## C# Coding guidelines

### Variable and method naming

* Use meaningful names for functions and function parameters. Function names should specify the exact action the method takes, and for most cases will start with a verb, for example:  
    
  string GetJSONField(string fieldName, string jsonStr))
* Use specific names for variables; "value", "equals", "data" (etc.) are bad names.
* Don't start variables with o\_, obj\_, m\_ etc.
* Don’t use the private keyword for private functions  
    
  void Initialize()
* Use Upper Camel Case for classes:  
    
  HandController
* Use Lower Camel Case for public and protected class members:   
    
  public float webRequestTimeoutSeconds = 1.0f;
* Use Lower Camel Case with a leading ‘\_’ or without it for private class members:   
    
  float \_secretWebRequestTimeoutSeconds = 1.0f;
* Use Upper Camel Case for function parameters:  
    
  string GetJSONField(string fieldName, string jsonStr)
* Variables that contain acronyms can leave the acronym capitalized for readability:  
    
  public string myXMLTag;
* Use explicit typing for local variables when the type of variable is obvious from the right side of the assignment:  
    
  Dictionary<string, List<int>> mylists = new Dictionary<string,List<int>>();

### Coding syntax

* Use inline braces for all multi-line conditionals or loops:

if(isAdmin == true)

{

ShowAdminUI();

Debug.Log(“Admin mode”);

}

* For single-line loops and conditionals, you have the option to ALWAYS use braces:

foreach (string value in pets)

{

Debug.Log(“I love “ + value);

}

### Order of Keywords

{public/protected/internal/private/protected internal } // access modifiers

new

{abstract/virtual/override} // inheritance modifiers

sealed

static

readonly

extern

unsafe

volatile

async

### Code comments

Avoid commented out code. It’s confusing and takes a lot of mental processing to discover the purpose of that code. We should not accept such code in pull-requests; if we want to revert old code, it’s better to write the code again or use source control.

*Note: please let me know if you have any additions or see a mistake*