Code Architecture

- Solution 'SimpleMiku' (8 projects)
- Miku.BLL
- Miku.BLL.Test
- Miku.Client
 - ▶ Properties
 - ▶ References
 - References

 - ▶ Controllers
 - ▶ i Helpers

 - ▶ Models
 - ▶ **i** Views
 - app.config
 - app.manifest
- ▶ Miku.Client.AutoUITest
- Miku.DAL
- ▶ Miku.DAL.Test
- MikuSetup

Code Style

```
ধ KBActionRecorderLing : Ad 🖯
                                         public class KBActionRecorderLing : ActionRecorderLing
Q

WBActionRecorderLing()

←
public KBActionRecorderLing()
■ WriteData (string, System.Windows.F
                                                  : base()
#Get Keyboard Action Datas
 GetDatas ()
 public KBActionRecorderLinq(string actionsListFileName)
 AddKeyboradActionNodesToA
                                                  : base(actionsListFileName)
 /// <summary>
                                              /// Writes the action data.
                                              /// </summary>
                                              /// <param name="keyEvents">The key events.</param>
                                              /// <param name="myKeg">My key.</param>
/// <param name="delayTime">The delay time.</param>
                                              public void WriteData(string keyEvents, System. Windows. Forms. Keys myKey,
                                   int delayTime)
                                     XElement newAction = new XElement ("Action", new XAttribute("Type", "KeyboardAct"), new XAttribute("DelayTime", delayTime.ToString()), new XElement("KeyData", ((int)myKey).ToString(), new XAttribute
                                     ("FriendlyName", myKey.ToString())),
                                                     new XElement("KeyEvent", keyEvents));
                                                  xRoot.Add(newAction);
                                              #region Get Keyboard Action Datas
                                              /// <summary>
/// Gets the action datas.
                                              /// </summary>
                                              /// <returns></returns>
                                              public Win32API.KeyEvent[] GetDatas()
   Part of the CodeSMART project
                                                  Tiet/Win390PT KawRrrent's datas - new Tiet/Win390PT KawRrrent's () .
```

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 'Mouse Eventhandler' as an object named 'sender'.	IActionStrategy.cs	19		Miku.Client
		33		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	12		
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			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
	KBMLLHook.cs	10		Miku.Client
65 CA1049 : Microsoft.Design : Implement IDisposable on 'KBMLLHook'.				
1 CA1060: Microsoft.Design: Because it is a P/Invoke method, 'Win32APLBitBlt[IntPtr, int, int, int, int, int, int, int, int	Win32API.cs	219		Miku.Client
3 CA1060: Microsoft.Design: Because it is a P/Invoke method, 'Win32APLCallNextHookEx(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, 'Win32APLEnumWindows(Win32APLCallBack, 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
NativeMethods. SafeNativeMethods or UnsafeNativeMethods.	Win32API.cs	94		Miku.Client
28 CA1060 : Microsoft.Design : Because it is a P/Invoke method, 'Win32API.Keybd_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, 'Win32APLPostMessage(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, 'Win3ZAPLRegisterHotKey(IntPtr, uint, 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(ImPtr, 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(IntPty)' 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 Win32APLBitBlt[IntPtr, int, int, int, int, int, int, int, int	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 'Win32APLCreateDC(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 'Win32APLEnumChildWindows(ImPtr, Win32APLCallBack, 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 'Win32APLEnumWindows(Win32APLCallBack, 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 'Win32APl.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
26 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
Dustine is assembly. 29 CA1401: Microsoft.Interoperability: Change the accessibility of P/Invoke 'Win32API.Keybd_Event(int, int, Win32API.KBEventFlag, int)' so that it is no longer visible from outside its assembly.	Win32API.cs	139		Miku.Client
1 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

	CA1401: Microsoft.Interoperability: Change the accessibility of P/Invoke 'Win32APLPostMessage(ImPtr, uint, int, int)' so that it is no longer visible from outside to assembly.	Win32API.cs	76	Miku.Client
	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
	2A1401 : Microsoft.Interoperability : Change the accessibility of P/Invoke 'Win32API.SendMessage(IntPtr, uint, int, int)' so that it is no longer visible from outside ts 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
	CA1401: Microsoft.Interoperability: Change the accessibility of P/Invoke 'Win32API.SetWindowsHookEx(Win32API.HookType, Win32API.HookProc, IntPtr, int)' so hat it is no longer visible from outside its assembly.	Win32API.cs	123	Miku.Client
	CA1401: Microsoft.Interoperability: Change the accessibility of P/Invoke Win32APL.UnhookWindowsHookEx(IntPtr)' so that it is no longer visible from outside its assembly.	Win32API.cs	129	Miku.Client
	CA1401: Microsoft.Interoperability: Change the accessibility of P/Invoke Win32APL.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
(CA1901 : Microsoft Portability : As it is declared in your code, parameter "Param" of P/Invoke "Win32APLPostMessage(IntPtr, uint, int, int) will be 4 bytes wide on 48-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 latform SK documentation for high determining what data type should be used instead of 'int'.	Win32API.cs	76	Miku.Client
6	ZA901: Microsoft Portability: As it is declared in your code, parameter 'Waram' of P/Invoke 'Win32API,PostMessage(IntPtr, uint, int, int)' will be 4 bytes wide on 44-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 latform SDK documentation for help determining what data type should be used instead of firt.	Win32API.cs	76	Miku.Client
9	ZAI901: Microsoft:Portability: As it is declared in your code, the return type of P/Invoke Win32API.PostMessage(InPtr, uint, int, int) will be 8 bytes wide on 64- it 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 JOK documentation for help determining what data type should be used instead of IntPtr'.	Win32API.cs	76	Miku.Client
F	A1901: MicrosoftPortability: 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 diclicates 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
9	CA2000 : Microsoft.Reliability : In method 'ActionRecorderLinq.InitializeRecordedFileFormat()', object 'stream' is not disposed along all exception paths. Call bystem.IDisposable.Dispose on object 'stream' before all references to it are out of scope.	ActionRecorderLing	65	Miku.Client
9	A2000 : Microsoft.Reliability : In method 'ActionRecorderLinq.InitializeRootOfActionListTree()', object 'stream' is not disposed along all exception paths. Call bystem.IDisposable.Dispose on object 'stream' before all references to it are out of scope.	ActionRecorderLing	76	Miku.Client
9	CA2000 : Microsoft.Reliability : In method 'ActionRecorderLinq.SaveActions()', object 'stream' is not disposed along all exception paths. Call lystem. Disposable.Dispose on object 'stream' before all references to it are out of scope.	ActionRecorderLing	86	Miku.Client
	CA2000: Microsoft.Reliability: In method 'ExceptionLogger.LogEvent(string, string)', object 'fs' is not disposed along all exception paths. Call bystem.IDisposable.Dispose on object 'fs' before all references to it are out of scope.	ExceptionLogger.cs	20	Miku.Client
	A2000 : Microsoft.Reliability : In method 'ExceptionLogger.LogException[Exception]', object 'fs' is not disposed along all exception paths. Call lystem.IDisposable.Dispose on object 'fs' before all references to it are out of scope.	ExceptionLogger.cs	38	Miku.Client
	A2000 : Microsoft.Reliability : In method 'ExceptionLogger.LogException(Exception)', object 'fs' is not disposed along all exception paths. Call bystem.IDisposable.Dispose on object 'fs' before all references to it are out of scope.	ExceptionLogger.cs	47	Miku.Client
	CA2000: Microsoft.Reliability: In method 'ExceptionLogger.LogException(Exception)', object 'fs' is not disposed along all exception paths. Call bystem.IDisposable.Dispose on object 'fs' before all references to it are out of scope.	ExceptionLogger.cs	60	Miku.Client
6	A2101: Microsoft.Globalization: To reduce security risk, marshal parameter "IpClassName" as Unicode, by setting DlItmport.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 BestiftMapping=false; for added security, also set ThrowOnUnmappablicCharstrue.	Win32API.cs	28	Miku.Client
	A2101: Microsoft Globalization: To reduce security risk, marshal parameter "IpString" as Unicode, by setting Dillimport.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 et BestiftMapping=false; for added security, also set ThrowOnUnmappableChar=true.	Win32API.cs	97	Miku.Client
6	A2101: Microsoft Globalization: To reduce security risk, marshal parameter "lpString" as Unicode, by setting Dillimport.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 Marshallas explicitly, and set BestiftMapping=false; for added security, also set ThrowOnUnmappablicChar=true.	Win32API.cs	94	Miku.Client
	A2101: Microsoft:Globalization: To reduce security risk, marshal parameter "ipszClass' as Unicode, by setting DllImport.CharSet to CharSetUnicode, or by explicitly marshaling the parameter as UnmanagedType_LPWStr. If you need to marshal this string as ANSI or system-dependent, specify Marshalkas explicitly, and set SetsfitMapping=false; for added security, also set ThrowOnUnmappableChar=true.	Win32API.cs	42	Miku.Client
•	A2101: Microsoft Globalization: To reduce security risk, marshal parameter "joszDevice" as Unicode, by setting Dillimport.CharSet to CharSet.Unicode, or by explicitly marshalling the parameter as UnmanagedType.LPWStr. If you need to marshal this string as ANSI or system-dependent, specify Marshallas explicitly, and et BestiffMapping=false; for added security, also set ThrowOnUnmappableChar=true.	Win32API.cs	232	Miku.Client
	A2101: Microsoft Globalization: To reduce security risk, marshal parameter "IpszDriver' as Unicode, by setting Dllimport.Charfet to CharSetUnicode, or by explicitly marshaling the parameter as UnmanagedType.LPWStr. If you need to marshal this string as ANSI or system-dependent, specify Marshallas explicitly, and et BestiffMapping=false; for added security, also set ThrowOnUnmappableChartrue.	Win32API.cs	232	Miku.Client
5	A2101: Microsoft Globalization: To reduce security risk, marshal parameter "pszOutput" as Unicode, by setting Dillmport.CharSet to CharSet.Unicode, or by sexplicitly marshalling the parameter as UnmanagedType.LPWStr. If you need to marshal this string as ANSI or system-dependent, specify Marshallas explicitly, and et BestiffMapping=false; for added security, also set ThrowOnUmappableChar-true.	Win32API.cs	232	Miku.Client
9	A2101: Microsoft:Globalization: To reduce security risk, marshal parameter "jaztWindow" as Unicode, by setting OllImport.CharSet to CharSet.Unicode, or by sexplicitly marshalling the parameter as UnmanagedType.LPWStr. If you need to marshal this string as ANSI or system-dependent, specify Marshallas explicitly, and et BestfitMapping=false; for added security, also set ThrowOnUmrappableCharctrue.	Win32API.cs	42	Miku.Client
l l	A2010: Microsoft Globalization: To reduce security risk, marshal parameter "pl\/Microsoft Globalization: To reduce security risk, marshal parameter as Unicode, by setting Dllimport.Charset to CharSet.Unicode, or oy explicitly marshaling the parameter as UnmanagedTypesLPWStr. If you need to marshal this string as ANSI or system-dependent, specify MarshalAs explicitly, and set BestFitMapping=false; for added security, also set ThrowOnlumappableChar=true.	Win32API.cs	28	Miku.Client
60 (CA2111 : Microsoft.Security : Make 'Hook.hookedInstance' private.	Hook.cs	15	Miku.Client
	CA2111 : Microsoft.Security : Make 'Hook.hookHandle' private.	Hook.cs	16	Miku.Client
	CA2111 : Microsoft.Security : Make 'Win32API.MouseEvent.dwExtraInfo' private.	Win32API.cs	200	Miku.Client