Padrão de Nomenclatura do Projeto:

* Visão Geral:
  + Este documento visa formalizar os padrões para nomenclatura dentro do projeto. Ele se encontra em conformidade com os padrões adotados pela Microsoft e pela comunidade de desenvolvedores .NET assim como os padrões já definidos que vierem a ser definidos entre os desenvolvedores.
* Premissas:
  + Este documento se baseia na premissa que os desenvolvedores optaram escrever o código na língua inglesa o que deve ser observado ao nomear qualquer estrutura dentro do projeto.
  + Todos os desenvolvedores devem ler este documento e também as referências.
  + Todos os desenvolvedores se comprometem em fiscalizar a corrigir uns aos outros a fim de sempre melhorar o projeto.
  + Este documento trata especificamente da linguagem C# com o framework ASP.NET Core, questões de nomenclatura em outras linguagens que são ou serão usadas no projeto serão tratadas em documento separados.
  + Este documento não trata de questões arquiteturais os padrões para estruturação do projeto, métodos, classes interfaces e outros serão tratados separadamente.
* Orientações:
  + **Faça:** 
    - Use “**PascalCasing” (ComecePalavrasComLetrasMaiusculas)** para nomear classes e métodos das classes. Ex.:
    - use “**camelCasing**” **(comecePalavrasComLetrasMinusculas)** para nomear parâmetros de métodos e variáveis internas.
    - Use o prefixo “is” ao tratar de booleanos. Ex.: isActive
    - Use o prefixo “has” para definir relação de posse. Ex: “hasChildren”
    - prefira nome longos mas inteligíveis a abreviações curtas. Ex.: “isValidToUpdate” a “valid”
    - Prefira a notação “var” com valores definindo o tipo a variáveis locais tipadas. Ex.: var nome = “lucas” ao invés de “string nome=lucas”
    - Use a pluralização para falar de coleções ao invés de definir como parte do nome. Ex: cars ao invés de carList ou carsList.
    - Use substantivos para nomear classes.
    - Use o prefixo “I” para nomear interfaces. Ex.: IAccount ou invés de Account.
    - Organize o namespace de maneira coerente. Ex.: Company.Product.Module.SubModule
    - Utilize {} alinhadas verticalmente.
    - Utilize enums sempre que for tratar de tipos pré definidos e com pouca mutabilidade.
  + **Evite:**
    - Nomear conceitos que não conheça ou não tenha domínio, verifique com os demais integrantes do grupo.
    - Adicionar outras estruturas dentro de um mesmo arquivo, por exemplo, uma classe e interfaces dentro do mesmo arquivo .cs.
    - Criar enums com valores atribuídos, sempre que possível utilize enums que não necessitem de um valor correspondente, salvo casos onde seja necessário um relacionamento a nível de banco, como uma estrutura de status com várias possibilidades.
  + **Não Faça:**
    - Usar capslock para definir constantes ou tipos estáticos.
    - Usar notação “**Hungarian”** ou seja, definir variáveis com prefixos que remetam ao seu tipo. Ex.: “int iContador” ou “string strNome”
    - Usar \_ para nomear variáveis exceto quanto forem de escopo da classe, privadas e readonly ou static.
    - Usar nome reservados da linguagem em variáveis, métodos, classes ou qualquer objeto do projeto. Ex.: var intCarro = 0;
    - Utilizar a nomenclatura var quando o tipo não for aparente ou entendível. Ex.: int var4 = ExampleClass.ResultSoFar();

Referências:

<https://msdn.microsoft.com/pt-br/library/ms229045(v=vs.110).aspx>

<https://msdn.microsoft.com/en-us/library/ff926074.aspx>

<http://www.dofactory.com/reference/csharp-coding-standards>