**Estándares De Desarrollo BackEnd**  
  
Para mayor detalle sobre los estándares en C# consultar: <https://msdn.microsoft.com/es-es/library/ms229002(v=vs.110).aspx>  
  
**Estándares:**

1. **Nombramiento**
   1. En general las nombres de variables o clases deben ser en español a menos de que sea un nombre técnico sin traducción o el framework requiera que el nombre sea en inglés (ej: Controller).
   2. Si el nombre de una variable o clase contiene la letra ‘ñ’, debe ser reemplazada por ‘ni’ (ej: Cumpleaños -> Cumpleanios) para evitar problemas durante la compilación y ejecución del código.
   3. Entidades: Las clases deben nombrarse siempre con la primera letra en mayúscula [Pascal Case].
   4. Propiedades: Las propiedades de una clase siempre deben empezar con una mayúscula, sin espacios, sin dos letras mayúsculas seguidas y si el nombre contiene más de una palabra, cada palabra debe iniciar con mayúscula [Pascal Case].
   5. Relaciones: Las entidades que sean relaciones que se deban mapear por medio de Entity Framework y que sean propiedad de una clase, deben empezar por la palabra virtual seguida de la entidad y seguido finalmente por el nombre que se le quiera dar, el nombre debe respetar el nombramiento Pascal Case.
   6. Llaves Foráneas: Toda relación debe llevar una llave foránea, por esta razón las llaves foráneas siempre deben ser de tipo GUID y su nombre debe ser la unión del nombre de la relación a la que representan seguido de la palabra Id respetando el nombramiento Pascal Case.
   7. Interfaces: Las interfaces deben empezar con una “i” mayúscula seguida del nombre de la entidad o de la tarea que se vaya implementar seguido de la palabra “Se” para servicios de entidad o “St” para servicios de tarea.
   8. Implementaciones de servicios: Si bien las implementaciones también son clases para distinguirlas agregamos al final la palabra “Se” para servicios de entidad o “St” para servicios de tarea.
   9. Carpetas: Las carpetas dentro de cada proyecto o dentro de la solución se deben nombrar con la primera letra en mayúscula, sin espacios, sin dos mayúsculas seguidas y si tiene más de una palabra cada palabra debe iniciar con la primera letra en mayúscula.
   10. Namespace: Sin importar la ubicación del archivo (clase, interfaz, controlador… etc.), el namespace debe ser igual al nombre del proyecto donde se encuentra ubicado el archivo.
   11. Métodos – Servicios Entidad: Para nombrar los métodos de los servicios de entidad debemos tener en cuenta el tipo de retorno para su nombramiento. Si el tipo de retorno es un arreglo de objetos el nombre debe ser Listar + [Nombre], ejemplo: ListarPorTipoDeCuenta, Si el tipo de retorno es un solo objeto el nombre debe ser Obtener + [Nombre], ejemplo: ObtenerPorCuenta. Para ambos casos siempre se debe respetar el nombramiento Pascal Case. Los otros métodos deberán ser nombrados Crear, Modificar y Borrar.
   12. Clases: En general para todas las clases siempre debemos respetar el nombramiento Pascal Case.  
       1. Map: Las clases que destinaremos para mapear las propiedades de nuestras entidades POCO, deben ser nombrados de la siguiente forma Nombre Entidad + Map, por ejemplo: CuentaMap.
       2. Modelos: El nombramiento de los modelos debe ser Nombre Entidad + Modelo, por ejemplo: CuentaModelo.
       3. Controladores: El nombramiento de los controladores debe ser Nombre Entidad +Controller, por ejemplo: CuentaController.
       4. Validadores: El nombramiento de los validadores debe ser Nombre Entidad + Validador, por ejemplo: CuentaValidador.
   13. Propiedades: Dependiendo de la visibilidad de las propiedades utilizaremos Pascal Case o Underscore. Para las propiedades públicas siempre se deben nombrar bajo Pascal Case y para las propiedades privadas siempre se deben nombrar bajo Underscore.  
       1. Llaves primarias [Id]: Siempre deben ser de tipo Guid.
       2. DbSet: Los DbSet siempre se consideran propiedades de los Contextos, por lo que su nombramiento debe ser el nombre de la entidad en Pascal Case.
       3. Enumeraciones: Las enumeraciones también son consideradas propiedades porque se encuentran dentro de una clase que debe ser instanciada para hacer uso de las mismas. El nombramiento de las enumeraciones debe ser en Pascal Case.
       4. Agregados En Modelos: Al momento de generar el map de una entidad a un modelo necesitamos información que se encuentran en los agregados, por esta razón, en los modelos agregaremos un atributo que se deberá nombrar así: Nombre Agregado + Propiedad Agregado, por ejemplo: La cuenta tiene un agregado que se llama Naturaleza, si queremos acceder al nombre del agregado debemos llamar la propiedad NaturalezaNombre.
       5. Recursos: El nombramiento de los recursos debe mantener la regla de nombramiento de usar Pascal Case.
   14. Regiones: El nombramiento de las regiones que agrupan código debe ser nombrado en Pascal Case.