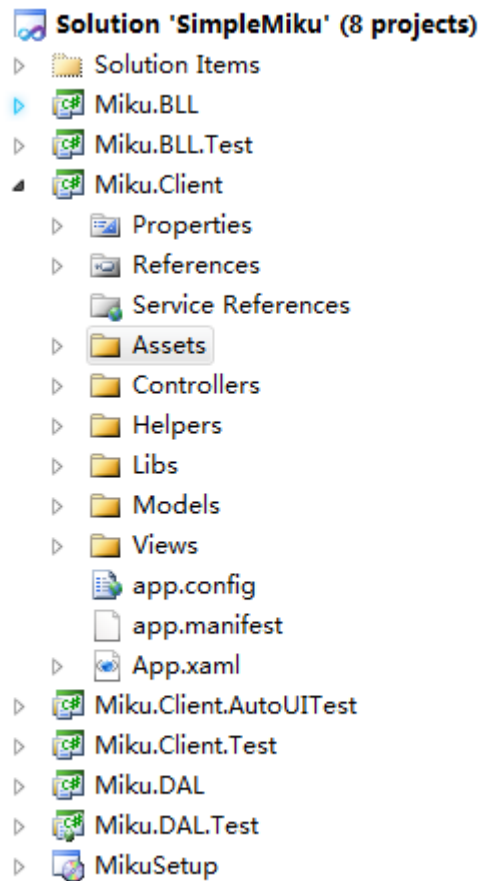
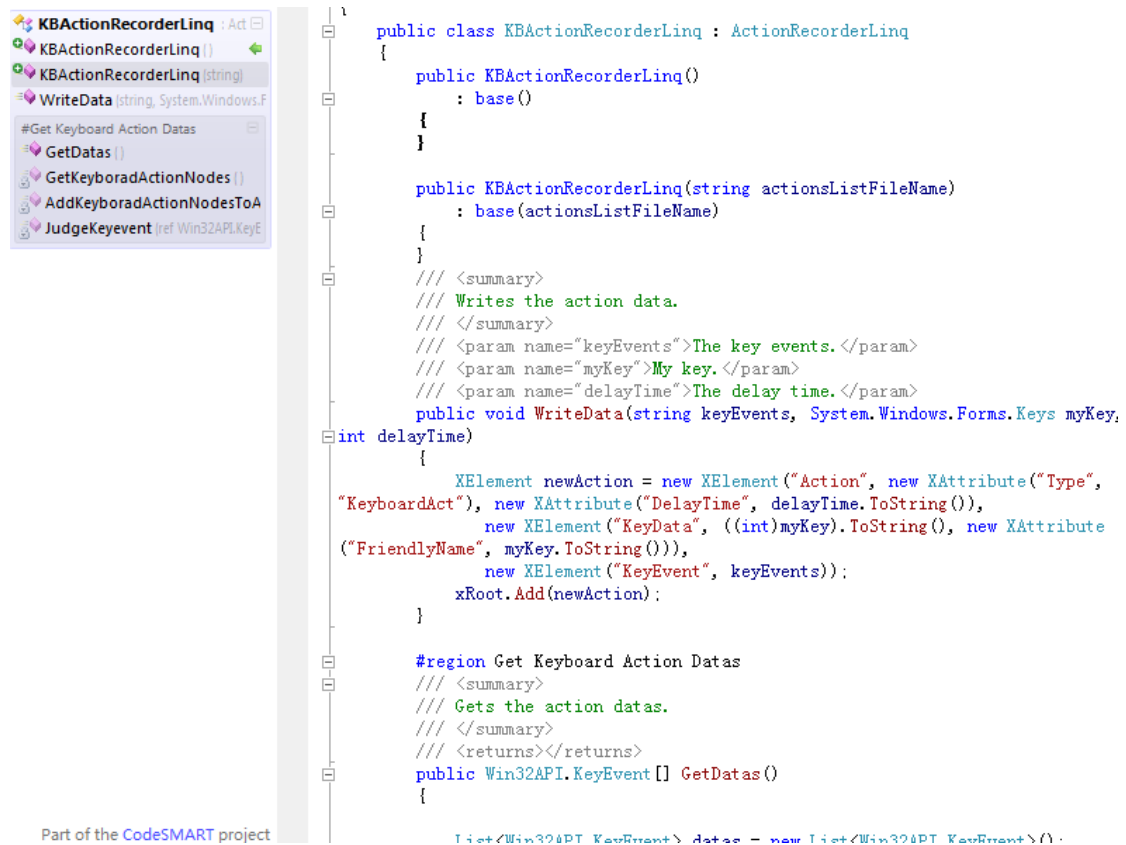


Code Architecture



Code Style



Code System Analysis

	Description	File	Line	Column	Project
62	CA1009 : Microsoft.Design : Declare the first parameter of 'KeyboardEventHandler' as an object named 'sender'.	KBDLLHook.cs	33		Miku.Client
66	CA1009 : Microsoft.Design : Declare the first parameter of 'KeyboardEventHandler' as an object named 'sender'.	KBMLLHook.cs	12		Miku.Client
69	CA1009 : Microsoft.Design : Declare the first parameter of 'MouseEventHandler' as an object named 'sender'.	KBMLLHook.cs	13		Miku.Client
72	CA1009 : Microsoft.Design : Declare the first parameter of 'MouseEventHandler' as an object named 'sender'.	MSLLHook.cs	40		Miku.Client
57	CA1009 : Microsoft.Design : Declare the first parameter of 'OnRecordingEventHandler' as an object named 'sender'.	IActionStrategy.cs	19		Miku.Client
63	CA1009 : Microsoft.Design : Declare the second parameter of 'KeyboardEventHandler' as an EventArgs, or an instance of a type that extends EventArgs, named 'e'.	KBDLLHook.cs	33		Miku.Client
67	CA1009 : Microsoft.Design : Declare the second parameter of 'KeyboardEventHandler' as an EventArgs, or an instance of a type that extends EventArgs, named 'e'.	KBMLLHook.cs	12		Miku.Client
70	CA1009 : Microsoft.Design : Declare the second parameter of 'MouseEventHandler' as an EventArgs, or an instance of a type that extends EventArgs, named 'e'.	KBMLLHook.cs	13		Miku.Client
73	CA1009 : Microsoft.Design : Declare the second parameter of 'MouseEventHandler' as an EventArgs, or an instance of a type that extends EventArgs, named 'e'.	MSLLHook.cs	40		Miku.Client
58	CA1009 : Microsoft.Design : Declare the second parameter of 'OnRecordingEventHandler' as an EventArgs, or an instance of a type that extends EventArgs, named 'e'.	IActionStrategy.cs	19		Miku.Client
64	CA1009 : Microsoft.Design : Remove all but the first two parameters of 'KeyboardEventHandler'.	KBDLLHook.cs	33		Miku.Client
68	CA1009 : Microsoft.Design : Remove all but the first two parameters of 'KeyboardEventHandler'.	KBMLLHook.cs	12		Miku.Client
71	CA1009 : Microsoft.Design : Remove all but the first two parameters of 'MouseEventHandler'.	KBMLLHook.cs	13		Miku.Client
74	CA1009 : Microsoft.Design : Remove all but the first two parameters of 'MouseEventHandler'.	MSLLHook.cs	40		Miku.Client
59	CA1049 : Microsoft.Design : Implement IDisposable on 'Hook'.	Hook.cs	13		Miku.Client
65	CA1049 : Microsoft.Design : Implement IDisposable on 'KBMLLHook'.	KBMLLHook.cs	10		Miku.Client
1	CA1060 : Microsoft.Design : Because it is a P/Invoke method, 'Win32API.BitBlt(IntPtr, int, int, int, IntPtr, int, int, int)' should be defined in a class named NativeMethods, SafeNativeMethods, or UnsafeNativeMethods.	Win32API.cs	219		Miku.Client
3	CA1060 : Microsoft.Design : Because it is a P/Invoke method, 'Win32API.CallNextHookEx(IntPtr, int, int, IntPtr)' should be defined in a class named NativeMethods, SafeNativeMethods, or UnsafeNativeMethods.	Win32API.cs	126		Miku.Client
5	CA1060 : Microsoft.Design : Because it is a P/Invoke method, 'Win32API.CreateDC(string, string, string, IntPtr)' should be defined in a class named NativeMethods, SafeNativeMethods, or UnsafeNativeMethods.	Win32API.cs	232		Miku.Client
10	CA1060 : Microsoft.Design : Because it is a P/Invoke method, 'Win32API.EnumChildWindows(IntPtr, Win32API.CallBack, ref bool)' should be defined in a class named NativeMethods, SafeNativeMethods, or UnsafeNativeMethods.	Win32API.cs	91		Miku.Client
12	CA1060 : Microsoft.Design : Because it is a P/Invoke method, 'Win32API.EnumWindows(Win32API.CallBack, ref bool)' should be defined in a class named NativeMethods, SafeNativeMethods, or UnsafeNativeMethods.	Win32API.cs	86		Miku.Client
14	CA1060 : Microsoft.Design : Because it is a P/Invoke method, 'Win32API.FindWindow(string, string)' should be defined in a class named NativeMethods, SafeNativeMethods, or UnsafeNativeMethods.	Win32API.cs	28		Miku.Client
18	CA1060 : Microsoft.Design : Because it is a P/Invoke method, 'Win32API.FindWindowEx(IntPtr, IntPtr, string, string)' should be defined in a class named NativeMethods, SafeNativeMethods, or UnsafeNativeMethods.	Win32API.cs	42		Miku.Client
22	CA1060 : Microsoft.Design : Because it is a P/Invoke method, 'Win32API.GetClassName(IntPtr, StringBuilder, int)' should be defined in a class named NativeMethods, SafeNativeMethods, or UnsafeNativeMethods.	Win32API.cs	97		Miku.Client
25	CA1060 : Microsoft.Design : Because it is a P/Invoke method, 'Win32API.GetWindowText(IntPtr, StringBuilder, int)' should be defined in a class named NativeMethods, SafeNativeMethods, or UnsafeNativeMethods.	Win32API.cs	94		Miku.Client
28	CA1060 : Microsoft.Design : Because it is a P/Invoke method, 'Win32API.Keyboard_Event(int, int, Win32API.KBEventFlag, int)' should be defined in a class named NativeMethods, SafeNativeMethods, or UnsafeNativeMethods.	Win32API.cs	139		Miku.Client
30	CA1060 : Microsoft.Design : Because it is a P/Invoke method, 'Win32API.Mouse_Event(Win32API.MouseEventFlag, int, int, uint, UIntPtr)' should be defined in a class named NativeMethods, SafeNativeMethods, or UnsafeNativeMethods.	Win32API.cs	158		Miku.Client
32	CA1060 : Microsoft.Design : Because it is a P/Invoke method, 'Win32API.PostMessage(IntPtr, uint, int, int)' should be defined in a class named NativeMethods, SafeNativeMethods, or UnsafeNativeMethods.	Win32API.cs	76		Miku.Client
37	CA1060 : Microsoft.Design : Because it is a P/Invoke method, 'Win32API.RegisterHotKey(IntPtr, uint, uint, uint)' should be defined in a class named NativeMethods, SafeNativeMethods, or UnsafeNativeMethods.	Win32API.cs	211		Miku.Client
39	CA1060 : Microsoft.Design : Because it is a P/Invoke method, 'Win32API.SendMessage(IntPtr, uint, int, int)' should be defined in a class named NativeMethods, SafeNativeMethods, or UnsafeNativeMethods.	Win32API.cs	65		Miku.Client
41	CA1060 : Microsoft.Design : Because it is a P/Invoke method, 'Win32API.SetCursorPos(int, int)' should be defined in a class named NativeMethods, SafeNativeMethods, or UnsafeNativeMethods.	Win32API.cs	204		Miku.Client
43	CA1060 : Microsoft.Design : Because it is a P/Invoke method, 'Win32API.SetWindowsHookEx(Win32API.HookType, Win32API.HookProc, IntPtr, int)' should be defined in a class named NativeMethods, SafeNativeMethods, or UnsafeNativeMethods.	Win32API.cs	123		Miku.Client
45	CA1060 : Microsoft.Design : Because it is a P/Invoke method, 'Win32API.UnhookWindowsHookEx(IntPtr)' should be defined in a class named NativeMethods, SafeNativeMethods, or UnsafeNativeMethods.	Win32API.cs	129		Miku.Client
47	CA1060 : Microsoft.Design : Because it is a P/Invoke method, 'Win32API.UnregisterHotKey(IntPtr, int)' should be defined in a class named NativeMethods, SafeNativeMethods, or UnsafeNativeMethods.	Win32API.cs	214		Miku.Client
49	CA1060 : Microsoft.Design : Because it is a P/Invoke method, 'Win32API.UpdateWindow(IntPtr)' should be defined in a class named NativeMethods, SafeNativeMethods, or UnsafeNativeMethods.	Win32API.cs	50		Miku.Client
2	CA1401 : Microsoft.Interoperability : Change the accessibility of P/Invoke 'Win32API.BitBlt(IntPtr, int, int, int, IntPtr, int, int, int)' so that it is no longer visible from outside its assembly.	Win32API.cs	219		Miku.Client
4	CA1401 : Microsoft.Interoperability : Change the accessibility of P/Invoke 'Win32API.CallNextHookEx(IntPtr, int, int, IntPtr)' so that it is no longer visible from outside its assembly.	Win32API.cs	126		Miku.Client
6	CA1401 : Microsoft.Interoperability : Change the accessibility of P/Invoke 'Win32API.CreateDC(string, string, string, IntPtr)' so that it is no longer visible from outside its assembly.	Win32API.cs	232		Miku.Client
11	CA1401 : Microsoft.Interoperability : Change the accessibility of P/Invoke 'Win32API.EnumChildWindows(IntPtr, Win32API.CallBack, ref bool)' so that it is no longer visible from outside its assembly.	Win32API.cs	91		Miku.Client
13	CA1401 : Microsoft.Interoperability : Change the accessibility of P/Invoke 'Win32API.EnumWindows(Win32API.CallBack, ref bool)' so that it is no longer visible from outside its assembly.	Win32API.cs	86		Miku.Client
15	CA1401 : Microsoft.Interoperability : Change the accessibility of P/Invoke 'Win32API.FindWindow(string, string)' so that it is no longer visible from outside its assembly.	Win32API.cs	28		Miku.Client
19	CA1401 : Microsoft.Interoperability : Change the accessibility of P/Invoke 'Win32API.FindWindowEx(IntPtr, IntPtr, string, string)' so that it is no longer visible from outside its assembly.	Win32API.cs	42		Miku.Client
23	CA1401 : Microsoft.Interoperability : Change the accessibility of P/Invoke 'Win32API.GetClassName(IntPtr, StringBuilder, int)' so that it is no longer visible from outside its assembly.	Win32API.cs	97		Miku.Client
26	CA1401 : Microsoft.Interoperability : Change the accessibility of P/Invoke 'Win32API.GetWindowText(IntPtr, StringBuilder, int)' so that it is no longer visible from outside its assembly.	Win32API.cs	94		Miku.Client
29	CA1401 : Microsoft.Interoperability : Change the accessibility of P/Invoke 'Win32API.Keyboard_Event(int, int, Win32API.KBEventFlag, int)' so that it is no longer visible from outside its assembly.	Win32API.cs	139		Miku.Client
31	CA1401 : Microsoft.Interoperability : Change the accessibility of P/Invoke 'Win32API.Mouse_Event(Win32API.MouseEventFlag, int, int, uint, UIntPtr)' so that it is no longer visible from outside its assembly.	Win32API.cs	158		Miku.Client

36	CA1401 : Microsoft.Interoperability : Change the accessibility of P/Invoke 'Win32API.PostMessage(IntPtr, uint, int, int)' so that it is no longer visible from outside its assembly.	Win32API.cs	76	Miku.Client
38	CA1401 : Microsoft.Interoperability : Change the accessibility of P/Invoke 'Win32API.RegisterHotKey(IntPtr, uint, uint, uint)' so that it is no longer visible from outside its assembly.	Win32API.cs	211	Miku.Client
40	CA1401 : Microsoft.Interoperability : Change the accessibility of P/Invoke 'Win32API.SendMessage(IntPtr, uint, int, int)' so that it is no longer visible from outside its assembly.	Win32API.cs	65	Miku.Client
42	CA1401 : Microsoft.Interoperability : Change the accessibility of P/Invoke 'Win32API.SetCursorPos(int, int)' so that it is no longer visible from outside its assembly.	Win32API.cs	204	Miku.Client
44	CA1401 : Microsoft.Interoperability : Change the accessibility of P/Invoke 'Win32API.SetWindowsHookEx(Win32API.HookType, Win32API.HookProc, IntPtr, int)' so that it is no longer visible from outside its assembly.	Win32API.cs	123	Miku.Client
46	CA1401 : Microsoft.Interoperability : Change the accessibility of P/Invoke 'Win32API.UnhookWindowsHookEx(IntPtr)' so that it is no longer visible from outside its assembly.	Win32API.cs	129	Miku.Client
48	CA1401 : Microsoft.Interoperability : Change the accessibility of P/Invoke 'Win32API.UnregisterHotKey(IntPtr, int)' so that it is no longer visible from outside its assembly.	Win32API.cs	214	Miku.Client
51	CA1401 : Microsoft.Interoperability : Change the accessibility of P/Invoke 'Win32API.UpdateWindow(IntPtr)' so that it is no longer visible from outside its assembly.	Win32API.cs	50	Miku.Client
34	CA1901 : Microsoft.Portability : As it is declared in your code, parameter 'lParam' of P/Invoke 'Win32API.PostMessage(IntPtr, uint, int, int)' will be 4 bytes wide on 64-bit platforms. This is not correct, as the actual native declaration of this API indicates it should be 8 bytes wide on 64-bit platforms. Consult the MSDN Platform SDK documentation for help determining what data type should be used instead of 'int'.	Win32API.cs	76	Miku.Client
33	CA1901 : Microsoft.Portability : As it is declared in your code, parameter 'wParam' of P/Invoke 'Win32API.PostMessage(IntPtr, uint, int, int)' will be 4 bytes wide on 64-bit platforms. This is not correct, as the actual native declaration of this API indicates it should be 8 bytes wide on 64-bit platforms. Consult the MSDN Platform SDK documentation for help determining what data type should be used instead of 'int'.	Win32API.cs	76	Miku.Client
35	CA1901 : Microsoft.Portability : As it is declared in your code, the return type of P/Invoke 'Win32API.PostMessage(IntPtr, uint, int, int)' will be 8 bytes wide on 64-bit platforms. This is not correct, as the actual native declaration of this API indicates it should be 4 bytes wide on 64-bit platforms. Consult the MSDN Platform SDK documentation for help determining what data type should be used instead of 'IntPtr'.	Win32API.cs	76	Miku.Client
50	CA1901 : Microsoft.Portability : As it is declared in your code, the return type of P/Invoke 'Win32API.UpdateWindow(IntPtr)' will be 8 bytes wide on 64-bit platforms. This is not correct, as the actual native declaration of this API indicates it should be 4 bytes wide on 64-bit platforms. Consult the MSDN Platform SDK documentation for help determining what data type should be used instead of 'IntPtr'.	Win32API.cs	50	Miku.Client
75	CA2000 : Microsoft.Reliability : In method 'ActionRecorder.Linq.InitializeRecordedFileFormat()', object 'stream' is not disposed along all exception paths. Call System.IDisposable.Dispose on object 'stream' before all references to it are out of scope.	ActionRecorder.Linq	65	Miku.Client
76	CA2000 : Microsoft.Reliability : In method 'ActionRecorder.Linq.InitializeRootOfActionListTree()', object 'stream' is not disposed along all exception paths. Call System.IDisposable.Dispose on object 'stream' before all references to it are out of scope.	ActionRecorder.Linq	76	Miku.Client
77	CA2000 : Microsoft.Reliability : In method 'ActionRecorder.Linq.SaveActions()', object 'stream' is not disposed along all exception paths. Call System.IDisposable.Dispose on object 'stream' before all references to it are out of scope.	ActionRecorder.Linq	86	Miku.Client
53	CA2000 : Microsoft.Reliability : In method 'ExceptionLogger.LogEvent(string, string, string)', object 'fs' is not disposed along all exception paths. Call System.IDisposable.Dispose on object 'fs' before all references to it are out of scope.	ExceptionLogger.cs	20	Miku.Client
54	CA2000 : Microsoft.Reliability : In method 'ExceptionLogger.LogException(Exception)', object 'fs' is not disposed along all exception paths. Call System.IDisposable.Dispose on object 'fs' before all references to it are out of scope.	ExceptionLogger.cs	38	Miku.Client
55	CA2000 : Microsoft.Reliability : In method 'ExceptionLogger.LogException(Exception)', object 'fs' is not disposed along all exception paths. Call System.IDisposable.Dispose on object 'fs' before all references to it are out of scope.	ExceptionLogger.cs	47	Miku.Client
56	CA2000 : Microsoft.Reliability : In method 'ExceptionLogger.LogException(Exception)', object 'fs' is not disposed along all exception paths. Call System.IDisposable.Dispose on object 'fs' before all references to it are out of scope.	ExceptionLogger.cs	60	Miku.Client
16	CA2101 : Microsoft.Globalization : To reduce security risk, marshal parameter 'lpClassName' as Unicode, by setting DllImport.CharSet to CharSet.Unicode, or by explicitly marshaling the parameter as UnmanagedType.LPWStr. If you need to marshal this string as ANSI or system-dependent, specify MarshalAs explicitly, and set BestFitMapping=false; for added security, also set ThrowOnUnmappableChar=true.	Win32API.cs	28	Miku.Client
24	CA2101 : Microsoft.Globalization : To reduce security risk, marshal parameter 'lpString' as Unicode, by setting DllImport.CharSet to CharSet.Unicode, or by explicitly marshaling the parameter as UnmanagedType.LPWStr. If you need to marshal this string as ANSI or system-dependent, specify MarshalAs explicitly, and set BestFitMapping=false; for added security, also set ThrowOnUnmappableChar=true.	Win32API.cs	97	Miku.Client
27	CA2101 : Microsoft.Globalization : To reduce security risk, marshal parameter 'lpString' as Unicode, by setting DllImport.CharSet to CharSet.Unicode, or by explicitly marshaling the parameter as UnmanagedType.LPWStr. If you need to marshal this string as ANSI or system-dependent, specify MarshalAs explicitly, and set BestFitMapping=false; for added security, also set ThrowOnUnmappableChar=true.	Win32API.cs	94	Miku.Client
20	CA2101 : Microsoft.Globalization : To reduce security risk, marshal parameter 'pszClass' as Unicode, by setting DllImport.CharSet to CharSet.Unicode, or by explicitly marshaling the parameter as UnmanagedType.LPWStr. If you need to marshal this string as ANSI or system-dependent, specify MarshalAs explicitly, and set BestFitMapping=false; for added security, also set ThrowOnUnmappableChar=true.	Win32API.cs	42	Miku.Client
8	CA2101 : Microsoft.Globalization : To reduce security risk, marshal parameter 'pszDevice' as Unicode, by setting DllImport.CharSet to CharSet.Unicode, or by explicitly marshaling the parameter as UnmanagedType.LPWStr. If you need to marshal this string as ANSI or system-dependent, specify MarshalAs explicitly, and set BestFitMapping=false; for added security, also set ThrowOnUnmappableChar=true.	Win32API.cs	232	Miku.Client
7	CA2101 : Microsoft.Globalization : To reduce security risk, marshal parameter 'pszDriver' as Unicode, by setting DllImport.CharSet to CharSet.Unicode, or by explicitly marshaling the parameter as UnmanagedType.LPWStr. If you need to marshal this string as ANSI or system-dependent, specify MarshalAs explicitly, and set BestFitMapping=false; for added security, also set ThrowOnUnmappableChar=true.	Win32API.cs	232	Miku.Client
9	CA2101 : Microsoft.Globalization : To reduce security risk, marshal parameter 'pszOutput' as Unicode, by setting DllImport.CharSet to CharSet.Unicode, or by explicitly marshaling the parameter as UnmanagedType.LPWStr. If you need to marshal this string as ANSI or system-dependent, specify MarshalAs explicitly, and set BestFitMapping=false; for added security, also set ThrowOnUnmappableChar=true.	Win32API.cs	232	Miku.Client
21	CA2101 : Microsoft.Globalization : To reduce security risk, marshal parameter 'pszWindow' as Unicode, by setting DllImport.CharSet to CharSet.Unicode, or by explicitly marshaling the parameter as UnmanagedType.LPWStr. If you need to marshal this string as ANSI or system-dependent, specify MarshalAs explicitly, and set BestFitMapping=false; for added security, also set ThrowOnUnmappableChar=true.	Win32API.cs	42	Miku.Client
17	CA2101 : Microsoft.Globalization : To reduce security risk, marshal parameter 'lpWindowName' as Unicode, by setting DllImport.CharSet to CharSet.Unicode, or by explicitly marshaling the parameter as UnmanagedType.LPWStr. If you need to marshal this string as ANSI or system-dependent, specify MarshalAs explicitly, and set BestFitMapping=false; for added security, also set ThrowOnUnmappableChar=true.	Win32API.cs	28	Miku.Client
60	CA2111 : Microsoft.Security : Make 'Hook.hookedInstance' private.	Hook.cs	15	Miku.Client
61	CA2111 : Microsoft.Security : Make 'Hook.hookHandle' private.	Hook.cs	16	Miku.Client
52	CA2111 : Microsoft.Security : Make 'Win32API.MouseEvent.dwExtraInfo' private.	Win32API.cs	200	Miku.Client