

BeatIt!

Estándares de Implementación

Versión 1.1



Historia de revisiones

Fecha	Versión	Descripción	Autor
27/08/2014	1.0	Descripción general de los estándares de implementación.	Luciana Martínez Cristian Bauza
30/08/2014	1.1	Revisión de SQA	Pablo Olivera

Contenido

- [1. Convenciones Generales](#)
- [2. Convenciones Específicas](#)
 - [2.1. Variables](#)
 - [2.2. Constantes](#)

[2.3. Indentación](#)
[3. Otros](#)

1. Convenciones Generales

Los comentarios se deben realizar siguiendo el estándar de JavaDoc para Android. Para el caso de C# y Windows Phone, los comentarios que preceden a una clase o método se deben escribir con la estructura xml adecuada, la cual se genera automáticamente si tipeamos `///`.

Al comienzo de cada clase se debe hacer un pequeño resumen que especifique para que se va a usar la misma.

Se debe describir brevemente cada atributo.

Al comienzo de cada método dar una reseña de que hace, especificando parámetros de entrada y retorno.

En el caso de que el algoritmo de algún método sea complejo, realizar un comentario explicativo de qué es lo que se está haciendo.

Evitar los comentarios obvios.

2. Convenciones Específicas

2.1. Variables

Se debe evitar usar variables de una sola letra (a, b, c.), siempre deben ser lo más descriptivas posibles de lo que representan y siempre deben comenzar con minúscula, en caso de ser más de una palabra, las subsiguientes deben comenzar con mayúscula Si (camelCase), por ejemplo: **velocidadActual**

Es una buena idea utilizar los mismos nombres para variables temporales que cumplen el mismo rol. Por ejemplo:

```
// i para bucles for  
for (i = 0; i < 100; i++)  
{  
  
    // j para bucles for anidados  
    for (j = 0; j < 100; j++)  
    {  
  
        }  
    }  
}
```

De esta forma, se hace más fácil entender cuál es el propósito de una variable a simple vista.

2.2. Constantes

Se deben escribir en mayúsculas y ser lo más descriptivas posibles, por ejemplo: **VELOCIDADMAX**

2.3. Indentación

El código dentro de un bloque anidado siempre debe estar tabulado hacia la derecha un espacio más que su contenedor.

3. Otros

Las llaves de apertura de un bloque siempre deben comenzar en la línea siguiente a la que termina el cabezal, por ejemplo:

```
function saludo(sexo)
{
  if(sexo == 'M')
  {
    alert('Buen día señor');
  }
  else
  {
    alert('Buen día señora');
  }
}
```

Limitar el ancho de las líneas, es más legible un código con columnas angostas que anchas.