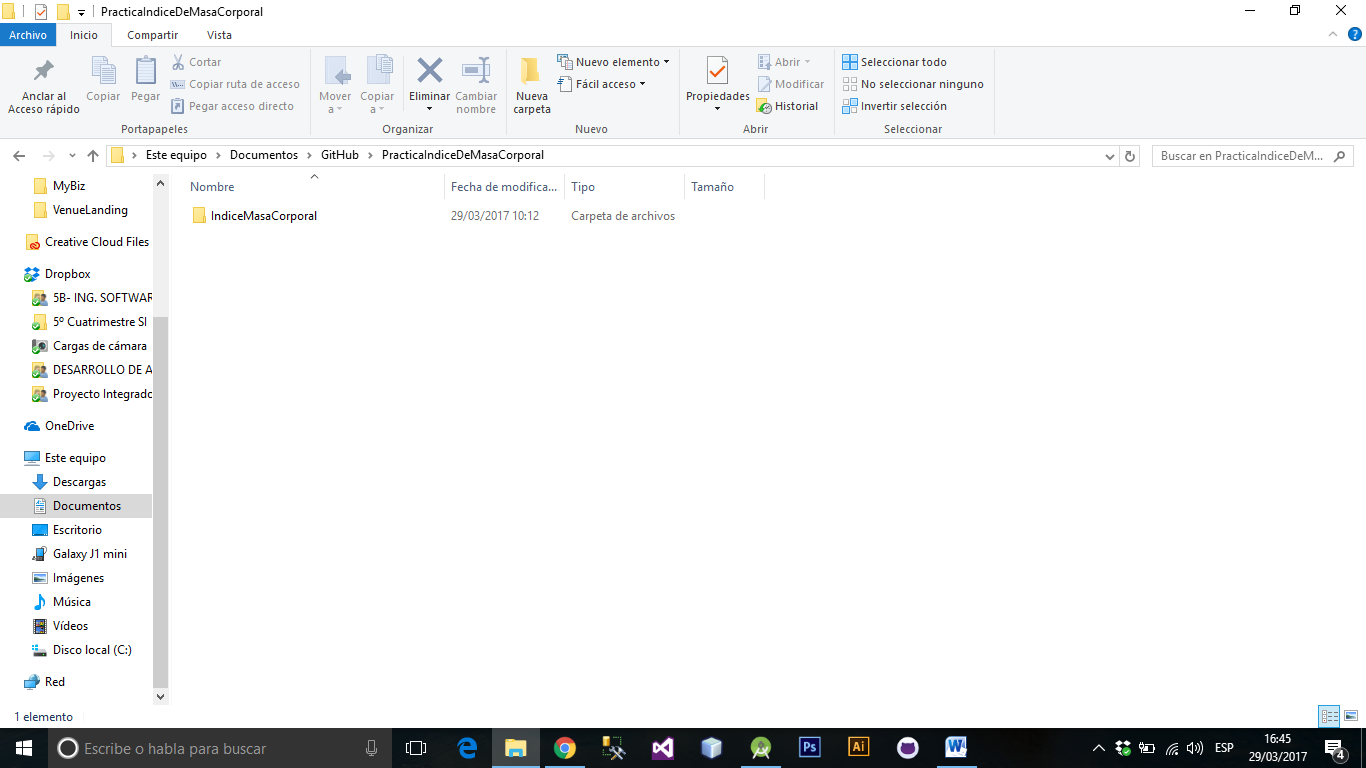
# Código fuente según estándar

|  |  |  |
| --- | --- | --- |
| Nº/Categoría | Descripción del estándar | Ejemplo |
| 1 | Métodos | CalcularIMC() |
| 2 | Comentarios | //En este método |
| 3 | Variables y constantes | EditText peso, altura; |
| 4 | Clases | MainActivity.java |

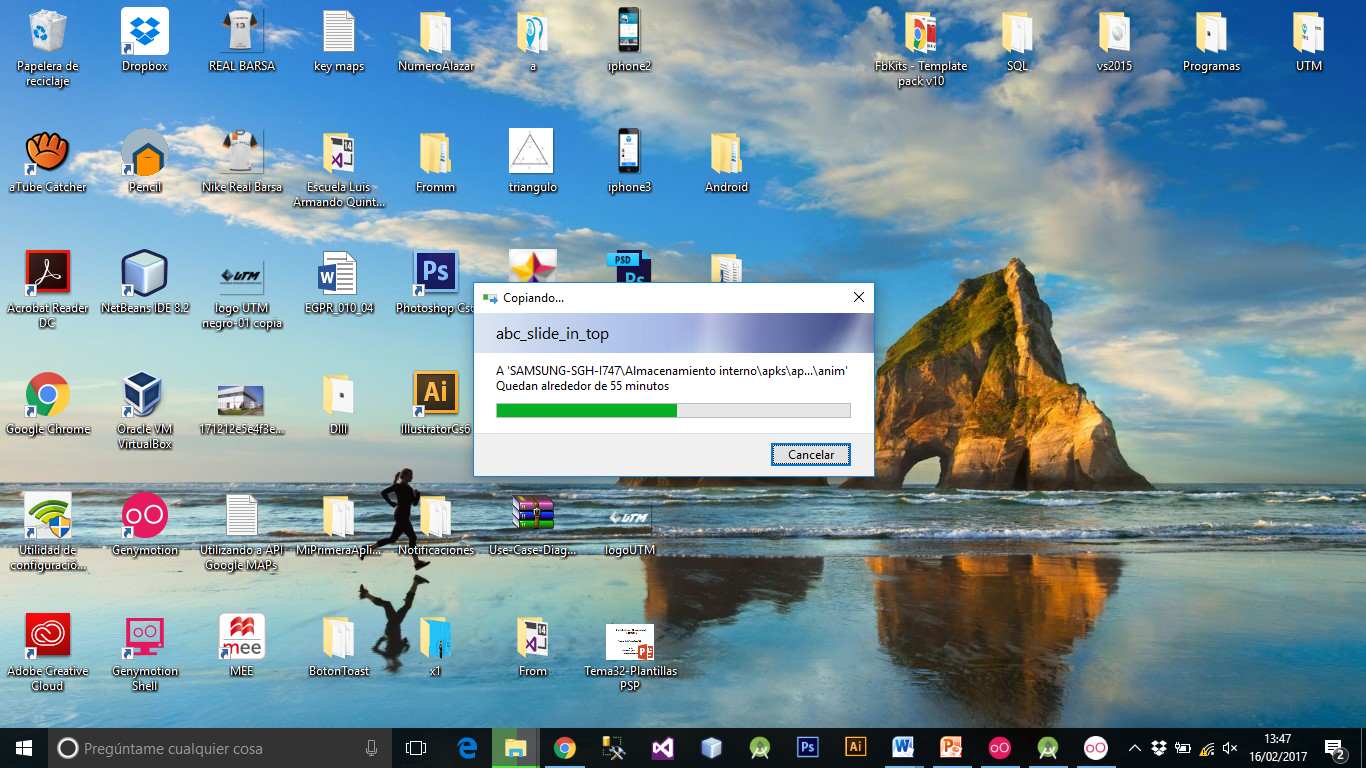
1. Ejemplo Métodos: Public void CalcularIMC(){}
2. Ejemplo Comentarios: EditText peso; // Variable que recibe el peso de la persona
3. Ejemplo Variables y constantes : EditTex altura; constante IMC= 0.0
4. Ejemplo Clases: public class MainActivity extends AppCompatActivity {}

# Aplicación ejecutable e instrucciones para ejecutar

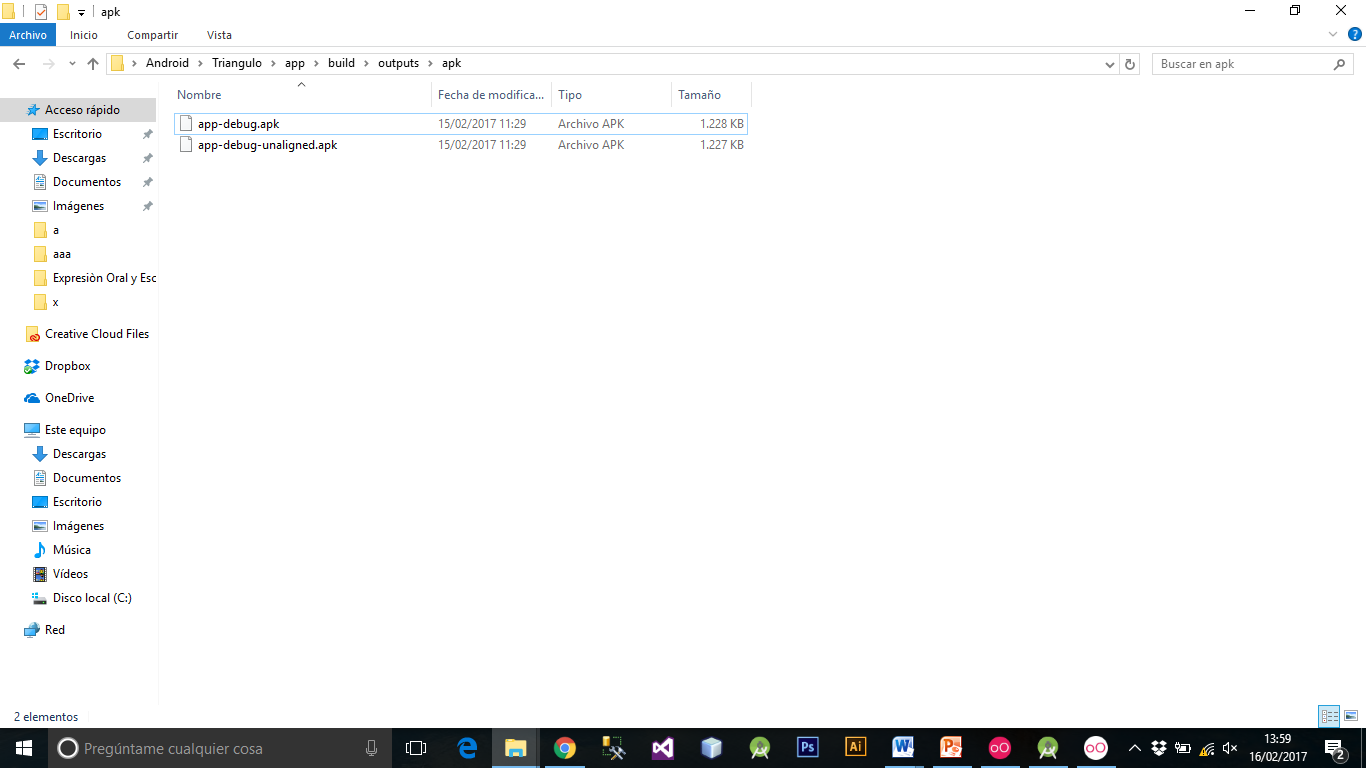
1. Ubicar la ruta donde se generan los proyectos Android



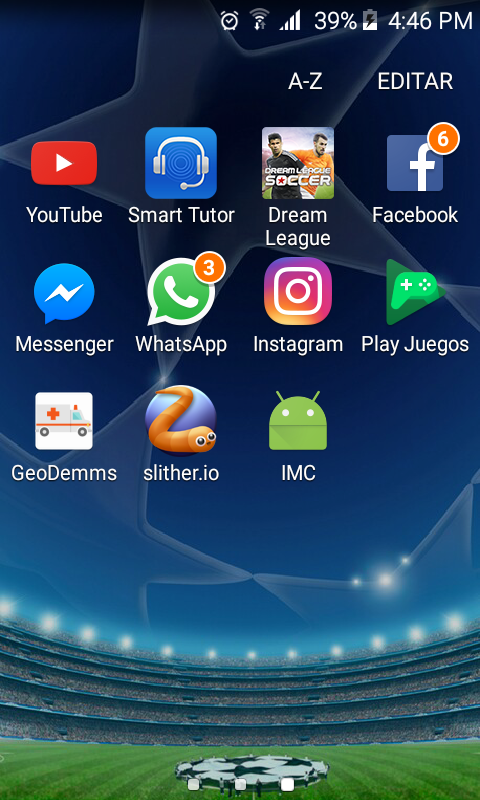
1. Se envía la carpeta “build” en su celular de esta manera.



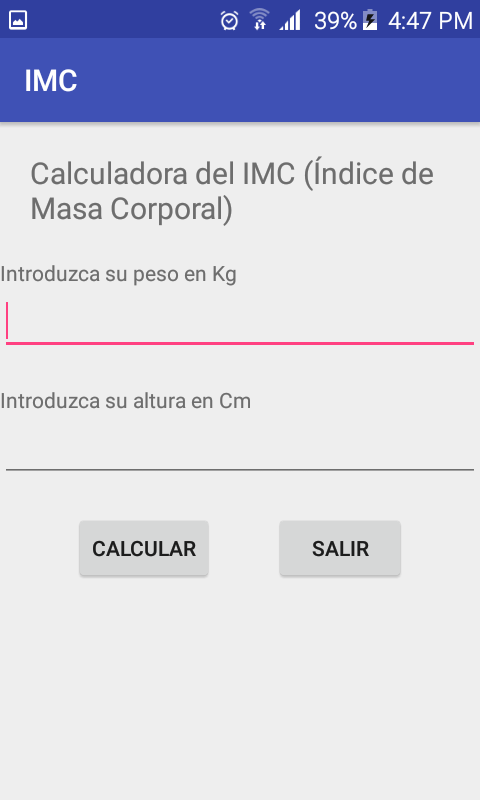
1. Se instala el APK que se encuentra dentro de la carpeta



1. Se instala y te aparecerá el icono de la aplicación con el nombre correspondiente.

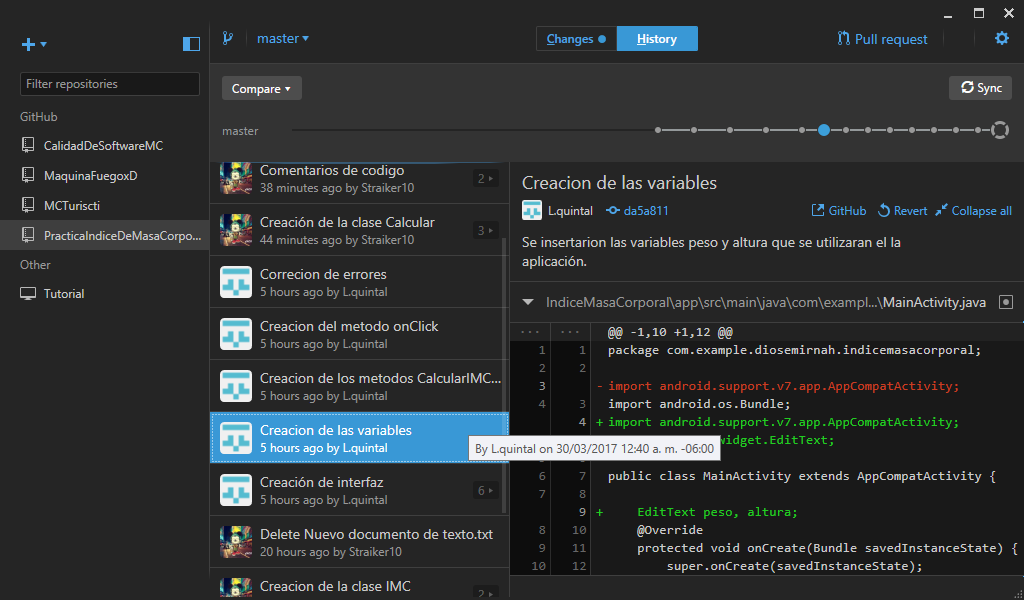


1. Y por último se corre la aplicación.



# Link del repositorio

<https://github.com/Straiker10/PracticaIndiceDeMasaCorporal>



# Conclusión

Podemos concluir que la disciplina en el proceso de desarrollo de software es, sin lugar a dudas, uno de los elementos fundamentales para tal propósito debemos comenzar a entenderla y aplicarla, por lo cual la metodología PSP es parte de un proceso cíclico que ayuda al mejoramiento personal de las actividades. Para el caso particular ayudo al proceso desarrollo de la aplicación móvil implementada, pruebas, planeación y herramientas que ayudan a la gestión de todo esto.

Con PSP, el trabajo que requiere de técnicas superiores va siendo más común día a día. Asimismo exige que el desempeño personal del desarrollador sea cada vez de mayor calidad. (Quintal)

Gracias a esta nueva implementación pude notar el cambio y la nueva restructuración en cuanto al desarrollo de esta actividad, adoptando así un nuevo control y manejo adecuado de las diversas tareas o actividades que se llevan acabó dentro de un proyecto. También es increíble notar las facilidades que un controlador de versiones en este caso Github puede tener hacia nosotros ya que es de gran ayuda su funcionamiento, dando así una gran capacidad de conocimiento en cuanto a que se realiza y que realiza cada quien en un proyecto, de igual manera es increíble ver como proyectos tan pequeños son un poco complicados en cuanto a la repartición de tareas, pero gracias a estos métodos todos realizan las cosas igual.(Diosemir Nah)