



Estándar de Programación .iOS

1. ASPECTOS GENERALES A CONSIDERAR:

- Correcto uso del manejo de la memoria.
- Utilizar componentes nativos de iOS para los mensajes, ya que para el diseño se tendrá que apegar a los prototipos actuales de los StoryBoards, los cuales están muy personalizados para el negocio.
- El modelo del desarrollo tendrá que ser MVC.
- Debe existir legibilidad en el código.
- Cuando se realice más de una afectación en la base de datos, ésta debe quedar transaccionada.
- Cuidar y controlar el manejo de errores, si ocurre un error, informar en pantalla y que éste no afecte en nada la situación actual, por ejemplo, si se iba a guardar algo y falló, no limpiar pantalla o el set de datos para que se pueda intentar guardar de nuevo.
- Todos los componentes utilizados deberán aplicar en la versión actual de iOS, no utilizar componentes que hayan quedado obsoletos en versiones anteriores de iOS. Ejemplo: el componente UIAlertView quedó obsoleto en iOS 9.0 y actualmente estamos en la versión 9.3 de iOS, por ende, no se debería utilizar.

2. ESPECIFICACIONES DE CODIFICACION XCODE-iOS

- Toda variable declarada deberá ser utilizada, no declarar variables si no se va a usar.
- Las variables locales deben ser inicializadas.
- Utilizar métodos de Instancia y de Clase según sea el caso y deberán empezar con Mayúsculas.
- No mezclar idiomas en la codificación y cuidar ortografía.
- Cuidar el uso de los Selectores.
- No dejar código comentado.
- No crear métodos que no se vayan a usar.
- Dado que la mayoría de nombres de variables, métodos y clases se construyen concatenando varias palabras, se recomienda utilizar una mezcla de mayúsculas y minúsculas para simplificar la lectura. Además, la primera letra de cada palabra debe estar en mayúscula. Para las variables, ponga la primera letra de cada palabra en mayúscula, exceptuando la primera.
- No utilizar números o cadenas literales, como por ejemplo: "For i = 1 To 7". En su lugar, se deberán utilizar constantes.
- Agregar sangría al código, ya que ésta hace que sea más fácil de leer. [Formato]
- Utilizar espacios antes y después de los operadores, ejemplo: int = 9.



3. VERSIONAMIENTO DEL CODIGO

- Solo subir código necesario, no subir archivos innecesarios.
- Detalle del proceso de versionamiento:
 1. SuKarne entrega branch a la fábrica.
 2. La fábrica deberá administrar de manera local (en algún servidor de ellos mismos) el código.
 3. La fábrica hace los desarrollos solicitados, ejecuta pruebas unitarias y hace sus correcciones.
 4. Una vez concluido el desarrollo ya incluyas pruebas unitarias, la fábrica deberá entregar a SuKarne un checklist de implementación (scripts, configuraciones) y el ejecutable de la aplicación.
 5. SuKarne monta en ambiente de calidad la aplicación y ejecuta pruebas integrales.
 6. Si se encuentran defectos, SuKarne regresa relación de defectos a la fábrica y el flujo regresa al paso 3.
 7. Cuando ya no existen defectos en la aplicación, SuKarne le solicita a la fábrica versionar todos los cambios realizados.
 8. La fábrica versiona en el repositorio de SuKarne todos los cambios ya sea de código y/o de base de datos. Es importante mencionar que el versionamiento es con subversion (SVN).
 9. De esta manera nos aseguramos que la fábrica entregará a SuKarne una sola revisión de SVN.