SunVot SDK Tool Kit Developer's Guide V1.2

CHANGSHA SUNVOTE LIMITED

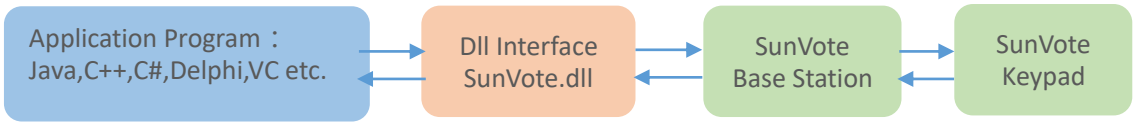
April 2017

Contents

1. Overview

The SunVote wireless response system consists of keypad, base station and main control computer. The base station connects to the computer via USB, Ethernet or WLAN access. Commands are transmitted from the control computer to keypads via base station. SunVote SDK runs on computer and allows the users to develop application software to work with SunVote wireless system. It provides API function calling interface, which has no UI and is applicable to develop application software for self-owned brand users. The interface is simple to realize the management of audience response system, transmission and receipt of voting orders and data.

The call structure is shown as below:



By now, the SunVote SDK can work on Windows7, Windows8, and Windows10.

Base station: SunVote SDK works with SunVote base stations of all series

Keypad: SunVote SDK works with SunVote keypads of all series

SunVote base station connects to the computer via USB, Ethernet or WLAN access.

2. Programming Instruction

Users should follow the basic steps below to develop an application on SunVote SDK:

Step 1: Load SunVote SDK, the specific loading methods depends on programming tool and operation system. See details in sample code below.

Step 2: Initialize BaseConnection object.

Step 3: Set basic connection object of BaseConnection in applications.

Step 4: Call Open method of BaseConnection to connect a specified base station.

Step 5: Call Start method in application objects such as Choice, Judge as necessary.

Step 6: Write the code to process response data in return events of an application object, or perform integrated data processing after logging the response data of events.

Step 7: Call Stop method of application objects.

Step 8: Call Close method of BaseConnection to exit the application program.

Step 9: Call the corresponding method of BaseManage and KeypadManege to get and set the keypad hardware parameters as necessary.

3. BaseConnection

3.1 Usage

It's used to connect a specific base station. The base station may be connected by USB or TCP/IP alternatively and should be connected successfully before performing other application object.

3.2 Attribute

Attribute Name	Type	Parameter Description	Note
DemoMode	bool	=true, enable DemoMode =false, disable DemoMode	Enable DemoMode to simulate polling without SunVote hardware being connected for demonstration or test. It only supports the following two KeyStatus() events randomly generated after calling Start() and simulated key value will be modified. 1) Roll call by a specific key 2) Choice
DemoKeyIDs	string	It's customized to set Keypad ID string for Demomode. Consecutive Keypad IDs can be separated by ","; or "-".	DemoKeyIDs: '1, 3-5' means using Keypads ID 1, 3, 4, 5 for demo.
BaseIP	string	It's customized to set base IP address, which works only for TCP/IP connection. Use "," to separate several bases.	BaseIP: '200.200.100.251, 200.200.100.252' represents two bases connected.
IsWriteErrorLog	bool	=true, to log errors; =false, not to log errors	
ProtocolType	Int	=1 Default ARS protocol. It defaults to be 1 when has no setting =2 CRS protocol =3 EVS protocol =4 General protocol	These 4 protocols mainly differ on roll call, background login and keypad management.
BaseUsedByApp	string	Read only. It is the application which is occupying base connection. " " means the base is available.	

3.3 Method

3.3.1 Open Connection

Method	Open (int Mode, string BaseIDs)
Parameter	Mode: Base connection type =1 connect by USB; =2 connect by TCP/IP; Set base IP first before calling this method.
	BaseIDs The base ID string to be connected. Consecutive keypad IDs can be separated by "," or "-", i.e. '1, 3-5' means connecting base ID 1, 3, 4, 5.
Return Value	No return value. It will generate BaseOnline events.

Sample Code:

```
BaseConnection baseConn = new SunVote.BaseConnection();

baseConn.Open(1,'1-3'); //connect base 1-3 in USB mode

or

//Set base IP to be connected

baseConn.BaseIP = '200.200.100.251,200.200.100.252';

baseConn.Open(2,'1,2');  //connect base 1, 2 in TCP/IP mode
```

3.3.2 Close connection

Method	Close()
Parameter	Null
Return Value	No return value. It will generate **BaseOnline** events.

Sample Code:

```
BaseConnection baseConn = new SunVote.BaseConnection();

baseConnection.Close(); //Close base connection
```

3.4 Event

3.4.1 Base station online

Event	BaseOnline(int BaseID, int BaseState)
Parameter	**BaseID**: Base station ID number
	BaseState: Base connection state

	= 1 Successful = 0 Connection failed or closed = -1 Not support = -2 Base ID to be connected is invalid = -3 Port error. Didn't find a valid port or it's occupied. = -4 Base station is disconnected (auto generated by SDK monitor and reconnect)

Sample Code:

```
baseConn.BaseOnLine += new
SunVote.IBaseConnectionEvents_BaseOnLineEventHandler(baseConn_BaseOnline);

public void baseConn_BaseOnline(int BaseID, int BaseState);
{
    if(BaseState == 1) {
        System.out.println("Connect Base Success");
    }
    else{
        System.out.println("Connect Base Failture");
    }
}
```

Note:

1) **Open** method generally runs when the software started, and for once only if attribute and parameter haven't been modified. SDK shall monitor the base connection state in real time after started, and reconnect automatically if detected a base disconnection.

2) Call **Close** method to log out software.

4. BaseManage

4.1 Usage

This object is used to read and write (get and set) hardware parameters of the connected base station.

4.2 Attribute

Attribute Name	Type	Parameter Description	Note
BaseConnection	Object	\	Initialize the BaseConnection object before setting attribute, then call a method after setting attribute.

4.3 Method

Initialize BaseManage object and set the attribute of BaseConnection before calling a method

Sample Code:

```
BaseManage baseManage = new SunVote.BaseManage();

BaseConnection baseConn = new SunVote.BaseConnection();

baseManage.BaseConnection = baseConn;
```

4.3.1 Read basic features of the voting system

Method	GetBasicFeature(long BaseID)
Parameter	BaseID: Base station ID
Return Value	No return value. It will generate **BasicFeature** events.

Sample Code:

```
baseManage.GetBasicFeature(1); //Get basic fetures of base station 1
```

4.3.2 Set basic features of the voting system

Method	SetBasicFeature(int BaseID, int KeyReportMode, int KeyOffTime, int BackLightMode, int BuzzerMode , int CommitMode)
Parameter	BaseID: Specified base station ID.
	KeyReportMode: Keypad report mode: =0:not report =1:report in standby =2:report in polling =3:report in 1&2
	KeyOffTime: Keypad off time: = customized, to set time from 1-254 min =0:use the default time in keypad firmware = 255:keep keypad on all time
	BackLightMode: Backlight mode =0: off

	=1: on when tapping buttons =2:off after submitting, otherwise backlight will keep on till polling closed =3:on all the time
	BuzzerMode Buzz Mode: =0: off =1: on
	CommitMode: Submit By =0:specific button（i.e. OK button） =1:auto submit (valid for single choice, vote, appraisal and judge only)
Return Value	No return value. It will generate **BasicFeature** events.

Sample Code:

```
// Set the hardware parameters for base 1: not report keypad state, off in 10 minutes, backlight off
// buzz off, submit by a specific key
baseManage.SetBasicFeature(1, 0, 10, 0, 0, 0);
```

4.3.3 Get the base station configurations

Method	GetConfig(int BaseID)
Parameter	**BaseID** Base station ID
Return Value	No return value. It will generate BaseConfig events.

Sample Code:

```
baseManage.GetConfig(1); //Get the configuration of base station 1
```

4.3.4 Set the base station configurations

Method	SetConfig(int BaseID, int BaseNewID, int BaseChannel, int KeyIDMin, int KeyIDMax, int RFPower)
Parameter	**BaseID:**original base ID (0~32). 0 is not to specify a base ID (for single base connection only).
	BaseNewID:new base ID (1~32)
	BaseChannel:base Channel 433M (0~32)、2.4G(1-32)
	KeyIDMin:it's optional. Set the minimum value of keypad ID for this base station. It's used for marking only and convenient to view the keypad scope.
	KeyIDMax:it's optional. Set the maximum value of keypad ID for this base station. It's used for marking only and convenient to view the keypad scope.
	RFPower:RF Grade of base station: =0, default value in base firmware =1, full power =2, medium power

	=3, small power
Return Value	No return value. It will generate BaseConfig events.

Sample Code:

```
// Set new ID 1, channel 7, keypad scope 1-400, default RF value for the connected base
// 0 is used for single base station connection only
baseManage. SetConfig(0, 1, 7, 1, 400, 0);
```

Note:

If set a new base ID, you have to call BaseConnect method again to assure base station works normally.

4.3.5 Read the base station hardware information

Method	GetModelInfo (int BaseID)
Parameter	**BaseID** Base Station ID =0 All connected base stations >0 Base station of the specified ID
Return Value	No return value. It will grenerate BaseModelInfo events.

Sample Code:

```
baseManage.GetModelInfo (1);    //Get hardware information of Base 1
```

4.3.6 Start match

Method	string StartMatch(int BaseID)
Parameter	**BaseID:** base station ID. You need to specify a base ID.
Return Value	String, ='-1' did not set BaseConnection attribute ='0' Start match successfully
	When keypad matched successfully, it will generate MatchStatus events.

Sample Code:

```
baseManage.StartMatch(1);    //Make base 1 to start match
```

Note:

Once the base started match, it cannot start other applications or objects. Only make one base in match state. If the base failed to connect after start match, reconnect the base and system will connect automatically and return to the start status.

4.3.7 Exit match

Method	string StopMatch( )
Parameter	Null
Return Value	String, ='-1' did not set BaseConnection attribute ='0' Stop match successfully

Sample Code:

```
baseManage.StopMatch();    //Stop match
```

Note:

If the base failed to connect after stopping match, reconnect the base, then system will connect automatically and return to the stop status.

4.3.8 Get base station software dongle information

Method	GetSoftDog(int BaseID, string Password)
Parameter	**BaseID:** Base station ID =0 All (all base stations connected) >0 Specify a base ID
	Password: User password (8 bytes, for ASC character only)
Return Value	No return value. It will generate BaseSoftDog events

Sample Code:

```
//Get the softdog of base 1, the base default password is '@SUNVOTE'

baseManage.GetSoftDog (1,"@SUNVOTE");
```

4.3.9 Set base station software dongle information

Method	SetSoftDog(int BaseID,string PassWord, string UserPart1, string UserPart2)
Parameter	**BaseID:** a specified base station ID
	Password: User password (8 bytes, for ASC character only)
	UserPart1: User customized info 1, 16 bytes, for ASC character only
	UserPart2: User customized info 2, 16 bytes, for ASC character only
Return Value	No return value. It will generate BaseSoftDog events.

Sample Code:

```
//Set softdog of base 1. Set correct base password, otherwise unable to set softdog.
```

```
baseManage.SetSoftDog (1,"@SUNVOTE","userInfo1","userInfo2");
```

4.3.10 Start channel evaluation

Used to analyze the base channels to see if there is interference around.

Method	string StartChannelEvaluate (int BaseID)
Parameter	**BaseID** Base station ID =0 All (all base stations connected) >0 Specify a base ID
Return Value	String, = '-1' did not set BaseConnection attribute = '0' Start channel analysis successfully
	It will generate ChannelEvaluate events.

Sample Code:

```
//Base 1 start channel analysis

baseManage.StartChannelEvaluate(1);
```

Note:

It works for wireless base station only. You can't perform any other operations (except stop channel analysis), as the computer needs to keep waiting for analysis result under this mode. You have to call **StopChannelEvaluate()** method to stop.

4.3.11 Exit channel evaluation

Method	String StopChannelEvaluate ( )
Parameter	Null
Return Value	String, = '-1' did not set BaseConnection attribute = '0' Stop channel analysis successfully

Sample Code:

```
baseManage.StopChannelEvaluate();    //Stop channel analysis
```

4.3.12 Get base station IP address

Method	GetIPAddress (int BaseID)
Parameter	**BaseID:** Base station ID =0 All (all base stations connected) >0 Specify a base ID

Return Value	No return value. It will generate BaseIPAddress events.

Sample Code:

```
baseManage.GetIPAddress(1);    //Get IP address of base 1
```

4.3.13 Set base station IP address

Method	SetIPAddress (int BaseID, string Address)
Parameter	**BaseID:** Specify a base ID
	Address : IP address (i.e. 200.200.100.251). Base IP address must be unique.
Return Value	No return value. It will generate BaseIPAddress events.

Sample Code:

```
//Set the base 1 IP address to 200.200.100.251

baseManage.SetIPAddress (1,"200.200.100.251");
```

4.3.14 Get base station MAC address

Method	GetMACAddress (int BaseID)
Parameter	**BaseID:** Base station ID =0 All (all base stations connected) >0 Specify a base ID
Return Value	No return value. It will generate BaseMACAddress events.

Sample Code:

```
baseManage.GetMACAddress (1); //Get the base 1 MAC address
```

4.3.15 Set base station MAC address

Method	SetMACAddress (long BaseID, string Address)
Parameter	**BaseID:** Specify a base ID
	Address: MAC address (i.e. 1E-30-6E-A2-45-02). Base MAC address must be unique.
Return Value	No return value. It will generate BaseMACAddress events.

Sample Code:

```
//Set the base 1 MAC address to 1E-30-6E-A2-45-02

baseManage.SetMACAddress(1,"1E-30-6E-A2-45-02");
```

4.3.16 Get base station additional configuration

Method	GetAddConfig (int BaseID)
Parameter	**BaseID:** Base station ID =0 All (all base stations connected) >0 Specify a base ID
Return Value	No return value. It will generate BaseAddConfig events.

Sample Code:

```
baseManage.GetMACAddress(1);    //Get the other configuration of base 1
```

4.3.17 Set base station additional configuration

Method	SetAddConfig (int BaseID, int MatchMode, string BaseName)
Parameter	**BaseID:** Specify a base ID
	MatchMode and identification mode(ID/SN): =1 Match Mode ID =2 Free Mode ID =3 Auto Mode ID =4 Match Mode SN =5 Free Mode SN =6 Auto Mode SN
	BaseName: Base station name, 12 bytes, for ASC character only.
Return Value	No return value. It will generate BaseAddConfig event.

Sample Code:

```
//Set base stion 1 to match mode ID, base name as MyBase

baseManage.SetMACAddress (1, 1, "MyBase");
```

4.3.18 Get mobile base station enabled information

Method	GetEnabledMobileBase (int BaseID)
Parameter	**BaseID:** Base station ID =0 All (all base stations connected) >0 Specify a base ID
Return Value	No return value. It will generate BaseEnabledMobileBase events.

Sample Code:

```
baseManage.GetEnabledMobileBase (1);//Get to see if base 1 enabled mobile base or not
```

4.3.19 Set mobile base station enabled information

Method	SetEnabledMobileBase (int BaseID, bool Enabled)
Parameter	**BaseID:** Specify a base ID
	Enabled: To enable mobile base for transmission or not
Return Value	No return value. It will generate BaseEnabledMobileBase events.

Sample Code:

```
//Set base 1 to enable mobile base for transmission

baseManage.SetEnabledMobileBase (1, true);
```

4.3.20 Get base station subnet mask

Method	GetSubnetMask (int BaseID)
Parameter	**BaseID:** Base station ID =0 All (all base stations connected) >0 Specify a base ID
Return Value	No return value. It will generate BaseSubnetMask events.

Sample Code:

```
baseManage.GetSubnetMask(1);    //Get the subnet mask of base 1
```

4.3.21 Set base stations subnet mask

Method	SetSubnetMask (int BaseID, string Mask)
Parameter	**BaseID:** Specify a base ID
	Mask: Subnet mask, i.e. "255.255.255.0"
Return Value	No return value. It will generate BaseSubnetMask events.

Sample Code:

```
//Set subnect mask of base 1to '255.255.255.0'

baseManage.SetEnabledMobileBase (1, "255.255.255.0");
```

4.3.22 Get base station gateway address

Method	GetGatewayAddress (int BaseID)
Parameter	**BaseID:** Base station ID =0 All (all base stations connected)

	>0 Specify a base ID
Return Value	No return value. It will generateBaseGatewayAddress events.

Sample Code:

```
baseManage.GetGatewayAddress(1); //Get gateway address of base 1
```

4.3.23 Set base station gateway address

Method	SetGatewayAddress (int BaseID, string Address)
Parameter	**BaseID:** Specify a base ID
	Address: Base gateway address, i.e. '200.200.100.1'
Return Value	No return value. It will generate BaseGatewayAddress events.

Sample Code:

```
//Set gateway address of base 1 to'200.200.100.1'

baseManage.SetGatewayAddress(1,"200.200.100.1");
```

4.3.24 Get base station channels

Method	GetMultichannel (int BaseID)
Parameter	**BaseID:** Base station ID =0 All (all base stations connected) >0 Specify a base ID
Return Value	No return value. It will generate BaseMultichannel events.

Sample Code:

```
baseManage.GetMultichannel (1); //Get the multiple channel info of base 1
```

4.3.25 Set base station channels

Method	SetMultichannel(int BaseID, bool EnabledRelay, bool EnabledKeypadApplyRelay, int ChannelNumber, int Channel1, int Channel2, int Channel3, int Channel4)
Parameter	**BaseID:** Specify a base ID
	EnabledRelay =True, enable mobile base. =False, disable mobile base
	EnabledKeypadApplyRelay =True, allow keypad to apply mobile base. =False, not allow keypad to apply mobile base.
	ChannelNumber: Number of channels is up to 4
	Channel1: channel number is 1.

	Channel2: channel number is 2. **Channel3:** channel number is 3. **Channel4:** channel number is 4.
Return Value	No return value. It will generate BaseMultichannel events.

Sample Code:

```
// Set base 1 to enable mobile base, allow keypad to apply mobile base, and 4 channels,

// channel 1 is 1, channel 2 is 3, channel 3 is 5 and channel 4 is 7

baseManage.SetMultichannel (1, true, true, 4, 1, 3, 5, 7);
```

4.3.26 Get keypad language

Method	GetKeyLanguage(int BaseID)
Parameter	**BaseID:** Base station ID =0 All (all base stations connected) >0 Specify a base ID
Return Value	No return value. It will generate KeyLanguage events.

Sample Code:

```
baseManage.GetKeyLanguage(1); //Get the keypad language of base 1
```

4.3.27 Set keypad language

Method	SetKeyLanguage(int BaseID, int Language)
Parameter	**BaseID:** Specify a base ID
	Language: Language Type = 0 User language 用户指定 在键盘上设定 = 1 Chinese = 2 English = 3 Customized language 1
Return Value	No return value. It will generate KeyLanguage events.

Sample Code:

```
//Set keypad language of base 1 to Chinese

baseManage.SetKeyLanguage(1, 1);
```

4.3.28 Get keypad vibration information

Method	GetKeyVibration(int BaseID)
Parameter	**BaseID:** Base station ID =0 All (all base stations connected) >0 Specify a base ID
Return Value	No return value. It will generate KeyVibration events.

Sample Code:

```
baseManage.GetKeyVibration(1);    //Get the keypad vibration mode of base 1
```

4.3.29 Set keypad vibration information

Method	SetKeyVibration(int BaseID, int Vibration)
Parameter	**BaseID:** specify a base ID
	Vibration: vibration mode =0 Off =1 On
Return Value	No return value. It will generate KeyVibration events.

Sample Code:

```
//Set the keypad vibration of base 1 off

baseManage.SetEnabledMobileBase (1, 0);
```

4.3.30 Get base station check code

Method	GetCheckCodeFirst(int BaseID)
Parameter	**BaseID:** Base station ID =0 All (all base stations connected) >0 Specify a base ID
Return Value	No return value. It will generate CheckCodeFirst events.

Sample Code:

```
baseManage.GetCheckCodeFirst(1); //Get the verification mode of base 1
```

4.3.31 Set base station check code

Method	SetCheckCodeFirst(int BaseID, int CheckMode)
Parameter	**BaseID:** specify a base ID
	CheckMode Verification Mode

	=0 Off =1 Last 4 digits =2 Last 6 digits
Return Value	No return value. It will generate CheckCodeFirst events.

Sample Code:

```
//Set the verification mode of base 1 off

baseManage.SetCheckCodeFirst(1, 0);
```

4.3.32 Get hiding keypad ID information

Method	GetKeyIDHide(int BaseID)
Parameter	**BaseID:** Base station ID =0 All (all base stations connected) >0 Specify a base ID
Return Value	No return value. It will generate KeyIDHide events.

Sample Code:

```
baseManage.GetKeyIDHide(1); //Get base 1 display mode is to show keypad ID or not
```

4.3.33 Set hiding keypad ID information

Method	SetKeyIDHide(int BaseID, int KeyIDHide)
Parameter	**BaseID:** specify a base ID
	KeyIDHide: Show keypad ID or not = 0 Show = 1 Hide
Return Value	No return value. It will generate KeyIDHide events.

Sample Code:

```
//Set base 1 to hide keypad ID

baseManage.SetKeyIDHide(1, 1);
```

4.3.34 Get keypad showing shares

Method	GetKeySharesShow(int BaseID)
Parameter	**BaseID:** Base station ID =0 All (all base stations connected) >0 Specify a base ID

Return Value	No return value. It will generate KeySharesShow events.

Sample Code:

```
baseManage.GetKeySharesShow(1); //Get the base 1 to show keypad shares
```

4.3.35 Set keypad showing shares

Method	SetKeySharesShow(int BaseID, int KeySharesShow)
Parameter	**BaseID:** specify a base ID
	KeySharesShow: Show keypad shares or not = 0 Hide = 1 Show
Return Value	No return value. It will generate KeySharesShow events.

Sample Code:

```
//Set base 1 to show keypad shares

baseManage.SetKeySharesShow(1, 1);
```

4.3.36 Change base station channel automatically

Method	AutoChangeChannel (int BaseID)
Parameter	BaseID: Base station ID =0 All (all base stations connected) >0 Specify a base ID
Return Value	No return value. It will generate ChannelAutoChange event.

Sample Code:

```
baseManage.AutoChangeChannel(1);    // Base 1 changes channel automatically
```

4.4 Event

4.4.1 Base station configuration

Event	BaseConfig(int BaseID,int BaseChannel, int KeyIDMin, int KeyIDMax, int RFPower)
Parameter	**BaseID:** Base station ID
	BaseChannel: Base station channel
	KeyIDMin: Minimum value keypad ID
	KeyIDMax: Maximum value of keypad ID
	RFPower: Base RF grade: =0 Default value in firmware =1 Full power

	=2 Medium power =3 Small power

Sample Code:

```
baseManage.BaseConfig += new
SunVote.IBaseManageEvents_BaseConfigEventHandler(baseManage_BaseConfig);

public void baseManage_BaseConfig(int BaseID, int BaseChannel, int KeyIDMin, int KeyIDMax, int
    RFPower)

{

    string str = string.Format("BaseID:{0}, BaseChannel:{1}, KeyIDMin:{2}, KeyIDMax:{3},
RFPower:{4}",BaseID, BaseChannel, KeyIDMin, KeyIDMax, RFPower);

    MessageBox.Show(str);

}
```

4.4.2 Base station hardware information

Event	BaseModelInfo(int BaseID, int HModel, int HVer, int SVer, string HSerial)
Parameter	**BaseID:** base station ID
	HModel: hardware Model Type
	HVer: hardware Version
	SVer: firmware Version
	HSerial: hardware Serial No.

Sample Code:

```
baseManage.BaseModelInfo += new
IBaseManageEvents_BaseModelInfoEventHandler(baseManage_BaseModelInfo);

public void baseManage_BaseModelInfo(int BaseID, int HModel, int HVer, int SVer, string HSerial)

{

    string str = string.Format("BaseID:{0}, HModel:{1}, HVer:{2}, SVer:{3}, HSerial:{4}", BaseID,
HModel, HVer, SVer, HSerial);

    MessageBox.Show(str);

}
```

4.4.3 Match status

Event	MatchStatus(int KeyID, int HModel, int HVer, int SVer, string HSerial)
	BaseID: base station ID

Parameter	**HModel**: hardware Model Type
	HVer: hardware Version
	SVer: firmware Version
	HSerial: hardware Serial No.

Sample Code:

```
baseManage. MatchStatus+= new
IBaseManageEvents_MatchStatusEventHandler(baseManage_MatchStatus);

public void baseManage_MatchStatus(int KeyID, int HModel, int HVer, int SVer, string HSerial)

{

    string str = string.Format("KeyID:{0}, HModel:{1}, HVer:{2}, SVer:{3}, HSerial:{4}", KeyID,
HModel, HVer, SVer, HSerial);

    MessageBox.Show(str);

}
```

4.4.4 Base station software dongle

Event	BaseSoftDog (int BaseID, string UserPart1, string UserPart2)
Parameter	**BaseID:** base station ID
	UserPart1: User customized info 1
	UserPart2: User customized info 2

Sample Code:

```
baseManage.BaseSoftDog += new
IBaseManageEvents_BaseSoftDogEventHandler(baseManage_BaseSoftDog);

public void baseManage_BaseSoftDog(int BaseID,string UserPart1, string UserPart2)

{

    string str = string.Format("BaseID:{0}, UserPart1:{1},    UserPart2:{2}", BaseID, UserPart1,
UserPart2);

    MessageBox.Show(str);

}
```

4.4.5 Base station channel evaluation

Event	ChannelEvaluate (int BaseID, int ChannelNo, int RssiMax, int RssiAvg)
Parameter	**BaseID:** base station ID
	ChannelNo: channel No.

	RssiMax: max interference value of received signal strength indication (-dBm), smaller value represents stronger interference
	RssiAvg: average interference value of received signal strength indication (-dBm), smaller value represents stronger interference

Sample Code:

```
baseManage.ChannelEvaluate += new
IBaseManageEvents_ChannelEvaluateEventHandler(baseManage_ChannelEvaluate);

public void baseManage_ChannelEvaluate(int BaseID, int ChannelNo, int RssiMax, int RssiAvg)

{

    string str = string.Format("BaseID:{0}, ChannelNo:{1},    RssiMax:{2}, RssiAvg:{3}", BaseID,
ChannelNo, RssiMax, RssiAvg);

    MessageBox.Show(str);

}
```

4.4.6 Basic features of the voting system

Event	BasicFeature(int BaseID, int KeyReportMode, int KeyOffTime, int BackLightMode, int BuzzerMode, int CommitMode)
Parameter	**BaseID:** base station ID
	KeyReportMode: keypad report mode: =0:Not report =1:Report in standby =2:Report in polling =3:Report in 1&2
	KeyOffTime: keypad off time = customized, to set time from 1-254 minutes. =0: default time in keypad firmware = 255: keep keypad on all time
	BackLightMode: Backlight Mode =0: Off =1: On when tapping buttons =2: Off after submitting, otherwise backlight will keep on till polling closed =3: On
	BuzzerMode: buzz mode =0: Off =1: On
	CommitMode: submit by =0: Specific button（i.e. OK button） =1: auto submit (valid for single choice, vote, appraisal and jude only)

Sample Code:

```
baseManage.BasicFeature += new
IBaseManageEvents_BasicFeatureEventHandler(baseManage_BasicFeature);

public void baseManage_BasicFeature(int BaseID, int ChannelNo, int RssiMax, int RssiAvg, int BaseID,
int ChannelNo)

{

    string str = string.Format("BaseID:{0}, KeyReportMode:{1},    KeyOffTime:{2},
BackLightMode:{3},    BuzzerMode:{4}, CommitMode:{5}", BaseID, KeyReportMode, KeyOffTime,
BackLightMode, BuzzerMode, CommitMode);

    MessageBox.Show(str);

}
```

4.4.7 Basic station IP address

Event	BaseIPAddress (int BaseID, string Address)
Parameter	**BaseID:** base station ID
	Address: Base IP address

Sample Code:

```
baseManage.BaseIPAddress += new
IBaseManageEvents_BaseIPAddressEventHandler(baseManage_BaseIPAddress)

public void baseManage_BaseIPAddress (int BaseID, string Address)

{

    string str = string.Format("BaseID:%0, Address:{1}", BaseID, Address);

    MessageBox.Show(str);

}
```

4.4.8 Basic station MAC address

Event	BaseMACAddress (int BaseID:Long, string Address)
Parameter	**BaseID:** base station ID
	Address: base MAC address

Sample Code:

```
baseManage.BaseMACAddress += new
IBaseManageEvents_BaseMACAddressEventHandler(baseManage_BaseMACAddress)

public void baseManage_BaseMACAddress(int BaseID, string Address)
```

```
{
    string str = string.Format("BaseID:{0}, Address:{1}", BaseID, Address);
    MessageBox.Show(str);
}
```

4.4.9 Basic station additional configuration

Event	BaseAddConfig(int BaseID, int MatchMode, string BaseName)
Parameter	**BaseID**: base station ID
	MatchMode and identification mode (ID/SN) =1 Match Mode ID =2 Free Mode ID =3 Auto Mode ID =4 Math Mode SN =5 Free Mode SN =6 Auto Mode SN
	BaseName: base station name

Sample Code:

```
baseManage.BaseAddConfig += new
IBaseManageEvents_BaseAddConfigEventHandler(baseManage_BaseAddConfig)

public void baseManage_BaseAddConfig(int BaseID, int MatchMode, string BaseName)
{
    string str = string.Format("BaseID:%0, MatchMode:{1}, BaseName:{2}", BaseID, MatchMode,
BaseName);
    MessageBox.Show(str);
}
```

4.4.10 Enable mobile base station

Event	BaseEnabledMobileBase (int BaseID, bool Enabled)
Parameter	**BaseID**: base station ID
	Enabled: enable a mobile base for transmission or not

Sample Code:

```
baseManage.BaseEnabledMobileBase += new
IBaseManageEvents_BaseEnabledMobileBaseEventHandler(baseManage_BaseEnabledMobileBase)

public void baseManage_BaseEnabledMobileBase (int BaseID, bool Enabled)
```

```
{
    string str = string.Format("BaseID:{0}, Enabled:{1}", BaseID, Enabled);
    MessageBox.Show(str);
}
```

4.4.11 Base station subnet mask

Event	BaseSubnetMask (int BaseID, string Mask)
Parameter	**BaseID**: base station ID
	Mask: Subnet mask address

Sample Code:

```
baseManage.BaseSubnetMask += new
IBaseManageEvents_BaseSubnetMaskEventHandler(baseManage_BaseSubnetMask)
public void baseManage_BaseSubnetMask (int BaseID,string Mask)
{
    string str = string.Format("BaseID:{0}, Mask:{1}", BaseID, Mask);
    MessageBox.Show(str);
}
```

4.4.12 Base station gateway address

Event	BaseGatewayAddress(int BaseID, string Address)
Parameter	**BaseID**: base station ID
	Address: base gateway address

Sample Code:

```
baseManage.BaseGatewayAddress += new
IBaseManageEvents_BaseGatewayAddressEventHandler(baseManage_BaseGatewayAddress)
public void baseManage_BaseGatewayAddress(int BaseID,string Address)
{
    string str = string.Format("BaseID:{0}, Address:{1}", BaseID, Address);
    MessageBox.Show(str);
}
```

4.4.13 Base station multi channels

Event	BaseMultichannel (int BaseID, bool EnabledRelay, bool EnabledKeypadApplyRelay, int ChannelNumber, int Channel1, int Channel2, int Channel3, int Channel4)
Parameter	**BaseID:** Specify a base ID
	EnabledRelay =True, enable mobile base. =False, disable mobile base
	EnabledKeypadApplyRelay =True, allow keypad to apply mobile base. =False, not allow keypad to apply mobile base.
	ChannelNumber: Number of channels is up to 4
	Channel1: channel number is 1. **Channel2:** channel number is 2. **Channel3:** channel number is 3. **Channel4:** channel number is 4.

Sample Code:

```
baseManage.BaseMultichannel += new
IBaseManageEvents_BaseMultichannelEventHandler(baseManage_BaseMultichannel)

public void baseManage_BaseMultichannel(int BaseID,bool EnabledRelay, bool
EnabledKeypadApplyRelay,int ChannelNumber,int Channel1, int Channel2, int Channel3, int Channel4)

{

    string str = string.Format("BaseID:{0}, EnabledRelay:{1}, EnabledKeypadApplyRelay:{2},
ChannelNumber:{3}, Channel1:{4}, Channel2:{5}, Channel3:{6}, Channel4:{7}", BaseID, EnabledRelay,
EnabledKeypadApplyRelay, ChannelNumber, Channel1, Channel2, Channel3, Channel4);

    MessageBox.Show(str);

}
```

4.4.14 Keypad language

Event	KeyLanguage (int BaseID, int Language)
Parameter	BaseID: specify a base ID
	Language: keypad language

Sample Code:

```
baseManage.KeyLanguage += new
IBaseManageEvents_KeyLanguageEventHandler(baseManage_KeyLanguage)

public void baseManage_KeyLanguage(int BaseID,int Language)
```

```
{
    string str = string.Format("BaseID:{0}, Language:{1}", BaseID, Language);
    MessageBox.Show(str);
}
```

4.4.15 Keypad vibration

Event	KeyVibration(int BaseID, int Vibration)
Parameter	**BaseID:** specify a base ID
	Vibration: vibration mode =0 Off =1 On

Sample Code:

```
baseManage.KeyVibration += new IBaseManageEvents_KeyVibrationEventHandler(baseManage_KeyVibration)
public void baseManage_KeyVibration (int BaseID,int Vibration)
{
    string str = string.Format("BaseID:{0}, Vibration:{1}", BaseID, Vibration);
    MessageBox.Show(str);
}
```

4.4.16 Base station check code

Event	CheckCodeFirst(int BaseID, int CheckMode)
Parameter	**BaseID:** specify a base ID
	CheckMode: Verification Mode =0 Off =1 Last 4 digits =2 Last 6 digits

Sample Code:

```
baseManage.CheckCodeFirst += new IBaseManageEvents_CheckCodeFirstEventHandler(baseManage_CheckCodeFirst)
public void baseManage_CheckCodeFirst (int BaseID,int CheckMode)
{
```

```
        string str = string.Format("BaseID:{0}, CheckMode:{1}", BaseID, CheckMode);

        MessageBox.Show(str);

    }
```

4.4.16 Hide or show keypad ID

Event	KeyIDHide(int BaseID,int KeyIDHide)
Parameter	**BaseID:** specify a base ID
	KeyIDHide: show keypad ID or not = 0 Show = 1 Hide

Sample Code:

```
    baseManage.KeyIDHide += new IBaseManageEvents_KeyIDHideEventHandler(baseManage_KeyIDHide)

    public void baseManage_KeyIDHide(int BaseID,int KeyIDHide)

    {

        string str = string.Format("BaseID:{0}, KeyIDHide:{1}", BaseID, KeyIDHide);

        MessageBox.Show(str);

    }
```

4.4.17 Hide or show keypad shares

Event	KeySharesShow(int BaseID,int SharesShow)
Parameter	**BaseID:** specify a base ID
	KeySharesShow: show keypad shares or not = 0 Hide = 1 Show

Sample Code:

```
    baseManage.KeySharesShow += new
IBaseManageEvents_KeySharesShowEventHandler(baseManage_KeySharesShow)

    public void baseManage_KeySharesShow(int BaseID,int SharesShow)

    {

        string str = string.Format("BaseID:{0}, SharesShow:{1}", BaseID, SharesShow);

        MessageBox.Show(str);

    }
```

4.4.18 change base station channel automatically

Event	ChannelAutoChange(int BaseID,int ChannelNo)
Parameter	BaseID: specify a base ID
	ChannelNo New channel

Sample Code:

```
baseManage.ChannelAutoChange += new
IBaseManageEvents_ChannelAutoChangeEventHandler(baseManage_ChannelAutoChange)

public void baseManage_ChannelAutoChange (int BaseID,int ChannelNo)

{

    string str = string.Format("BaseID:{0}, ChannelNo:{1}", BaseID, ChannelNo);

    MessageBox.Show(str);

}
```

4.4.19 Base station channel interference

Event	ChannelInterference(int BaseID, int ChannelNo)
Parameter	BaseID: specify a base ID
	ChannelNo: occupied channel

Sample Code:

```
baseManage.ChannelInterference += new
IBaseManageEvents_ChannelInterferenceEventHandler(baseManage_ChannelInterference)

public void baseManage_ChannelInterference(int BaseID,int ChannelNo)

{

    string str = string.Format("BaseID:{0}, ChannelNo:{1}", BaseID, ChannelNo);

    MessageBox.Show(str);

}
```

Note:

Base station ID ="0" means performing on all connected base station. For a single connected base, ID is necessary to set.

5. KeypadManage

5.1 Usage Description

This object is used to get and set keypad hardware parameters for the keypads matched to the connected base station. Generally, users have to press a specific button of keypad (i.e. OK) to get and set parameters.

5.2 Attribute Description

Attribute Name	Type	Parameter Description	Note
BaseConnection	Object	\	Initialize the BaseConnection object before setting attribute, then call a method after setting attribute.

5.3 Method

Initialize BaseManage object and set the attribute of BaseConnection before calling a method

Sample Code:

```
KeypadManage keypadManage = new SunVote.KeypadManage();

BaseConnection baseConn = new SunVote.BaseConnection();

keypadManage.BaseConnection = baseConn;
```

5.3.1 Turn off keypads remotely

Turn off keypads by broadcasting command from base station. It's used to save battery after polling closed.

Method	RemoteOff (int KeyID)
Parameter	**KeyID:** keypad ID to turn off =0 Power off all keypads >0 Power off specified keypad
Return Value	No return value

Sample Code:

```
keypadManage.RemoteOff(0); //Power off all keypads remotely
```

Note:

Repeat calling this method for several times to assure all keypads off successfully.

5.3.2 Get keypad configuration

Method	GetConfig( )
Parameter	Null, when call this method, you must press a specific button on keypad (i.e. OK)
Return Value	No return value. It will generate KeyConfig or KeyConfigMatchMode events.

Sample Code:

```
keypadManage.GetConfig(); //To press a specific button of keypad to get keypad configuration
```

5.3.3 Set keypad configuration

Method	SetConfig(int KeyID, int OffTime)
Parameter	**KeyID:** keypad ID **OffTime:** auto turn off time for the keypad, =X 0≤ X ≤ 255, if X=0 use the default time set in keypad firmware, X=255, keep keypad on all the time
Return Value	No return value. It will generate KeyConfig events.

Sample Code:

```
keypadManage.SetConfig(1,10); // Set the off time of keypad 1 to 10 minutes
```

5.3.4 Get keypad hardware information

Method	GetModelInfo( )
Parameter	Null, when call this method, you must press a specific button on keypad (i.e. OK)
Return Value	No return value. It will generate KeyModelInfo events.

Sample Code:

```
keypadManage.GetModelInfo(); //To press a specific button on keypad to get keypad hardware info
```

5.3.5 Start keypad communication test

Method	string StartCommTest(int KeyID)
Parameter	**KeyID:** keypad ID
Return Value	string: = '-1': the attribute of BaseConnection is not set = '0':start successfully
	It will generate KeyCommTest events.

Sample Code:

```
keypadManage.StartCommTest(1); //Perform communication test for keypad 1
```

5.3.6 Exit keypad communication test

Method	string StopCommTest ( )
Parameter	Null
Return Value	string: = '-1': the attribute of BaseConnection is not set = '0':start successfully

Sample Code:

```
keypadManage.StopCommTest(); //Stop keypad communication test
```

5.3.7 Show keypad information

Method	ShowKeyInfo (int KeyID,int ShowMode)
Parameter	KeyID: keypad ID =0:all Keypads >0:specified Keypad ID
	ShowMode: show mode =1 show keypad ID in bold
Return Value	No return value.

Sample Code:

```
keypadManage.ShowKeyInfo(0,1); //Show Keypad ID of all Keypads
```

5.3.8 Get student name on the keypad

Method	GetStudentName(int KeyID)
Parameter	KeyID: keypad ID =0:all Keypads >0:specified Keypad ID
Return Value	No return value. It will generate KeyStudentName events.

Sample Code:

```
keypadManage.GetStudentName(1); //Get student name of keypad 1
```

5.3.9 Set student name on the keypad

Method	SetStudentName (int KeyID, string StudentName)
Parameter	KeyID: Specified Keypad ID
	StudentName: Student name
Return Value	No return value. It will generate KeyStudentName events.

Sample Code:

```
keypadManage.SetStudentName(1,'Alice'); // Set student name of keypad 1 to Alice
```

5.3.10 Get student ID on the keypad

Method	GetStudentID(int KeyID)
Parameter	KeyID: keypad ID =0:all Keypads >0:specified Keypad ID
Return Value	No return value. It will generate KeyStudentID events.

Sample Code:

```
keypadManage.GetStudentID(1); //Get student ID of keypad 1
```

5.3.11 Set student ID on the keypad

Method	SetStudentID(int KeyID, string StudentID)
Parameter	KeyID: specified Keypad ID
	StudentName: student name
Return Value	No return value. It will generate KeyStudentID events.

Sample Code:

```
//Set Student name of keypad 1 as 20160101

keypadManage.SetStudentID (1,'20160101');
```

5.3.12 Keypad ID range to turn off remotely

Method	RemoteOffRange(int KeyIDMin, int KeyIDMax)
Parameter	**KeyIDMin:** Start Keypad ID
	KeyIDMax: End Keypad ID
Return Value	No return value.

Sample Code:

```
//Remotely power off keypads of ID 1-100

keypadManage. RemoteOffRange(1, 100);
```

5.3.13 Set keypad match code, ID and channel

It's usually used to modify the match code of a batch of keypads working with a new base by broadcasting commands. In this way, there is no necessary to match keypads to new base station by pressing specific buttons separately.

Method	SetMatchIDChannel(int KeyID, string BaseHSerial, int ZoneMode, int IDMin, int IDMax,string HSerial, int NewChannel, int IDPlus, int IDValue)
Parameter	KeyID: keypad ID =0:all Keypads >0:specified Keypad ID
	BaseHSerial: leave new match code empty to keep the original match code
	ZoneMode: set keypad range to modify =0: all keypads =1: specified keypad IDs =2: specific Hardware Serial No.
	IDMin: start keypad ID is valid when ZoneMode=1
	IDMax: end keypad ID is valid when ZoneMode=1
	HSerial: hardware Serial No. is valid when ZoneMode=2
	NewChannel : when new channel=0, it means keeping the original channel
	IDPlus: how to modify Keypad ID =0: keep original Keypad ID =1: add a specific value for all keypad IDs =2: reduce a specific value for all keypad IDs =3: modify the keypad ID of a specific hardware serial No. to a specific ID value
	IDValue: when IDPlus=1 or 2, it represents the value to add or reduce; IDPlus=3 means a new Keypad ID
Return Value	No return value. As it's a broadcast command, call this method several times to assure a successful modification.

Sample Code:

```
// Change the match code of keypads 1-100 to 123456, and keep its original ID and channel

keypadManage.SetMatchIDChannel(0,'123456',1,1,100,'',0,0,0);
```

5.4 Event

5.4.1 Keypad configuration

When protocol type is ARS/EVS in base connection, it will return this even after calling the method to get and set keypad configuration.

Event	KeyConfig(int KeyID,int OffTime)
Parameter	**KeyID:** keypad ID **OffTime:** auto turn off time for the keypad,

Sample Code:

```
keypadManage.KeyConfig += new
SunVote.IKeypadManageEvents_KeyConfigEventHandler(keypadManage_KeyConfig);

public void keypadManage_KeyConfig(int KeyID, int OffTime)

{

    string str = string.Format("KeyID:%0, OffTime:%1", KeyID, OffTime);

    MessageBox.Show(str);

}
```

5.4.2 keypad hardware information

Event	KeyModelInfo(int KeyID,int HModel, int HVer, int SVer, string HSerial)
Parameter	**KeyID:** keypad ID **HModel: h**ardware model type **HVer: h**ardware version **SVer:** software version **HSerial:** hardware serial No.

Sample Code:

```
keypadManage.KeyModelInfo += new
SunVote.IKeypadManageEvents_KeyModelInfoEventHandler(keypadManage_KeyModelInfo);

public void keypadManage_KeyModelInfo(int KeyID, int HModel, int HVer, int SVer, string HSerial)

{

    string str = string.Format("KeyID:{0}, HModel:{1}, HVer:{2}, SVer:{3}, HSerial:{4}", KeyID, HModel,
HVer, SVer, HSerial);

    MessageBox.Show(str);

}
```

5.4.3 keypad communication test

Event	KeyCommTest (int KeyID, int SendTimes, int ReceiveTimes, int BaseRSSI, int KeyRSSI)
	KeyID: keypad ID

Parameter	**SendTimes:** times to send
	ReceiveTimes: times to receive data
	BaseRSSI: base station received signal strength indication from keypad (-dbm), and the smaller value is stronger.
	KeyRSSI: keypad received signal strength indication from base station (-dbm)

Sample Code:

```
keypadManage.KeyCommTest += new
SunVote.IKeypadManageEvents_KeyCommTestEventHandler(keypadManage_KeyCommTest);

public void keypadManage_KeyCommTest (int KeyID, int SendTimes, int ReceiveTimes, int BaseRSSI, int
KeyRSSI)

{

    string str = string.Format("KeyID:{0}, SendTimes:{1}, ReceiveTimes:{2}, BaseRSSI:{3}, KeyRSSI:{4}",
KeyID, SendTimes, ReceiveTimes, BaseRSSI, KeyRSSI);

    MessageBox.Show(str);

}
```

5.4.4 Students name on the keypad

Event	KeyStudentName (int KeyID,string StudentName)
Parameter	KeyID: specified Keypad ID
	StudentName: student name

Sample Code:

```
keypadManage.KeyStudentName += new
SunVote.IKeypadManageEvents_KeyStudentNameEventHandler(keypadManage_ KeyStudentName);

public void keypadManage_ KeyStudentName(int KeyID, string StudentName)

{

    string str = string.Format("KeyID:{0}, StudentName:{1}",KeyID, StudentName);

    MessageBox.Show(str);

}
```

5.4.5 Student ID on the keypad

Event	KeyStudentID(int KeyID, string StudentID)
Parameter	KeyID: specified Keypad ID
	StudentID: student ID

Sample Code:

```
keypadManage.KeyStudentID += new
SunVote.IKeypadManageEvents_KeyStudentIDEventHandler(keypadManage_ KeyStudentID);

public void keypadManage_ KeyStudentID (int KeyID, string StudentID)

{

    string str = string.Format("KeyID:{0}, StudentID:{1}",KeyID, StudentID);

    MessageBox.Show(str);

}
```

6. HardwareMonitor

6.1 Usage Description

This object is used to monitor the real-time status of keypads.

6.2 Attribute Description

Attribute Name	Type	Parameter Description	Note
BaseConnection	Object		Initialize the BaseConnection object before setting attribute, and then call a method after setting attribute.
Enabled	bool	=True to enable hardwaremonitor, and there is no need to call start method. After enabling, KeyStatus event will come along with the event of key-press.	Hardware monitor may slow down the response speed for other application objects. Recommend enabling it when the number of keypads is small and no other application object is enabled. Note that you have to set the keypad report mode to 1, 2 or 3 in basic features of base station.

Sample Code:

```
HardwareMonitor hardwareMonitor= new SunVote.HardwareMonitor();
```

```
BaseConnection baseConn = new SunVote.BaseConnection();

hardwareMonitor.BaseConnection = baseConn;

hardwareMonitor.Enabled    = true;
```

6.3 Method

Null. Set Enabled=true to open hardware monitor

6.4 Event

6.4.1 Keypad status(ID)

When the identification mode in additional configuration of base station is keypad ID, this event will come after enabling hardware monitor.

Event	KeyStatus(string BaseTag, int KeyID, int InputStatus, int ChargeStatus, double BatteryVoltage, int RfIntensity)
Parameter	**BaseTag: b**ase station tag
	KeyID: keypad ID
	InputStatus: inputstatus is the submitting status of key value (0: No keypress after start; 1: No sumbit; 2: Submitted; -1 Offline or malfunction)
	ChargeStatus: Charge status (0: No charging; 1: Charging)
	BatteryVoltage: battery voltage (Volt)
	RfIntensity: radio frequency intensity to receive signal (-dBm)
	KeyTime: time duration from base command to a keypress

Sample Code:

```
hardwareMonitor.KeyStatus += new
SunVote.IHardwareMonitorEvents_KeyStatusEventHandler(hardwareMonitor_KeyStatus);

public void hardwareMonitor_KeyStatus(string BaseTag, int KeyID, int InputStatus, int ChargeStatus, double
BatteryVoltage, int RfIntensity)

{

    string str = string.Format("BaseTag:{0}, KeyID:{1}, InputStatus:{2}, ChargeStatus:{3},
BatteryVoltage:{4}, RfIntensity:{5}", BaseTag, KeyID, InputStatus, ChargeStatus, BatteryVoltage,
RfIntensity);

    MessageBox.Show(str);

}
```

6.4.2 Keypad status(SN)

When the identification mode in additional configuration of base station is keypad SN, this event

will come after enabling hardware monitor.

Event	KeyStatusSN(string BaseTag, string KeySN, int InputStatus, int ChargeStatus, double BatteryVoltage, int RfIntensity, double KeyTime)
Parameter	**BaseTag:** base station tag
	KeySN: keypad SN
	InputStatus: inputstatus is the submitting status of key value (0: No keypress after start; 1: No sumbit; 2: Submitted; -1 Offline or malfunction)
	ChargeStatus: charge status (0: No charging; 1: Charging)
	BatteryVoltage: battery voltage (Volt)
	RfIntensity: radio frequency intensity to receive signal (-dBm)
	KeyTime: time duration from base command to a keypress

Sample Code:

```
hardwareMonitor.KeyStatusSN += new
SunVote.IHardwareMonitorEvents_KeyStatusSNEventHandler(hardwareMonitor_KeyStatusSN);

public void hardwareMonitor_KeyStatusSN(string BaseTag, int KeyID, int InputStatus, int ChargeStatus,
double BatteryVoltage, int RfIntensity, double KeyTime)

{

    string str = string.Format("BaseTag:{0}, KeyID:{1}, InputStatus:{2}, ChargeStatus:{3},
BatteryVoltage:{4}, RfIntensity:{5}, KeyTime:{6}", BaseTag, KeyID, InputStatus, ChargeStatus,
BatteryVoltage, RfIntensity, KeyTime);

    MessageBox.Show(str);

}
```

7. HardwareTest

7.1 Usage Description

This object is used to make keypads respond data quickly and automatically, and normally used to detect if keypads are online or battery is enough before official use.

7.2 Attribute Description

Attribute Name	Type	Parameter Description	Note
BaseConnection	Object		Initialize the BaseConnection object before setting attribute, and then call a method after setting attribute.

Mode	int	= 0 to report the keypad monitor status = 1 Simulate polling of single choice from 3 options. Keypad 1 pressed A, keypad 2 pressed B, keypad 3 pressed C, keypad 4 pressed A, and keypad 5 pressed B…successively.	
FirstCommitTime	int	= X, 0~255s, means after start, it will simulate keypress randomly in 0~255 seconds. X=0 to report status immediately	
IntervalCommitTime	int	= X, 0~255s, means after last submitting, it will resubmit in 0~255 seconds. X=0 to submit data once only.	
StartMode	int	= 0 resume = 1 clear to start = 2 resubmit & resume	

Sample Code

```
HardwareTest hardwareTest= new SunVote. HardwareTest();

BaseConnection baseConn = new SunVote.BaseConnection();

hardwareTest.BaseConnection = baseConn;

hardwareTest.Mode = 0;

hardwareTest.FirstCommitTime = 0;

hardwareTest.IntervalCommitTime = 0;

hardwareTest.StartMode = 1;
```

7.3 Method

7.3.1 Start hardware test

Method	string Start()
Parameter	Null
Return Value	string: = '-1': the attribute of BaseConnection is not set = '0': start successfully

	When Mode=0 and the identification mode in additional configuration of base station is keypad ID, it will generate KeyMonitorStatus events.
	When Mode=0 and the identification mode in additional configuration of base station is keypad SN, it will generate KeyMonitorStatusSN events.
	When Mode=1 and the identification mode in additional configuration of base station is keypad ID, it will generate KeyTestStatus events.
	When Mode=1 and the identification mode in additional configuration of base station is keypad SN, it will generate KeyTestStatusSN events.

Sample Code:

```
hardwareTest.Start(); //Start keypad test
```

7.3.2 Stop hardware test

Method	string Stop()
Parameter	Null
Return Value	string: = '-1': the attribute of BaseConnection is not set = '0': start successfully

Sample Code:

```
hardwareTest.Stop(); //Stop keypad test
```

7.4 Event

7.4.1 Keypad monitor status (ID)

When Mode=0 and the identification mode in additional configuration of base station is Keypad ID, it will generate this event.

Event	KeyMonitorStatus(string BaseTag, int KeyID, int InputStatus, int ChargeStatus, double BatteryVoltage, int RfIntensity)
Parameter	**BaseTag: b**ase station tag
	KeyID: keypad ID
	InputStatus: inputstatus is the submitting status of key value (0: No keypress after start; 1: No sumbit; 2: Submitted; -1 Offline or malfunction)
	ChargeStatus: Charge status (0: No charging; 1: Charging)
	BatteryVoltage: battery voltage (Volt)
	RfIntensity: radio frequency intensity to receive signal (-dBm)

Sample Code:

```
hardwareTest.KeyMonitorStatus += new
IHardwareTestEvents_KeyMonitorStatusEventHandler(hardwareTest_KeyMonitorStatus);
```

```
public void hardwareTest_KeyMonitorStatus(string BaseTag, int KeyID, int InputStatus, int ChargeStatus,
double BatteryVoltage, int RfIntensity)

{

    string str = string.Format("BaseTag:{0}, KeyID:{1}, InputStatus:{2}, ChargeStatus:{3},
BatteryVoltage:{4}, RfIntensity:{5}", BaseTag, KeyID, InputStatus, ChargeStatus, BatteryVoltage,
RfIntensity);

    MessageBox.Show(str);

}
```

7.4.2 Keypad monitor status (keypad SN)

When Mode=0 and the identification mode in additional configuration of base station is keypad SN, it will generate this event.

Event	KeyMonitorStatusSN(string BaseTag, string KeySN, int InputStatus, int ChargeStatus, double BatteryVoltage, int RfIntensity, double KeyTime)
Parameter	**BaseTag: b**ase station tag
	KeySN: keypad SN
	InputStatus: inputstatus is the submitting status of key value (0: No keypress after start; 1: No sumbit; 2: Submitted; -1 Offline or malfunction)
	ChargeStatus: Charge status (0: No charging; 1: Charging)
	BatteryVoltage: battery voltage (Volt)
	RfIntensity: radio frequency intensity to receive signal (-dBm)
	KeyTime: time duration from base command to a keypress

Sample Code:

```
hardwareTest.KeyMonitorStatusSN+= new
SunVote.IHardwareTestEvents_KeyMonitorStatusSNEventHandler(hardwareTest_KeyMonitorStatusSN);

public void hardwareTest_KeyMonitorStatusSN(string BaseTag, int KeyID, int InputStatus, int ChargeStatus,
double BatteryVoltage, int RfIntensity, double KeyTime)

{

    string str = string.Format("BaseTag:{0}, KeyID:{1}, InputStatus:{2}, ChargeStatus:{3},
BatteryVoltage:{4}, RfIntensity:{5}, KeyTime:{6}", BaseTag, KeyID, InputStatus, ChargeStatus,
BatteryVoltage, RfIntensity, KeyTime);

    MessageBox.Show(str);
```

```
}
```

7.4.3 Keypad test status (keypad ID)

When Mode=1 and the identification mode in additional configuration of base station is keypad ID, it will generate this event.

Event	KeyTestStatus(string BaseTag, int KeyID, string KeyValue)
Parameter	**BaseTag: b**ase station tag
	KeyID: keypad ID
	KeyValue: key value of keypad

Sample Code:

```
hardwareTest.KeyTestStatus += new
SunVote.IHardwareTestEvents_KeyTestStatusEventHandler(hardwareTest_KeyTestStatus);

public void hardwareTest_KeyTestStatus(string BaseTag, int KeyID, string KeyValue)

{

    string str = string.Format("BaseTag:{0}, KeyID:{1}, InputStatus:{2}", BaseTag, KeyID, KeyValue);

    MessageBox.Show(str);

}
```

7.4.4 Keypad test status (keypad SN)

When Mode=1 and the identification mode in additional configuration of base station is keypad SN, it will generate this event.

Event	KeyTestStatusSN(string BaseTag, string KeySN, string KeyValue, double KeyTime)
Parameter	**BaseTag: b**ase station tag
	KeySN: keypad SN
	KeyValue: key value of keypad
	KeyTime: time duration from base command to a keypress

Sample Code:

```
hardwareTest.KeyTestStatusSN += new
SunVote.IHardwareTestEvents_KeyTestStatusSNEventHandler(hardwareTest_KeyTestStatusSN);

public void hardwareTest_KeyTestStatusSN (string BaseTag, string KeySN, string KeyValue, double KeyTime)

{
```

```
        string str = string.Format("BaseTag:{0}, KeySN:{1}, KeyValue:{2}, KeyTime:{3}", BaseTag, KeySN,
    KeyValue, KeyTime);

        MessageBox.Show(str);

    }
```

8. Request

8.1 Usage Description

This object is used to enable a remote control (i.e. 50R/R51), who will send a request to base station proactively for service or sending command to start/stop polling instead of the base station.

8.2 Attribute Description

Attribute Name	Type	Parameter Description	Note
BaseConnection	Object		Initialize the BaseConnection object before setting attribute, and then call a method after setting attribute.
Enabled	bool	=True to enable a remote control	

Sample Code:

```
Request request = new SunVote.Request();

BaseConnection baseConn = new SunVote.BaseConnection();

request.BaseConnection = baseConn;

request.Enabled         = true;
```

8.3 Method

Null. Set Enabled=true to enable a remote control. The remote control will generate relative events after receiving data.

8.4 Event

8.4.1 Keypad status based on keypad ID

When the identification mode in additional configuration of base station is keypad ID, this event will come after enabling hardware monitor.

Event	KeyStatus(string BaseTag,int KeyID,int ReqType, int ReqData)
Parameter	**BaseTag: b**ase station tag
	KeyID: keypad ID
	ReqType: Request Type =1 Request service =2 Request to speak =3 Request to ask questions
	ReqData:Request data is 1 temporarily.

Sample Code:

```
request.KeyStatus += new SunVote.IRequestEvents_KeyStatusEventHandler(request_KeyStatus);

public void request_KeyStatus(string BaseTag, int KeyID, int ReqType, int ReqData)

{

    string str = string.Format("BaseTag:{0}, KeyID:{1}, ReqType:{2}, ReqData:{3}", BaseTag, KeyID,
ReqType, ReqData);

    MessageBox.Show(str);

}
```

8.4.2 Keypad status based on keypad SN

When the identification mode in additional configuration of base station is keypad SN, this event will come after enabling hardware monitor.

Event	KeyStatusSN(string BaseTag, string KeySN, int ReqType, int ReqData)
Parameter	**BaseTag: b**ase station tag
	KeySN: keypad SN
	ReqType: Request Type =1 Request service =2 Request to speak =3 Request to ask questions
	ReqData:Request data is 1 temporarily.

Sample Code:

```
request.KeyStatusSN += new SunVote.IRequestEvents_KeyStatusSNEventHandler(request_KeyStatusSN);

public void request_KeyStatusSN(string BaseTag, string KeySN, int ReqType, int ReqData)

{
```

```
        string str = string.Format("BaseTag:{0}, KeySN:{1}, ReqType:{2}, ReqData:{3}", BaseTag, KeySN,
ReqType, ReqData);

        MessageBox.Show(str);

    }
```

8.4.3 Remote control event in keypad ID.

When the identification mode in additional configuration of base station is Keypad ID, this event will come after enabling hardware monitor.

Event	ChairControl(string BaseTag, int KeyID, int ReqType, string ReqInfo)
Parameter	**BaseTag: b**ase station tag
	KeyID: keypad ID
	ReqType: Request Type =1 Remote control (i.e. 50R/R51)
	ReqInfo: The key value sent from the remote control is set to a specific command by application program.

Sample Code:

```
    request.ChairControl += new IRequestEvents_ChairControlEventHandler(request_ChairControl);

    public void request_ChairControl(string BaseTag, int KeyID, int ReqType, string ReqInfo)

    {

        string str = string.Format("BaseTag:{0}, KeyID:{1}, ReqType:{2}, ReqInfo:{3}", BaseTag, KeyID,
ReqType, ReqInfo);

        MessageBox.Show(str);

    }
```

8.4.3 Remote control event in keypad SN

When the identification mode in additional configuration of base station is keypad SN, this event will come after enabling hardware monitor

Event	ChairControlSN(string BaseTag, string KeySN, int ReqType, string ReqInfo)
Parameter	**BaseTag: b**ase station tag
	KeySN: keypad SN
	ReqType: Request Type =1 Remote control (i.e. 50R/R51)

	ReqInfo: The key value sent from the remote control is set to a specific command by application program.

Sample Code:

```
request.ChairControlSN += new IRequestEvents_ChairControlSNEventHandler(request_ChairControlSN);

public void request_ChairControlSN(string BaseTag, string KeySN, int ReqType, string ReqInfo)

{

    string str = string.Format("BaseTag:{0}, KeySN:{1}, ReqType:{2}, ReqInfo:{3}", BaseTag, KeySN,
ReqType, ReqInfo);

    MessageBox.Show(str);

}
```

9. SignIn

9.1 Usage Description

Roll call (Signin) is used to count the number of participants before conferences or activities. There are three modes as follows: roll call by a specific key, UID or specific key with SN. Login mode also called background login, which allows the user to start other applications at the same time. In this way, the late comer may join the response without interfering the polling process of other participants.

9.2 Attribute Description

Attribute Name	Type	Parameter Description	Note
BaseConnection	Object		Initialize the BaseConnection object before setting attribute, and then call a method after setting attribute.
Mode	int	= 0 Specific key = 1 UID = 2 Specific key with SN	
LoginMode	int	= 0 Specific key = 1 UID = 2 Student ID login = 3 Student Name login = 4 Auto ID	

		= 5 Auto Name	
LoginCodeMode	int	= 0 Auto code = 1 Manual input	
LoginAuthorizeMode	int	= 0 Auto authority = 1 Manual authority	
BackgroundSignIn	bool	= True, to enable background login. Users can start other applications at the same time. Call stop method to exit. =False to disable background login	It is effective when login mode is not set as specific key.
StartMode	int	= 0 Resume = 1 Clear to start = 2 Resubmit & resume	The attribute must be same as the previous ones.

Sample Code

```
SignIn signIn = new SunVote.SignIn();

BaseConnection baseConn = new SunVote.BaseConnection();

signIn.BaseConnection = baseConn;

signIn.Mode = 0;

signIn.StartMode = 1;
```

9.3 Method

9.3.1 start signin

Method	string Start()
Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully
	When the identification mode in additional configuration of base station is Keypad ID, It will generate KeyStatus events.
	When the identification mode in additional configuration of base station is Keypad SN, it will generate KeyStatusSN events.
	After start background login, it will generate BackgroundSignInStatus events.

Sample Code:

```
signIn.Mode = 0;        //Specific key

signIn.BackgroundSignIn = false; //Disable background login
```

```
signIn.StartMode = 1; //Restart

signIn.Start();
```

OR

```
signIn.Mode = 1;        //UID

signIn.BackgroundSignIn = true; //start background login

signIn.StartMode = 1; //Restart

signIn.Start();
```

9.3.2 Stop signin

Method	string Stop()
Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully

Sample Code:

```
signIn.Stop(); //Stop roll call
```

9.3.3 start login signin

Method	string StartLogin()
Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully
	When the identification mode in additional configuration of base station is Keypad ID, It will generate KeyLoginStatus events.
	When the identification mode in additional configuration of base station is Keypad SN, it will generate KeyLoginStatusSN events.

Sample Code:

```
signIn.LoginMode = 2;    //Roll call by Student ID login

signIn.LoginCodeMode = 0;  //Auto ID

signIn.LoginAuthorizeMode = 0;    //Auto authority
```

```
signIn.StartMode = 1;  // Restart

signIn.StartLogin();    //Start background login
```

9.3.4 Stop login signin

Method	string StopLogin()
Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully

Sample Code:

```
signIn.StopLogin(); //Stop background login
```

9.3.5 Authorize keypads in keypad ID

When start background login and auth mode is manual authority, you have to call this method to authorize the keypads. The authority results include success or failure. Only keypads which are authorized successfully shall continue to vote.

Method	SetAuthorize (int KeyID, int AuthMode)
Parameter	KeyID: keypad ID
	AuthMode: authority mode =1: success login =2: error input =3: failed to authorize
Return Value	Null. It will generate KeyAuthorize events.

Sample Code:

```
signIn.SetAuthorize(1,1);//Set keypad ID 1 as success login
```

9.3.6 Authorize keypads in keypad SN

When start background login and auth mode is manual authority, you have to call this method to authorize the keypads. The authority results include success or failure. Only keypads which are authorized successfully shall continue to vote.

Method	SetAuthorizeSN (string HSerial,int AuthMode)
Parameter	HSerial: keypad Serial No.
	AuthMode: authority mode

	=1: success login =2: error input =3: failed to authorize
Return Value	Null. It will generate KeyAuthorizeSN events.

Sample Code:

```
// Set keypad SN "123456" as success login

signIn.SetAuthorize('123456',1);
```

9.3.7 Stop background signin

Call this method to exit background login after start. It's different from the stop method of roll call. Stop roll call will not stop background login and you have to call this method after stopping roll call.

Method	StopBackgroundSignIn ()
Parameter	Null
Return Value	Null. It will generate BackgroundSignInStatus events.

Sample Code:

```
signIn. StopBackgroundSignIn(); //Stop background login
```

9.3.8 Set keypad ID dynamically

When start background login and login code mode is manual, you have to call this method to set keypad ID.

Method	SetDynamicID(string SerialID, int NewKeyID)
Parameter	**SerialID:** keypad Serial No.
	NewKeyID: new Keypad ID
Return Value	No return value. It will generate DynamicID events.

Sample Code:

```
//set the new ID to 2 for keypad SN"123456"

signIn.SetDynamicID('123456', 2);
```

9.3.9 Show sign in result on keypad

Some keypad type may support to show roll call result on keypad.

Method	ShowResult(int KeyID, int NoArrival, int ShouldArrival, int RealArrival)

Parameter	**KeyID:** Keypad ID refers to =0: All Keypads; >0: Specific keypad
	NoArrival: The number of absent
	ShouldArrival: The number to be expected
	RealArrival: The number of actual arrival
Return Value	Null

Sample Code:

```
//All Keypads show Absent 2, Expected 20, Actual 18

signIn.ShowResult(0, 2, 20, 18);
```

9.4 Event

9.4.1 Keypad status in keypad ID when sign in

When protocol type is ARS/EVS in base connection and the identification mode in additional configuration of base station is keypad ID, it will return this event.

Event	signIn.KeyStatus(string BaseTag, int KeyID, int ValueType, string KeyValue)
Parameter	**BaseTag:** base station tag
	KeyID: keypad ID
	ValueType: value type (corresponds to roll call mode)
	KeyValue: key value of keypad

Sample Code:

```
signIn.KeyStatus += new ISignInEvents_KeyStatusEventHandler(signIn_KeyStatus);

public void signIn_KeyStatus(string BaseTag, int KeyID, int ValueType, string KeyValue)

{

    string str = string.Format("BaseTag:{0},KeyID:{1},ValueType:{2}, KeyValue:{3}",  BaseTag, KeyID,
ValueType, KeyValue);

    MessageBox.Show(str);

}
```

9.4.2 Keypad status in keypad SN when sign in

When protocol type is ARS/EVS in base connection and the identification mode in additional configuration of base station is keypad SN, it will return this event.

Event	KeyStatusSN(string BaseTag, string KeySN, int ValueType, string KeyValue, double KeyTime)

Parameter	**BaseTag:** base station tag
	KeySN: keypad SN
	ValueType: value type (corresponds to roll call mode)
	KeyValue: key value of keypad
	KeyTime: time duration from base command to a keypress

Sample Code:

```
signIn.KeyStatusSN += new ISignInEvents_KeyStatusSNEventHandler(signIn_KeyStatusSN);

public void signIn_KeyStatusSN(string BaseTag, string KeySN, int ValueType, string KeyValue, double KeyTime)

{

    string str = string.Format("BaseTag:{0},KeySN:{1},ValueType:{2}, KeyValue:{3},KeyTime:{4}", BaseTag, KeySN, ValueType, KeyValue, KeyTime);

    MessageBox.Show(str);

}
```

9.4.3 Keypress status

When protocol type is CRS/ General in base connection and the roll call mode is by specific key, it will return this event.

Event	KeyPressStatus(string BaseTag, int KeyID, string KeyValue, double KeyTime)
Parameter	**BaseTag:** base station tag
	KeyID: keypad ID
	KeyValue: key value of keypad
	KeyTime: time duration from base command to a keypress

Sample Code:

```
signIn.KeyPressStatus += new ISignInEvents_KeyPressStatusEventHandler(signIn_KeyPressStatus);

public void signIn_KeyPressStatus(string BaseTag, int KeyID, string KeyValue, double KeyTime)

{

    string str = string.Format("BaseTag:{0},KeyID:{1},KeyValue:{2},KeyTime:{3}", BaseTag, KeyID, KeyValue, KeyTime);

    MessageBox.Show(str);

}
```

9.4.4 Background sign in status

Event	BackgroundSignInStatus(string BaseTag, int State)
Parameter	**BaseTag:** base station tag
	State: login status =0 Stop =1 Start

Sample Code:

```
signIn.BackgroundSignInStatus += new
ISignInEvents_BackgroundSignInStatusEventHandler(signIn_BackgroundSignInStatus);

public void signIn_BackgroundSignInStatus(string BaseTag, int State)

{

    string str = string.Format("BaseTag:{0},State:{1}", BaseTag, State);

    MessageBox.Show(str);

}
```

9.4.5 Authorize keypads in keypad ID

Event	KeyAuthorize(string BaseTag, int KeyID, int AuthMode)
Parameter	**BaseTag:** base station tag
	KeyID: keypad ID
	AuthMode: auth result mode in authority command

Sample Code:

```
signIn.KeyAuthorize += new ISignInEvents_KeyAuthorizeEventHandler(signIn_KeyAuthorize);

public void signIn_KeyAuthorize(string BaseTag, int KeyID, int AuthMode)

{

    string str = string.Format("BaseTag:{0},KeyID:{1},AuthMode{2}", BaseTag, KeyID, AuthMode);

    MessageBox.Show(str);

}
```

9.4.6 Authorize keypads in keypad SN

Event	KeyAuthorizeSN(string BaseTag, int KeyID, string HSerial, int AuthMode)
Parameter	**BaseTag:** base station tag
	KeyID: keypad ID

	HSerial: Keypad Serial No.
	AuthMode: auth result mode in authority command

Sample Code:

```
signIn.KeyAuthorizeSN += new ISignInEvents_KeyAuthorizeSNEventHandler(signIn_KeyAuthorizeSN);

public void signIn_KeyAuthorizeSN(string BaseTag, int KeyID, string HSerial, int AuthMode)

{

    string str = string.Format("BaseTag:{0},KeyID:{1}, HSerial {2},AuthMode{3}", BaseTag, KeyID, HSerial,
AuthMode);

    MessageBox.Show(str);

}
```

9.4.7 Keypad login information

Event	KeyLoginHardInfo(string BaseTag, int KeyID, int HModel, int HVer, int SVer, string HSerial)
Parameter	**BaseTag:** base station tag
	KeyID: keypad ID
	HModel: hardware model type
	HVer: hardware version
	SVer: software version
	HSerial: hardware serial No.

Sample Code:

```
signIn.KeyLoginHardInfo += new ISignInEvents_ KeyLoginHardInfoEventHandler(signIn_KeyLoginHardInfo);

public void signIn_KeyLoginHardInfo(string BaseTag, int KeyID, int HModel, int HVer, int SVer,string HSerial)

{

    string str = string.Format("BaseTag:{0},KeyID:{1}, HModel:{2}, HVer:{3}, SVer:{4}, HSerial:{5}",
BaseTag, KeyID, HModel, HVer, SVer, HSerial);

    MessageBox.Show(str);

}
```

9.4.8 Keypad logout information

Event	KeyLoginHardInfoEx(string BaseTag, int KeyID, int HModel, string HVer, string SVer, string HSerial)
Parameter	**BaseTag:** base station tag
	KeyID: keypad ID
	HModel: hardware model type

	HVer: hardware version
	SVer: software version
	HSerial: hardware serial No.

Sample Code:

```
signIn.KeyLoginHardInfoEx += new
ISignInEvents_KeyLoginHardInfoExEventHandler(signIn_KeyLoginHardInfoEx);

public void signIn_KeyLoginHardInfoEx (string BaseTag, int KeyID, int HModel, string HVer, string
SVer,string HSerial)

{

    string str = string.Format("BaseTag:{0},KeyID:{1}, HModel:{2}, HVer:{3}, SVer:{4}, HSerial:{5}",
BaseTag, KeyID, HModel, HVer, SVer, HSerial);

    MessageBox.Show(str);

}
```

9.4.9 Keypad login status in keypad ID

When the identification mode in additional configuration of base station is Keypad ID, it will return this event.

Event	KeyLoginStatus(string BaseTag, int KeyID, int ValueType, string KeyValue)
Parameter	**BaseTag**: base station tag
	KeyID: keypad ID
	ValueType: value type (corresponds to background login mode) = 0 to exit background login
	KeyValue: key value of keypad

Sample Code:

```
signIn.KeyLoginStatus += new ISignInEvents_KeyLoginStatusEventHandler(signIn_KeyLoginStatus);

public void signIn_KeyLoginStatus(string BaseTag, int KeyID, int ValueType, string KeyValue)

{

    string str = string.Format("BaseTag:{0},KeyID:{1},ValueType:{2}, KeyValue:{3}",    BaseTag, KeyID,
ValueType, KeyValue);

    MessageBox.Show(str);

}
```

9.4.10 Keypad login status in keypad SN

When the identification mode in additional configuration of base station is keypad SN, it will return this event.

Event	KeyLoginStatusSN(string BaseTag, int KeyID, string HSerial, int ValueType, string KeyValue)
Parameter	**BaseTag**: base station tag
	KeyID: keypad ID
	HSerial: keypad SN
	ValueType: value type (corresponds to background login mode) = 0 to exit background login
	KeyValue: key value of keypad

Sample Code:

```
signIn.KeyLoginStatusSN += new ISignInEvents_KeyLoginStatusSNEventHandler(signIn_KeyLoginStatusSN);

public void signIn_KeyLoginStatusSN(string BaseTag, int KeyID, string HSerial, int ValueType, string KeyValue)

{

    string str = string.Format("BaseTag:{0},KeyID:{1},ValueType:{2}, KeyValue:{3}",  BaseTag, KeyID, ValueType, KeyValue);

    MessageBox.Show(str);

}
```

9.4.11 Set keypad ID dynamically

Event	DynamicID(string BaseTag, int KeyID, string HSerial, int State)
Parameter	**BaseTag**: base station tag
	KeyID: keypad ID
	HSerial: keypad SN
	State: status to set ID =0 Failed =1 Success

Sample Code:

```
signIn.DynamicID += new ISignInEvents_DynamicIDEventHandler(signIn_DynamicID);

public void signIn_DynamicID(string BaseTag, int KeyID, string HSerial, int State)

{
```

```
        string str = string.Format("BaseTag:{0},KeyID:{1},HSerial:{2},State:{3}", BaseTag, KeyID, HSerial, State);

        MessageBox.Show(str);

    }
```

9.4.12 Keypad login information in keypad SN

Event	KeyLoginInfoSN(string BaseTag, string KeySN, int KeyID, int KeyHModel, int KeyHVer, int KeySVer, string KeyValue, double KeyTime)
Parameter	**BaseTag**: base station tag
	KeySN: keypad SN
	KeyID: keypad ID
	KeyHModel: hardware model type
	KeyHVer: hardware version
	KeySVer: software version
	KeyValue: key value of keypad
	KeyTime: time duration from base command to a keypress

Sample Code:

```
signIn.KeyLoginInfoSN += new ISignInEvents_KeyLoginInfoSNEventHandler(signIn_KeyLoginInfoSN);

public void signIn_KeyLoginInfoSN(string BaseTag, string KeySN, int KeyID, int KeyHModel, int KeyHVer, int KeySVer, string KeyValue, double KeyTime)

{

    string str =
string.Format("BaseTag:{0},KeySN:{1},KeyID:{2},KeyHModel:{3},KeyHVer:{4},KeySVer:{5},KeyValue:{6},KeyTime:{7}",

        BaseTag, KeySN, KeyID, KeyHModel, KeyHVer, KeySVer, KeyValue, KeyTime);

    MessageBox.Show(str);

}
```

10. Choices

10.1 Usage Description

It is used to do choice without sorting, including single choice and multiple choice based on the different attribute settings.

10.2 Attribute description

Attribute Name	Type	Parameter Description	Note
BaseConnection	object		Initialize the BaseConnection object before setting attribute, and then call a method after setting attribute.
OptionsMode	int	= 0 show letter = 1 show number	
ModifyMode	int	= 0 modified = 1 not allow to modify	
SecrecyMode	int	= 0 show on keypad = 1 hide on keypad	
LessEnabled	int	= 0 within upper limit = 1 equal to upper limit	
Options	int	1≤M≤10	Option number
OptionalN	int	1≤N≤M	Optional number
PromptMode	int	=0 no prompt =1 tick or cross =2 1&right answer	Call it when stop
CorrectAnswer	string	Capital letters from A to J	Call it when stop
StartMode	int	= 0 continue = 1 clear to restart = 2 restart & resume	The attributes of Start Choices must be the same as the previous ones.

Sample Code:

```
BaseConnection baseConn = new SunVote.BaseConnection();

Choices choices = new SunVote.Choices();

choices.BaseConnection = baseConn;

choices.OptionsMode = 0;

choices.ModifyMode = 1;

choices.SecrecyMode = 0;

choices.LessEnabled = 1;

choices.Options = 6;

choices.OptionalN = 2;

choices.StartMode = 1;
```

10.3 Method

10.3.1 Start vote

Method	string Start()
Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully
	When the identification mode in additional configuration of base station is Keypad ID, It will generate KeyStatus events.
	When the identification mode in additional configuration of base station is Keypad SN, it will generate KeyStatusSN events.

Sample Code:

```
choices.OptionsMode = 0; //show letter

choices.ModifyMode = 0;//not allow to modify

choices.SecrecyMode = 0;//show

choices.LessEnabled = 1;//equal to upper limit

choices.Options = 6;    //option number is 6

choices.OptionalN = 2; //selectable number is 2

choices.StartMode = 1;        //clear to restart

choices.Start();   //start vote
```

10.3.2 Stop vote

Method	string Stop()
Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully

Sample Code:

```
choices.PromptMode = 1;    /tick or cross
```

```
choices.CorrectAnswer = "AC"; //correct answers are AC

choices.Stop(); //stop vote
```

10.4 Event

10.4.1 Keypad status in keypad ID for choice

When the identification mode in additional configuration of base station is keypad ID, it will return this event.

Event	KeyStatus(string BaseTag, int KeyID, string KeyValue, double KeyTime)
Parameter	**BaseTag:** base station tag
	KeyID: keypad ID
	KeyValue: key value of keypad
	KeyTime: time duration from base command to a keypress

Sample Code:

```
choices.KeyStatus += new IChoicesEvents_KeyStatusEventHandler(choices_KeyStatus);

public void choices_KeyStatus(string BaseTag, int KeyID, string KeyValue, double KeyTime)

{

    string str = string.Format("BaseTag:{0},KeyID:{1},KeyValue:{2}, KeyTime:{3}",BaseTag, KeyID,
KeyValue, KeyTime);

    MessageBox.Show(str);

}
```

10.4.2 Keypad status in keypad SN for choice

When the identification mode in additional configuration of base station is keypad SN, it will return this event.

Event	KeyStatusSN(string BaseTag, string KeySN, string KeyValue, double KeyTime)
Parameter	**BaseTag:** base station tag
	KeySN: keypad SN
	KeyValue: key value of keypad
	KeyTime: time duration from base command to a keypress

Sample Code:

```
choices.KeyStatusSN += new IChoicesEvents_KeyStatusSNEventHandler(choices_KeyStatusSN);

public void choices_KeyStatusSN(string BaseTag, string KeySN, string KeyValue, double KeyTime)
```

```
{
    string str = string.Format("BaseTag:{0}, KeySN:{1},KeyValue:{2}, KeyTime:{3}",BaseTag, KeySN,
KeyValue, KeyTime);
    MessageBox.Show(str);
}
```

11.True or False

11.1 Attribute description

Attribute Name	Type	Parameter Description	Note
BaseConnection	Object		Initialize the BaseConnection object before setting attribute, and then call a method after setting attribute.
Mode	int	= 1 True/ False =2 Yes/ No =3 Yes/No/ Abstain	
ModifyMode	int	= 0 not allow to modify = 1 modified	
SecrecyMode	int	= 0 show on keypad = 1 hide on keypad	
PromptMode	int	=0 no prompt =1 tick or cross =2 1& right answer	Call it when stop
CorrectAnswer	string	Capital letters from A to J	Call it when stop
StartMode	int	= 0 continue = 1 clear to restart = 2 restart & resume	The attributes of Start Choices must be the same as the previous ones.

Sample Code:

```
TrueFalse trueFalse = new SunVote.TrueFalse();
BaseConnection baseConn = new SunVote.BaseConnection();
trueFalse.BaseConnection = baseConn;
trueFalse.Mode = 0;
trueFalse.ModifyMode = 1;
trueFalse.SecrecyMode = 0;
trueFalse.StartMode = 1;
```

11.2 Method

11.2.1 Start vote

Method	string Start()
Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully
	When the identification mode in additional configuration of base station is Keypad ID, It will generate KeyStatus events.
	When the identification mode in additional configuration of base station is Keypad SN, it will generate KeyStatusSN events.

Sample Code:

```
trueFalse.Mode = 0; //yes/not

trueFalse.ModifyMode = 0;//not allowed to modify

trueFalse.SecrecyMode = 0;//show on the keypad

trueFalse.StartMode = 1;      //clear to restart

trueFalse.Start();//start vote
```

11.2.2 Stop vote

Method	string Stop()
Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully

Sample Code:

```
trueFalse.PromptMode = 1;    //Right or wrong

trueFalse.CorrectAnswer = '1'; //Correct answer is Yes

trueFalse.Stop(); //Stop vote
```

11.3 Event

11.3.1 Keypad status in keypad ID for judge

When the identification mode in additional configuration of base station is keypad ID, it will return this event.

Event	KeyStatus(string BaseTag, int KeyID, string KeyValue, double KeyTime)
Parameter	**BaseTag:** base station tag
	KeyID: keypad ID
	KeyValue: key value of keypad
	KeyTime: time duration from base command to a keypress

Sample Code:

```
trueFalse.KeyStatus += new ITrueFalseEvents_KeyStatusEventHandler(trueFalse_KeyStatus);

public void trueFalse_KeyStatus(string BaseTag, int KeyID, string KeyValue, double KeyTime)

{

    string str = string.Format("BaseTag:{0},KeyID:{1},KeyValue:{2}, KeyTime:{3}",BaseTag, KeyID,
KeyValue, KeyTime);

    MessageBox.Show(str);

}
```

11.3.2 Keypad status in keypad SN for judge

When the identification mode in additional configuration of base station is keypad SN, it will return this event.

Event	KeyStatusSN(string BaseTag, string KeySN, string KeyValue, double KeyTime)
Parameter	**BaseTag:** base station tag
	KeySN: keypad SN
	KeyValue: key value of keypad
	KeyTime: time duration from base command to a keypress

Sample Code:

```
trueFalse.KeyStatusSN += new ITrueFalseEvents_KeyStatusSNEventHandler(trueFalse_KeyStatusSN);

public void trueFalse_KeyStatusSN(string BaseTag, string KeySN, string KeyValue, double KeyTime)

{

    string str = string.Format("BaseTag:{0}, KeySN:{1},KeyValue:{2}, KeyTime:{3}",BaseTag, KeySN,
```

```
KeyValue, KeyTime);

        MessageBox.Show(str);

}
```

12.Ranking

12.1 Usage Description

Ranking is similar to choice, except that the voting result is submitted in order.

different attribute settings.

12.2 Attribute description

Attribute Name	Type	Parameter Description	Note
BaseConnection	object		Initialize the BaseConnection object before setting attribute, and then call a method after setting attribute.
OptionsMode	int	= 0 show letter = 1 show number	
ModifyMode	int	= 0 not allow to modify = 1 modified	
SecrecyMode	int	= 0 show on keypad = 1 hide on keypad	
LessEnabled	int	= 0 within upper limit = 1 equal to upper limit =2 allow repeat input =3 1&2, used in text	
OptionalN	int	1≤M≤10	Option number
OptionalCount	int	1≤N≤M	Optional number
PromptMode	int	=0 no prompt =1 tick or cross =2 1& right answer	Call it when stop
CorrectAnswer	string	capital letters from A to J	Call it when stop
StartMode	int	= 0 continue = 1 clear to restart = 2 restart & resume	The attributes must set the same as the previous ones.

Sample Code

```
Sequence sequence = new SunVote.Sequence();
```

```
BaseConnection baseConn = new SunVote.BaseConnection();
sequence.BaseConnection = baseConn;
sequence.OptionsMode = 0;
sequence.ModifyMode = 1;
sequence.LessEnabled = 2;
sequence.SecrecyMode = 0;
sequence.OptionalN = 6;
sequence.OptionalCount = 6;
sequence.StartMode = 1;
```

12.3 Method

12.3.1 Start vote

Method	string Start()
Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully
	When the identification mode in additional configuration of base station is Keypad ID, It will generate KeyStatus events.
	When the identification mode in additional configuration of base station is Keypad SN, it will generate KeyStatusSN events.

Sample Code

```
sequence.OptionsMode = 0; // show letter
sequence.ModifyMode = 0;// not allow to modify
sequence.SecrecyMode = 0;//show on the keypad
sequence.LessEnabled = 2;// allow repeat input
sequence.OptionalN = 6;     // option number is 6
sequence.OptionalCount = 6; // selectable number is 6
sequence.StartMode    = 1; ///clear to restart
```

```
sequence.Start();       //start to vote
```

12.3.2 Stop vote

Method	string Stop()
Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully

Sample Code

```
sequence.PromptMode = 1;    // Right or wrong

sequence.CorrectAnswer = 'CAAB'; // Correct answer is CAAB

sequence.Stop(); //STOP VOTE
```

12.4 Event

12.4.1 Keypad status in keypad ID for ranking

When the identification mode in additional configuration of base station is keypad ID, it will return this event.

Event	KeyStatus(string BaseTag, int KeyID, string KeyValue, double KeyTime)
Parameter	**BaseTag:** base station tag
	KeyID: keypad ID
	KeyValue: key value of keypad
	KeyTime: time duration from base command to a keypress

Sample Code

```
sequence.KeyStatus += new ISequenceEvents_KeyStatusEventHandler(sequence_KeyStatus);

public void sequence_KeyStatus(string BaseTag, int KeyID, string KeyValue, double KeyTime)

{

    string str = string.Format("BaseTag:{0},KeyID:{1},KeyValue:{2}, KeyTime:{3}",BaseTag, KeyID,
KeyValue, KeyTime);

    MessageBox.Show(str);

}
```

12.4.2 Keypad status in keypad SN for ranking

When the identification mode in additional configuration of base station is keypad SN, it will return this event.

Event	KeyStatusSN(string BaseTag, string KeySN, string KeyValue, double KeyTime)
Parameter	**BaseTag:** base station tag
	KeySN: keypad SN
	KeyValue: key value of keypad
	KeyTime: time duration from base command to a keypress

Sample Code

```
sequence.KeyStatusSN += new ISequenceEvents_KeyStatusSNEventHandler(sequence_KeyStatusSN);

public void sequence_KeyStatusSN(string BaseTag, string KeySN, string KeyValue, double KeyTime)

{

    string str = string.Format("BaseTag:{0}, KeySN:{1},KeyValue:{2}, KeyTime:{3}",BaseTag, KeySN, KeyValue, KeyTime);

    MessageBox.Show(str);

}
```

13.Number

13.1 Attribute description

Attribute Name	Type	Parameter Description	Note
BaseConnection	Object		Initialize the BaseConnection object before setting attribute, and then call a method after setting attribute.
Mode	int	= 0, Max 16 digits including decimal point =1, score of 0-100 integer with range. =2, score of decimals with range.	
ModifyMode	int	= 0 not allow to modify = 1 modified	

SecrecyMode	int	= 0 show on keypad = 1 hide on keypad	
Default	int		Mode is valid for 1
Min	double		Mode is valid except for 0
Max	double		Mode is valid except for 0
Decimal	int		Mode is valid for 2
PromptMode	int	=0 no prompt =1 tick or cross =2 1& right answer	Call it when stop
CorrectAnswer	string		Call it when stop
StartMode	int	= 0 continue = 1 clear to restart = 2 restart & resume	The attributes must be the same as the previous ones.

Sample Code

```
Number number = new SunVote.Number();

BaseConnection baseConn = new SunVote.BaseConnection();

number.BaseConnection = baseConn;

number.Mode = 1;

number.Min = 6;

number.Max = 100;

number.Default = 10;

number.StartMode = 1;
```

13.2 Method

13.2.1 Start vote

Method	string Start()
Parameter	Null
	String:

Return Value	= '-1': the attribute of BaseConnection is not set = '0': start successfully
	When the identification mode in additional configuration of base station is Keypad ID, It will generate KeyStatus events.
	When the identification mode in additional configuration of base station is Keypad SN, it will generate KeyStatusSN events.

Sample Code

```
number.Mode = 1; //score of integer

number.ModifyMode = 0;// not allow to modify

number.SecrecyMode = 0;//show on the keypad

number.Default = 0;//default is 0

number.Min = 0; //min is 0

number.Max = 100; //max is 100

number.StartMode = 1;       //clear to start

number.Start();  //start vote
```

12.2.2 Stop vote

Method	string Stop()
Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully

Sample Code

```
number.PromptMode = 1;    //right or wrong

number.CorrectAnswer = '95'; //correct answer is 95

number.Stop(); //stop vote
```

13.3 Event

13.3.1 Keypad status in keypad ID

When the identification mode in additional configuration of base station is keypad ID, it will return this event.

Event	KeyStatus(string BaseTag, int KeyID, string KeyValue, double KeyTime)
Parameter	**BaseTag:** base station tag
	KeyID: keypad ID
	KeyValue: key value of keypad
	KeyTime: time duration from base command to a keypress

Sample Code

```
number.KeyStatus += new INumberEvents_KeyStatusEventHandler(number_KeyStatus);

public void number_KeyStatus(string BaseTag, int KeyID, string KeyValue, double KeyTime)

{

    string str = string.Format("BaseTag:{0},KeyID:{1},KeyValue:{2}, KeyTime:{3}",BaseTag, KeyID,
KeyValue, KeyTime);

    MessageBox.Show(str);

}
```

13.3.2 Keypad status in keypad SN

When the identification mode in additional configuration of base station is keypad SN, it will return this event.

Event	KeyStatusSN(string BaseTag, string KeySN, string KeyValue, double KeyTime)
Parameter	**BaseTag:** base station tag
	KeySN: keypad SN
	KeyValue: key value of keypad
	KeyTime: time duration from base command to a keypress

Sample Code

```
number.KeyStatusSN += new INumberEvents_KeyStatusSNEventHandler(number_KeyStatusSN);

public void number_KeyStatusSN(string BaseTag, string KeySN, string KeyValue, double KeyTime)

{

    string str = string.Format("BaseTag:{0}, KeySN:{1},KeyValue:{2}, KeyTime:{3}",BaseTag, KeySN,
KeyValue, KeyTime);

    MessageBox.Show(str);

}
```

14.Text

14.1 Attribute description

Attribute Name	Type	Parameter Description	Note
BaseConnection	Object		Initialize the BaseConnection object before setting attribute, and then call a method after setting attribute.
ClozeType	int	= 0 any letter = 1 number options	
ModifyMode	int	= 0 not allow to modify = 1 modifiable	
SecrecyMode	int	= 0 show on keypad = 1 hide on Keypad	
MaxLength	int	1-16	
PromptMode	int	=0 no prompt =1 tick or cross =2 1& right answer	Call it when stop
CorrectAnswer	string		Call it when stop
StartMode	int	= 0 continue = 1 clear to restart = 2 restart & resume	The attributes must be the same as the previous ones.

Sample Code

```
Cloze cloze = new SunVote.Cloze();

BaseConnection baseConn = new SunVote.BaseConnection();

cloze.BaseConnection = baseConn;

cloze.ClozeType = 0;

cloze.ModifyMode = 1;

cloze.SecrecyMode = 0;

cloze.MaxLength = 5;

cloze.StartMode = 1;
```

14.2 Method

14.2.1 Start vote

Method	string Start()

Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully
	When the identification mode in additional configuration of base station is Keypad ID, It will generate KeyStatus events.
	When the identification mode in additional configuration of base station is Keypad SN, it will generate KeyStatusSN events.

Sample Code

```
cloze. ClozeType = 1; //Text

cloze.ModifyMode = 0;// not allow to modify

cloze.SecrecyMode = 0;//show on the keypad

cloze.MaxLength = 5;//max length is 5

cloze.StartMode = 1;   // clear to start

cloze.Start();      //start vote
```

14.2.2 Stop vote

Method	string Stop()
Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully

Sample Code

```
cloze.PromptMode = 1;    //right or wrong

cloze.CorrectAnswer = '5124'; //correct answer is 5124

cloze.Stop(); //stop vote
```

14.3 Event

14.3.1 Keypad status in keypad ID

When the identification mode in additional configuration of base station is keypad ID, it will return this event.

Event	KeyStatus(string BaseTag, int KeyID, string KeyValue, double KeyTime)
Parameter	**BaseTag:** base station tag
	KeyID: keypad ID
	KeyValue: key value of keypad
	KeyTime: time duration from base command to a keypress

Sample Code

```
cloze.KeyStatus += new IClozeEvents_KeyStatusEventHandler(cloze_KeyStatus);

public void cloze_KeyStatus(string BaseTag, int KeyID, string KeyValue, double KeyTime)

{

    string str = string.Format("BaseTag:{0},KeyID:{1},KeyValue:{2}, KeyTime:{3}",BaseTag, KeyID,
KeyValue, KeyTime);

    MessageBox.Show(str);

}
```

14.3.2 Keypad status in keypad SN

When the identification mode in additional configuration of base station is keypad SN, it will return this event.

Event	KeyStatusSN(string BaseTag, string KeySN, string KeyValue, double KeyTime)
Parameter	**BaseTag:** base station tag
	KeySN: keypad SN
	KeyValue: key value of keypad
	KeyTime: time duration from base command to a keypress

Sample Code

```
cloze.KeyStatusSN += new IClozeEvents_KeyStatusSNEventHandler(cloze_KeyStatusSN);

public void cloze_KeyStatusSN(string BaseTag, string KeySN, string KeyValue, double KeyTime)

{

    string str = string.Format("BaseTag:{0}, KeySN:{1},KeyValue:{2}, KeyTime:{3}",BaseTag, KeySN,
KeyValue, KeyTime);

    MessageBox.Show(str);

}
```

15.Rush Answer

15.1 Attribute description

Attribute Name	Type	Parameter Description	Note
BaseConnection	Object		Initialize the BaseConnection object before setting attribute, and then call a method after setting attribute.
Mode	int	= 0 specific key = 1 any key	
StartMode	int	= 0 continue = 1 clear to restart = 2 restart & resume	The attributes must be the same as the previous ones.

Sample Code

```
RushAnswer rushAnswer = new SunVote.RushAnswer();

BaseConnection baseConn = new SunVote.BaseConnection();

rushAnswer.BaseConnection = baseConn;

rushAnswer.Mode = 0;

rushAnswer.StartMode = 1;
```

15.2 Method

15.2.1 Start vote

Method	string Start()
Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully
	When the identification mode in additional configuration of base station is Keypad ID, It will generate KeyStatus events.
	When the identification mode in additional configuration of base station is Keypad SN, it will generate KeyStatusSN events.

Sample Code

```
rushAnswer.Mode := 0; //Specific Key
```

```
rushAnswer.StartMode := 1;//restart

rushAnswer.Start();     //start rush answer
```

15.2.2 Stop vote

Method	string Stop()
Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully

Sample Code

```
rushAnswer.Stop(); //stop rush answer
```

15.3 Event

15.3.1 Keypad status in keypad ID

When the identification mode in additional configuration of base station is keypad ID, it will return this event.

Event	KeyStatus(string BaseTag, int KeyID, string KeyValue, double KeyTime)
Parameter	**BaseTag:** base station tag
	KeyID: keypad ID
	KeyValue: key value of keypad
	KeyTime: time duration from base command to a keypress

Sample Code

```
rushAnswer.KeyStatus += new IRushAnswerEvents_KeyStatusEventHandler(rushAnswer_KeyStatus);

public void rushAnswer_KeyStatus(string BaseTag, int KeyID, string KeyValue, double KeyTime)

{

    string str = string.Format("BaseTag:{0},KeyID:{1},KeyValue:{2}, KeyTime:{3}",BaseTag, KeyID,
KeyValue, KeyTime);

    MessageBox.Show(str);

}
```

15.3.2 Keypad status in keypad SN

When the identification mode in additional configuration of base station is keypad SN, it will return

this event.

Event	KeyStatusSN(string BaseTag, string KeySN, string KeyValue, double KeyTime)
Parameter	**BaseTag:** base station tag
	KeySN: keypad SN
	KeyValue: key value of keypad
	KeyTime: time duration from base command to a keypress

Sample Code

```
rushAnswer.KeyStatusSN += new
IRushAnswerEvents_KeyStatusSNEventHandler(rushAnswer_KeyStatusSN);

public void rushAnswer_KeyStatusSN(string BaseTag, string KeySN, string KeyValue, double KeyTime)

{

    string str = string.Format("BaseTag:{0}, KeySN:{1},KeyValue:{2}, KeyTime:{3}",BaseTag, KeySN,
KeyValue, KeyTime);

    MessageBox.Show(str);

}
```

16.Vote

16.1 Vote (simple vote)

16.1.1 Attribute description

Attribute Name	Type	Parameter Description	Note
BaseConnection	Object		Initialize the BaseConnection object before setting attribute, and then call a method after setting attribute.
Mode	int	= 0 3-key vote = 1 2-key vote = 2 other options with title = 3 3-key vote with shares = 4 2-key vote with shares	Note : When vote with shares, please call Election cumulative voting to download shares.
ModifyMode	int	= 0 not allow to modify = 1 modifiable	
SecrecyMode	int	= 0 show on keypad u 1 hide on Keypad	

Title	String	Up to 14 bytes or 7 characters	
TitleVoteMode	int	=1 Yes/No =2 Yes/No/Abstain =3 For/Against/Abstain =4 Satisfied / Acceptable / Unsatisfied =5 Satisfied / Acceptable/ Unsatisfied / Not Sure =6 Satisfied / Acceptable / Neutral / Unsatisfied =7 VerySatisfied /Satisfied / Not Sure / Unsatisfied / Very Unsatisfied =8 Totally Agree / Agree / NotSure/Disagree/TotallyDisagree =9 Excellent / Good/ Poor =10 Excellent / Good / Average/ Poor =11 Excellent / Competent /Fair/ Incompetent =101 Customized Options	Built in 11, you can customize the 1, custom rules to download the keypad
DownloadSuccessKeyIDs	string	= '-2': without starting effective download = '-1': the attribute of BaseConnection is not set = '0': failed to download for all items =string: means keypads that download successfully	Read only attributes, set up automatically when you stop downloading
DownloadErrorKeyIDs	string	= '-2': without starting effective Download = '-1': the attribute of BaseConnection is not set =' 0': all download successfully = string : means keypad that Download unsuccessfully	Read only attributes, set up automatically when stop downloading
DownloadSuccessKeySNs	string	= '-2': without starting up valid download = '-1': the attribute of BaseConnection is not set = '0': all failed to download = String :indicates keypad that download succcessfully	Read only attributes, set up automatically when stop downloading
DownloadErrorKeySNs	string	= '-2': without starting up valid download = '-1': the attribute of BaseConnection is not set ='0': all download successfully = string :indicates keypad failed to download	Read only attributes, set up automatically when stop downloading

Start Mode	int	= 0 resume = 1 clear to restart = 2 Restart & Resume	The attributes must be the same as the previous ones.

Sample Code

```
Vote vote = new SunVote.Vote();

BaseConnection baseConn = new SunVote.BaseConnection();

vote.BaseConnection = baseConn;

vote.Mode = 1;

vote.ModifyMode = 0;

vote.SecrecyMode = 0;

vote.StartMode = 1;
```

16.1.2 Method

16.1.2.1 Start vote

Method	string Start()
Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully
	When the attribute of Mode<3 and the identification mode in additional configuration of base station is Keypad ID, It will generate KeyStatus events.
	When the attribute of Mode<3 and the identification mode in additional configuration of base station is Keypad SN, it will generate KeyStatusSN events.
	When the attribute of Mode >=3 and the identification mode in additional configuration of base station is Keypad ID, It will generate KeyBallotStatus events.
	When the attribute of Mode<>= and the identification mode in additional configuration of base station is Keypad SN, it will generate KeyBallotStatusSNevents.

Sample Code

```
vote.Mode := 2; //vote with title

vote.ModifyMode := 0;// not allow to modify

vote.SecrecyMode := 0;//show
```

```
vote.Title := 'please vote';    //vote title

vote.TitleVoteMode := 2; // Yes/No/Abstain

vote.StartMode := 1;   // restart

vote.Start();       //start up voting
```

16.1.2.2 Stop vote

Method	string Stop()
Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully

Sample Code

```
vote.Stop(); //Stop vote
```

16.1.2.3 Show vote result

Method	ShowResult(int KeyID, int PassStatus, int NoKey, int Yes, int No, int Abs)
Parameter	KeyID: the keypad ID =0: all keypads >0: specific keypad
	PassStatus: =0: not show =1: pass =2: not pass
	NoKey: the number of people who donot vote Yes: approval number No: opposition number Abs: waiver number
Return Value	Null

Sample Code

```
//the vote is passed with all users voting. 10 for Yes, 1 for No and 1 for Abstain

vote.ShowResult(0, 1, 0, 10, 1, 1);
```

16.1.2.4 Start to download rules based on keypad ID

Method	string StartDownloadRules (string KeyIDs, object DataBuf)
Parameter	KeyIDs: keypad ID to download
	DataBuf:downloaded data，two-dimensional character array DataBuf(M,N） M:0 (1 custom rule)。 N:1(0=default value，1=grade name)。 Default value:0—Y(grade number)，0 means empty Grade name: the name of grade 1，…, the name of grade Y". The number of grade is no more than 8 while the name is no more than 4 characters e.g.:DataBuf(0,1) The number of appraisal rules is 1 DataBuf(0,0)="1": default value is excellent DataBuf(0,1)=" Excellent / Good / Average/ Poor"
Return Value	String ='-1', the attribute of BaseConnection is not set. ='0', start successfully = '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate DataDownload event

Sample Code

```
string[,] lArray = new string[1, 2] { {"1", "excellent,good,average,poor" } };

vote.StartDownloadRules('1-10', lArray);
```

16.1.2.5 Start to download rules based on keypad SN

Method	string StartDownloadRulesSN (string KeySNs, object DataBuf)
Parameter	KeySNs: keypad SN to be downloaded
	DataBuf:downloaded data，two-dimensional character array DataBuf(M,N） M:0 (1 custom rule)。 N:1(0=default value，1=grade name)。 Default value:0 - Y(grade number)，0 means empty Grade name: the name of grade 1，…, the name of grade Y". The number of grade is no more than 8 while the name is no more than 4 characters

	e.g.:DataBuf(0,1) The number of appraisal rules is 1 DataBuf(0,0)="1": default value is excellent DataBuf(0,1)=" Excellent / Good / Average/ Poor"
Return Value	String ='-1', the attribute of BaseConnection is not set. ='0', start successfully
	Will generate DataDownloadSNevent

Sample Code

```
string[,]
 lArray = new string[1, 2] { {"1", "excellent,good,average,poor"} };
vote.StartDownloadRulesSN('123456,123457,123458', lArray);
```

16.1.2.6 Stop to download

Applied only when stop manually. Generally, it stops automatically when download ends, and no need to call this method.

Method	StopDownload ()
Parameter	Null
Return Value	Null, successful and failed download keypad attribute will be set up when it stops.

Sample Code

```
vote.StopDownload(); //stop to download
```

16.1.3 Event

16.1.3.1 Keypad status in keypad ID

When the identification mode in additional configuration of base station is keypad ID, it will return this event.

Event	KeyStatus(string BaseTag, int KeyID, string KeyValue, double KeyTime)
Parameter	**BaseTag:** base station tag
	KeyID: keypad ID
	KeyValue: key value of keypad
	KeyTime: time duration from base command to a keypress

Sample Code

```
vote.KeyStatus += new IVoteEvents_KeyStatusEventHandler(vote_KeyStatus);

public void vote_KeyStatus(string BaseTag, int KeyID, string KeyValue, double KeyTime)

{

    string str = string.Format("BaseTag:{0},KeyID:{1},KeyValue:{2}, KeyTime:{3}",BaseTag, KeyID,
KeyValue, KeyTime);

    MessageBox.Show(str);

}
```

16.1.3.2 Keypad status in keypad SN

When the identification mode in additional configuration of base station is keypad SN, it will return this event.

Event	KeyStatusSN(string BaseTag, string KeySN, string KeyValue, double KeyTime)
Parameter	**BaseTag:** base station tag
	KeySN: keypad SN
	KeyValue: key value of keypad
	KeyTime: time duration from base command to a keypress

Sample Code

```
vote.KeyStatusSN += new IVoteEvents_KeyStatusSNEventHandler(vote_KeyStatusSN);

public void vote_KeyStatusSN(string BaseTag, string KeySN, string KeyValue, double KeyTime)

{

    string str = string.Format("BaseTag:{0}, KeySN:{1},KeyValue:{2}, KeyTime:{3}",BaseTag, KeySN,
KeyValue, KeyTime);

    MessageBox.Show(str);

}
```

16.1.3.3 Download data on keypad based on keypad ID

This event will generate after calling StartDownloadRules().

Event	DataDownload(int KeyID, int DownloadStatus, string DownloadInfo)
	KeyID: keypad ID

Parameter	**DownloadStatus:**download status = 1: downloaded successfully = -1: failed to download = 0: downloaded is completed
Return Value	**DownloadInfo:** downloaded information (BEGIN、the current page/the total pages、STOP)

Sample Code

```
DataDownload (2,1, '1/4');// download the first page data successfully to keypad ID=2

DataDownload (3,-1, '1/4');// failed to download the first page data to keypad ID=3

DataDownload (2,1, 'STOP') // finished downloading to keypad ID=2

DataDownload (0,0, 'STOP') // finished downloading to all keypad
```

16.1.3.4 Download data on keypad based on keypad SN

Event	DataDownloadSN(string KeySN, int DownloadStatus, string DownloadInfo)
Parameter	KeySN: keypad SN
	DownloadStatus:download status = 1: downloaded successfully = -1: failed to download = 0: downloaded is completed
Return Value	**DownloadInfo:** downloaded information (BEGIN、the current page/the total pages、STOP)

Sample Code

```
DataDownloadSN('111',1,'1/4');// download the first page data successfully to keypad SN=111

DataDownloadSN('112',-1,'1/4');// failed to download the first page data to keypad SN=112

DataDownloadSN('113',1,'STOP') // finished downloading to keypad SN=113

DataDownloadSN(0,0,'STOP') // finished downloading to all keypad
```

16.1.3.5 Voting shares in keypad ID

When the identification mode in additional configuration of base station is keypad ID, it will generate this event.

Event	KeyBallotStatus(string BaseTag, int KeyID, string KeyValue, string Ballots, double KeyTime)
Parameter	**BaseTag:** base station tag
	KeyID: keypad ID
	KeyValue: key value of keypad
	Ballots: shares

	KeyTime: time duration from base command to a keypress

Sample Code

```
KeyBallotStatus('1',1,'1','10',1.8);

// While connect with base ID=1, keypad ID=1 vote for YES with 10 shares, and the key time is 1.8
```

16.1.3.6 Voting shares in keypad SN

When the identification mode in additional configuration of base station is keypad SN, it will generate this event.

Event	KeyBallotStatusSN(string BaseTag, string KeySN, string KeyValue, string Ballots, double KeyTime)
Parameter	**BaseTag:** base station tag
	KeySN: keypad SN
	KeyValue: key value of keypad
	Ballots: shares
	KeyTime: time duration from base command to a keypress

Sample Code

```
KeyBallotStatusSN('1','123456','1','10',1.8);
```

16.2 Batch Vote

16.2.1 Attribute description

Attribute Name	Type	Parameter Description	Note
BaseConnection	Object		Initialize the BaseConnection object before setting attribute, and then call a method after setting attribute.
BatchMode	int	= 0 consecutive Item ID = 1 random Item ID need to download random item index table = 2 item ID is limited	
ModifyMode	int	= 0 not allow to modify = 1 modifiable	
SecrecyMode	int	= 0 show on keypad =1 hide on keypad	
LessMode	int	= 0 within upper limit = 1 equal to upper limit	

NumberBegin	int	The start value of the vote object item number, and it cannot exceed the number of the downloaded item.	Valid when batch mode is not equal to 1
NumberEnd	int	The start value of the vote object item number, and it cannot exceed the number of the downloaded item.	Valid when batch mode is equal to 0
FixBallotMode	int	=0 unlimited =1 limited	
AvoidMode	int	=0 disable =1 enabled	Valid when Mode is 1, note: When the avoid mode is enabled, it needs to call the download method in the avoid table download object to download avoid table
LimitItemNumber	string	e.g. 1-3,5 stands for item No.1,2,3,5.	Valid when BatchMode is 2
Ballots	string	Limit ballots from 0 to 255, separated by a comma. When ballot is 0, there is no limitation.	Valid when FixBallotMode is 1
DownloadSuccessKeyIDs	string	= '-2': without starting effective download = '-1': the attribute of BaseConnection is not set = '0': failed to download for all items =string: means keypads that download successfully	Read only attributes, set up automatically when stop downloading
DownloadErrorKeyIDs	string	= '-2': without starting effective Download = '-1': the attribute of BaseConnection is not set =' 0': all download successfully = string : means keypad that Download unsuccessfully	Read only attributes, set up automatically when stop downloading
DownloadSuccessKeySNs	string	= '-2': without starting up valid download = '-1': the attribute of BaseConnection is not set = '0': all failed to download = String :indicates keypad that download succcessfully	Read only attributes, set up automatically when stop downloading
DownloadErrorKeySNs	string	= '-2': without starting up valid download = '-1': the attribute of BaseConnection is not set ='0': all download successfully = string :indicates keypad failed to download	Read only attributes, set up automatically when stop downloading

StartMode	int	= 0 resume = 1 clear to restart = 2 Restart & Resume	The attributes must set the same as the previous ones.

Sample Code

```
BatchVote batchVote = new SunVote.BatchVote();

BaseConnection baseConn = new SunVote.BaseConnection();

batchVote.BaseConnection = baseConn;
```

16.2.2 Method

16.2.2.1 Start vote

Method	string Start()
Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully
	When tthe identification mode in additional configuration of base station is Keypad ID, It will generate KeyStatus events.
	When the identification mode in additional configuration of base station is Keypad SN, it will generate KeyStatusSN events.

Sample Code

```
batchVote.NumberBegin = 1; //the begin number is 1

batchVote.NumberEnd = 5;//the end number is 5

batchVote.BatchMode = 0;// consecutive item ID

batchVote.LessMode = 1;// equal to upper limit

batchVote.FixBallotMode = 0;// ballot is not limited

batchVote.SecrecyMode = 1; //hide on keypads

batchVote.ModifyMode = 0;// not allow to modify

batchVote.AvoidMode = 0;// not enable avoid mode

batchVote.StartMode = 1;//clear to restart

batchVote.Start(); //start voting
```

16.2.2.2 Stop vote

Method	string Stop()
Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully

Sample Code

```
batchVote.Stop(); //stop vote
```

16.2.2.3 Start to download items based on keypad ID

Method	string StartDownloadItems(string KeyIDs, object DataBuf)
Parameter	KeyIDs: keypad ID to download
	DataBuf:downloaded data， two-dimensional character array DataBuf(M,N） M: the number of item name(0—X)。 X: If the name of the item is under 7 Chinese characters, it can save about 4096 items at the same time. If the name of the item is under 32 Chinese characters in average, it can save about 1000 items. N:1(0=rules ID， 1=item name string character) rules ID: 1=3- key vote (Yes/No/Abstain) 2=2-key vote (Yes/No) e.g. DataBuf(1,1),the number of item is 2。 DataBuf(0， 0)= "1" DataBuf(0， 1)= "vote 1" DataBuf(1， 0)= "2" DataBuf(1， 1)= "vote 2"
Return Value	String ='-1', the attribute of BaseConnection is not set. ='0', start successfully = '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate DataDownload event

Sample Code

```
string[,] lArray = new string[2, 2] { {"1", "ITEM1"},{"2", "ITEM2"} };

batchVote.StartDownloadItems('1-10', lArray);
```

16.2.2.4 Start to download items based on keypad SN

Method	string StartDownloadItemsSN (string KeySNs, object DataBuf)
Parameter	KeySNs: keypad SN to be downloaded
	DataBuf: downloaded data，two-dimensional character array DataBuf(M,N)
	M: the number of item name (0—X)。X:If the name of the item is under 7 Chinese characters, it can save about 4096 items at the same time. If the name of the item is under 32 Chinese characters in average, it can save about 1000 items.
	N:1(0= rules ID， 1= item name string character)
	rules ID: 1=3-key vote (Yes/No/Abstain)
	2=2-key vote(Yes/No)
	e.g. DataBuf(1,1), the number of item is 2。
	DataBuf(0， 0)= "1"
	DataBuf(0， 1)= "vote 1"
	DataBuf(1， 0)= "2"
	DataBuf(1， 1)= "vote 2"
Return Value	String
	='-1', the attribute of BaseConnection is not set.
	='0', start successfully
	= '-2': invalid keypad ID to be downloaded
	= '-3', invalid downloaded item or exceed the maximum value
	Will generate DataDownloadSNevent

Sample Code

```
string[,] lArray = new string[2, 2] { {"1", "item1"},{"2", "item2"} };

batchVote.StartDownloadItemsSN('123456,123457,123458', lArray);
```

16.2.2.5 Start to download random items based on keypad ID

Method	string StartDownloadRandomItems (string KeyIDs, object DataBuf)

Parameter	KeyIDs: keypad ID to be downloaded
	DataBuf: downloaded data，two-dimensional character array DataBuf(M,N) e.g. DataBuf(1) Random number item is 2， indicates that No.1 item and No.3 item are selected out to vote DataBuf(0)= "1" DataBuf(1)= "3"
Return Value	String ='-1', the attribute of BaseConnection is not set. ='0', start successfully = '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate DataDownloadevent

Sample Code

```
string[] lArray = new string[2] {"1","3"};

batchVote.StartDownloadRandomItems('1-10', lArray);
```

16.2.2.6 Start to download random items based on keypad SN

Method	string StartDownloadRandomItemsSN (string KeySNs, object DataBuf)
Parameter	KeySNs: keypad SN to be downloaded
	DataBuf: downloaded data，two-dimensional character array DataBuf(M,N) e.g. DataBuf(1) Random number item is 2， indicates that No.1 item and No.3 item are selected out to vote DataBuf(0)= "1" DataBuf(1)= "3"
Return Value	String ='-1', the attribute of BaseConnection is not set. ='0', start successfully = '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate DataDownloadSNevent

Sample Code

```
string[] lArray = new string[2] {"1","3"};
```

```
batchVote.StartDownloadRandomItemsSN('123456,123457,123458', lArray);
```

16.2.2.6 Stop to download

Applied only when stop manually. Generally, it stops automatically when download ends, and no need to call this method.

Method	StopDownload ()
Parameter	Null
Return Value	Null, successful and failed download keypad attribute will be set up when it stops.

Sample Code

```
batchVote.StopDownload(); //stop to download
```

16.2.3 Event

16.2.3.1 Keypad status in keypad ID

When the identification mode in additional configuration of base station is keypad ID, it will return this event.

Event	KeyStatus(string BaseTag, int KeyID, int CommitOK, string KeyValue)
Parameter	**BaseTag:** base station tag
	KeyID: keypad ID
	CommitOK: whethe to submit the voting data by pressing "OK" key on the keypad = 0: without pressing "OK" key = 1: pressed "OK" key
	KeyValue: key value of keypad

Sample Code

```
batchVote.KeyStatus += new IBatchVoteEvents_KeyStatusEventHandler(batchVote_KeyStatus);

public void batchVote_KeyStatus(string BaseTag, int KeyID, int CommitOK, string KeyValue)

{

    string str = string.Format("BaseTag:{0},KeyID:{1},CommitOK:{2}, KeyValue:{3}", BaseTag,
KeyID,CommitOK, KeyValue);

    MessageBox.Show(str);

}
```

16.2.3.2 Keypad status in keypad SN

When the identification mode in additional configuration of base station is keypad SN, it will return this event.

Event	KeyStatusSN(string BaseTag, string KeySN, int CommitOK, string KeyValue)
Parameter	**BaseTag:** base station tag
	KeySN keypad SN
	CommitOK: whethe to submit the voting data by pressing "OK" key on the keypad = 0: without pressing "OK" key = 1: pressed "OK" key
	KeyValue: key value of keypad

Sample Code

```
batchVote.KeyStatusSN += new IBatchVoteEvents_KeyStatusSNEventHandler(batchVote_KeyStatusSN);

public void batchVote_KeyStatusSN (string BaseTag, int KeySN, int CommitOK, string KeyValue)

{

    string str = string.Format("BaseTag:{0},KeySN:{1},CommitOK:{2}, KeyValue:{3}", BaseTag,
KeySN,CommitOK, KeyValue);

    MessageBox.Show(str);

}
```

16.2.3.3 Download data on the keypad

This event will generate after calling the download method (ID) to start downloading. DataDownload(0,0,'STOP') means that download is finished. So we can the keypad attribute about the keypad that downloaded successfully and unsuccessfully.

Event	DataDownload(int KeyID, int DownloadStatus, string DownloadInfo)
Parameter	**KeyID: keypad ID**
	DownloadStatus: download status =1. Download successfully =-1, download failure =0, download is finished
	DownloadInfo: download information
	KeyValue: key value of keypad

Sample Code

```
DataDownload (2,1, '1/4');// download the first page data successfully to keypad ID=2

DataDownload (3,-1, '1/4');// failed to download the first page data to keypad ID=3
```

```
DataDownload (2,1, 'STOP') // finished downloading to keypad ID=2

DataDownload (0,0, 'STOP') // finished downloading to all keypad
```

16.2.3.4 Download data on the keypad based on keypad SN

This event will generate after calling the download method (ID) to start downloading.

Event	DataDownloadSN(string KeySN, int DownloadStatus, string DownloadInfo)
Parameter	**KeySN: keypad SN**
	DownloadStatus: download status =1. Download successfully =-1, download failure =0, download is finished
	DownloadInfo: download information

Sample Code

```
DataDownloadSN('111',1,'1/4');// download the first page data successfully to keypad SN=111

DataDownloadSN('112',-1,'1/4');// failed to download the first page data to keypad SN=112

DataDownloadSN('113',1,'STOP') // finished downloading to keypad SN=113

DataDownloadSN(0,0,'STOP') // finished downloading to all keypad
```

17.Score

17.1 Simple score

17.1.1 Attribute description

Attribute Name	Type	Parameter Description	Note
BaseConnection	Object		Initialize the BaseConnection object before setting attribute, and then call a method after setting attribute.
Mode	int	= 0 no rule，With a maximum of 16 decimal places = 1 integer scoring 0-100 = 2 decimal scoring = 5 integer scoring 0-255 with Upper and lower limit, name and number = 6 batch integer scoring 0-255 with Upper and lower limit,name	

ModifyMode	int	= 0 not allow to modify = 1 modifiable	
SecrecyMode	int	= 0 show on keypad = 1 hide on keypad	
Default	int	0-100	Valid when mode is 1
Min	double		Valid when mode is not 0
Max	double		valid when mode is not 0
Decimal	int		Valid when mode is 2
ScoreTitle	string		Valid when Mode is 5
ScoreID	int		Valid when Mode is 5
ItemNumber	int		Valid when Mode is 6
LessMode	int		Valid when Mode is 6
PromptMode	int	=0 no prompt =1 tick or cross =2 1 & right answer	Call when stops
CorrectAnswer	string	= 0 within upper limit = 1 equal to upper limit	Call when stops
StartMode	int	= 0 resume = 1 clear to restart = 2 resume & restart	The attributes must set the same as the previous ones.

Sample Code

```
Score score = new SunVote.Score();

BaseConnection baseConn = new SunVote.BaseConnection();

score.BaseConnection = baseConn;
```

17.1.2 Method

17.1.2.1 Start vote

Method	string Start()
Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully
	When the attribute of Mode is not equal to 6 and the identification mode in additional configuration of base station is Keypad ID, It will generate KeyStatus events.
	When the attribute of Mode is not equal to 6 and the identification mode in additional configuration of base station is Keypad SN, it will generate KeyStatusSN events.

	When the attribute of Mode =6 and the identification mode in additional configuration of base station is Keypad ID, It will generate KeyBatchStatus events.

Sample Code

```
score.SecrecyMode = 1; //hide on keypad

score.ModifyMode = 1; //allow to modify

score.Mode = 1;    // integer scoring with default value, upper and lower limits

score.Default = 0; //the default value

score.Min = 0; //min value

score.Max = 100; //max value

score.StartMode = 1;// clear the restart

score.Start();//start vote
```

17.1.2.2 Stop vote

Method	string Stop()
Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully

Sample Code

```
score.Stop(); //stop vote
```

17.1.2.3 Show vote result on keypad

Method	ShowResult(int KeyID, string TotalScore)
Parameter	KeyID: the keypad ID =0: all keypads >0: specific keypad
	TotalScore: total score
Return Value	Null

Sample Code

```
score.ShowResult(0, '95.9'); // show that score is 95.9
```

17.1.2.4 Clear scores

Method	CleanRecord(int KeyID)
Parameter	KeyID: the keypad ID =0: all keypads >0: specific keypad
Return Value	Null, will generate KeyCleanRecord event.

Sample Code

```
score.CleanRecord(1); //Clear the score record on keypad ID=1
```

17.1.3 Event

17.1.3.1 Keypad status in keypad ID

When the identification mode in additional configuration of base station is keypad ID, it will return this event.

Event	KeyStatus(string BaseTag, int KeyID, string KeyValue, double KeyTime)
Parameter	**BaseTag:** base station tag
	KeyID: keypad ID
	KeyValue: key value of keypad
	KeyTime: time duration from base command to a keypress

Sample Code

```
score.KeyStatus += new IScoreEvents_KeyStatusEventHandler(score_KeyStatus);

public void score_KeyStatus(string BaseTag, int KeyID, string KeyValue, double KeyTime)

{

    string str = string.Format("BaseTag:{0},KeyID:{1},KeyValue:{2}, KeyTime:{3}",BaseTag, KeyID,
KeyValue, KeyTime);

    MessageBox.Show(str);

}
```

17.1.3.2 Keypad status in keypad SN

When the identification mode in additional configuration of base station is keypad SN, it will return this event.

Event	KeyStatusSN(string BaseTag, string KeySN, string KeyValue, double KeyTime)

Parameter	**BaseTag:** base station tag
	KeySN: keypad SN
	KeyValue: key value of keypad
	KeyTime: time duration from base command to a keypress

Sample Code

```
score.KeyStatusSN += new IScoreEvents_KeyStatusSNEventHandler(score_KeyStatusSN);

public void score_KeyStatusSN(string BaseTag, string KeySN, string KeyValue, double KeyTime)

{

    string str = string.Format("BaseTag:{0}, KeySN:{1},KeyValue:{2}, KeyTime:{3}",BaseTag, KeySN,
KeyValue, KeyTime);

    MessageBox.Show(str);

}
```

17.1.3.4 Keypad batch status

When the identification mode in additional configuration of base station is keypad ID, it will return this event.

Event	KeyBatchStatus(string BaseTag, int KeyID, int CommitOK, string KeyValue)
Parameter	**BaseTag:** base station tag
	KeyID: keypad ID
	CommitOK: whether to submitted by pressing "ok" key = 0: without pressing "ok" key = 1: pressed "ok" key
	KeyValue: key value of keypad

Sample Code

```
score.KeyBatchStatus += new IScoreEvents_KeyBatchStatusEventHandler(score_KeyBatchStatus);

public void score_KeyBatchStatus(string BaseTag, int KeyID, int CommitOK, string KeyValue)

{

    string str = string.Format("BaseTag:{0},KeyID:{1},CommitOK:{2}, KeyValue:{3}", BaseTag,
KeyID,CommitOK, KeyValue);

    MessageBox.Show(str);

}
```

17.1.3.5 Clear score record

Event	KeyCleanRecord(int KeyID, int CleanStatus)
Parameter	KeyID: keypad ID
	CleanStatus: whether to clear or not

Sample Code

```
score.KeyCleanRecord += new IScoreEvents_KeyCleanRecordEventHandler(score_KeyCleanRecord);

public void score_KeyCleanRecord(int KeyID, int CleanStatus)

{

    string str = string.Format("BaseTag:{0}, CleanStatus:{1}", KeyID, CleanStatus);

    MessageBox.Show(str);

}
```

17.2 Batch Score

17.2.1 Attribute description

Attribute Name	Type	Parameter Description	Note
BaseConnection	Object		Initialize the BaseConnection object before setting attribute, and then call a method after setting attribute.
BatchMode	int	= 0 consecutive Item ID = 1 random Item ID need to download random item index table = 2 item ID is limited	
ModifyMode	int	= 0 not allow to modify = 1 modifiable	
SecrecyMode	int	= 0 show on keypad =1 hide on keypad	
LessMode	int	= 0 within upper limit = 1 equal to upper limit	
NumberBegin	int	the voted object name's Item number begin value ,ID cannot exceed the number of downloaded item	Valid when batch mode is not 1
NumberEnd	int	the voted object name's Item number end value ,ID cannot exceed the number of downloaded item	Valid when batch mode is 0

AvoidMode	int	=0 Not enabled =1 enabled	Valid when Mode is equal to 1, note: When the avoid mode is enabled, it needs to call the download method in the avoid table download object to download avoid table
LimitItemNumber	string	e.g. 1-3,5 on behalf of No.1,2,3,5 item	Valid when BatchMode is 2
DownloadSuccessKeyIDs	string	= '-2': without starting up valid download。 = '-1': without setting up Base Connection。 = '0': all keypads failed to download = String indicateds the keypad downloaded successfully	Read only attributes, set up automatically when stop downloading
DownloadErrorKeyIDs	string	= '-2': without starting up valid download . = '-1': without setting up Base Connection =' 0': all keypads download successsfully = string: indicates the keypad failed to download	Read only attributes, set up automatically when stop downloading
StartMode	int	= 0 resume = 1 clear to restart = 2 resume & restart	The attributes must set the same as the previous ones.

Sample Code

```
BatchScore batchScore = new SunVote.BatchScore();

BaseConnection baseConn = new SunVote.BaseConnection();

batchScore.BaseConnection = baseConn;
```

17.2.2 Method

17.2.2.1 Start vote

Method	string Start()
Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully

	When tthe identification mode in additional configuration of base station is Keypad ID, It will generate KeyStatus events.

Sample Code

```
batchScore.NumberBegin = 1;    // the start Item ID is 1
batchScore.NumberEnd = 5;// the end Item ID is 5
batchScore.BatchMode = 0;// Consecutive Item ID
batchScore.LessMode = 1;// Equal To Upper Limit
batchScore.SecrecyMode = 1; //hide on keypad
batchScore.ModifyMode = 0;// not allow to modify
batchScore.AvoidMode = 0;// not enable avoid mode
batchScore.StartMode = 1;//clear to restart
batchScore.Start(); //start scoring
```

17.2.2.2 Stop vote

Method	string Stop()
Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully

Sample Code

```
batchScore.Stop(); //stop vote
```

17.2.2.3 Start to download items based on keypad ID

Method	string StartDownloadItems(string KeyIDs, object DataBuf)
Parameter	KeyIDs: keypad ID to download
	DataBuf:downloaded data，two-dimensional character array DataBuf(M,N） M: the number of item name(0—X)。 X: If the name of the item is under 7 Chinese characters, it can save about 4096 items at the same time. If the name of the item is under 32 Chinese characters in average, it can save about 1000 items. N:1(0=rules ID，1=item name string character)

	rules ID: X(customized score rules) e.g. DataBuf(1,1) ,the number of item is 2。 DataBuf(0， 0)= "1" DataBuf(0， 1)= "score item 1" DataBuf(1， 0)= "2" DataBuf(1， 1)= "score item 2"
Return Value	String ='-1', the attribute of BaseConnection is not set. ='0', start successfully = '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate DataDownload event

Sample Code

```
string[,] lArray = new string[2, 2] { {"1", "item1"},{"2", "item2"} };

batchScore.StartDownloadItems('1-10', lArray);
```

17.2.2.4 Start to download random items based on keypad ID

Method	string StartDownloadRandomItems (string KeyIDs, object DataBuf)
Parameter	KeyIDs: keypad ID to download
	DataBuf: downloaded data ， character array DataBuf(M） e.g. DataBuf(1) Random number item is 2， indicates that No.1 item and No.3 item are selected out to vote DataBuf(0)= "1" DataBuf(1)= "3"
Return Value	String ='-1', the attribute of BaseConnection is not set. ='0', start successfully = '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate DataDownload event

Sample Code

```
string[] lArray = new string[2] {"1","3"};
```

```
batchScore.StartDownloadRandomItems('1-10', lArray);
```

17.2.2.5 Start to download score rules based on keypad ID

Method	string StartDownloadRules(string KeyIDs, object DataBuf):String
Parameter	KeyIDs: keypad ID to download
	downloaded data, Two-dimensional character array DataBuf(M， N)。 M:0-7， the number of scoring rules (1—8)。 N:5(0=unit， 1= lower limit of score， 2= upper limit of score， 3=default value， 4=integer place， 5= decimal place)。 Value can be inputted no more than 8 places ， decimal point is deemed to one place。 Unit:no more than 2 chinese charater Lower limit of score:minimum value of digital input, showing " -1" when it is not used. Upper limit of score:maxmum value of digital input , showing " -1" when it is not used. Default value:giving defaut value automatically when no value is inputted, showing " -1" when it is not used. Integer place:demical place of digital input' Decimal place:decimal place of digital input。 e.g. DataBuf(1,5)the number of scoring rules is 2 DataBuf(0,0)="": The unit is empty。 DataBuf(0,1)="0": The lower limit of score is 0。 DataBuf(0,2)="100": The upper limit of score is 100 DataBuf(0,3)="0": The default value is 0。 DataBuf(0,4)="3":The Integer number is 3。 DataBuf(0,5)="0": The decimal number is 0. DataBuf(1,0)="score":The unit is score。 DataBuf(1,1)="0": The lower limit of score is 0。 DataBuf(1,2)="10": The upper limit of score is 10 DataBuf(1,3)="0": The default value is 0. DataBuf(1,4)="2": The Integer number is 3. DataBuf(1,5)="1": The decimal number is 1.
Return value	String ='-1', the attribute of BaseConnection is not set.

	='0', start successfully = '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate DataDownload event

Sample Code

```
string[,] lArray = new string[1, 6] {{"Score","0","10","0","2","1"}};

batchScore.StartDownloadRules('1-10', lArray);
```

17.2.2.6 Stop to download

Applied only when stop manually. Generally, it stops automatically when download ends, and no need to call this method.

Method	StopDownload ()
Parameter	Null
Return Value	Null, successful and failed download keypad attribute will be set up when it stops.

Sample Code

```
batchScore.StopDownload(); //stop to download
```

17.2.3 Event

17.2.3.1 Keypad status in keypad ID

When the identification mode in additional configuration of base station is keypad ID, it will return this event.

Event	KeyStatus(string BaseTag, int KeyID, int CommitOK, string KeyValue)
Parameter	**BaseTag:** base station tag
	KeyID: keypad ID
	CommitOK: whethe to submit the voting data by pressing "OK" key on the keypad = 0: without pressing "OK" key = 1: pressed "OK" key
	KeyValue: key value of keypad

Sample Code

```
batchScore.KeyStatus += new IBatchScoreEvents_KeyStatusEventHandler(batchScore_KeyStatus);

public void batchScore_KeyStatus(string BaseTag, int KeyID, int CommitOK, string KeyValue)

{
```

```
        string str = string.Format("BaseTag:{0},KeyID:{1},CommitOK:{0},KeyValue:{1}", BaseTag, KeyID,
    CommitOK, KeyValue);

        MessageBox.Show(str);

    }
```

17.2.3.2 Download data on the keypad

This event will generate after calling the download method (ID) to start downloading. DataDownload(0,0,'STOP') means that download is finished.

Event	DataDownload(int KeyID, int DownloadStatus, string DownloadInfo)
Parameter	**KeyID: keypad ID**
	DownloadStatus: download status =1. Download successfully =-1, download failure =0, download is finished
	DownloadInfo: download information

Sample Code

```
DataDownload (2,1, '1/4');// download the first page data successfully to keypad ID=2

DataDownload (3,-1, '1/4');// failed to download the first page data to keypad ID=3

DataDownload (2,1, 'STOP') // finished downloading to keypad ID=2

DataDownload (0,0, 'STOP') // finished downloading to all keypad
```

18. Batch Evaluation

18.1 Attribute description

Attribute Name	Type	Parameter Description	Note
BaseConnection	Object		Initialize the BaseConnection object before setting attribute, and then call a method after setting attribute.
BatchMode	int	= 0 continuous numbering = 1 random numbering, an index table for random number has to be downloaded. = 2 limited numbering	
ModifyMode	int	= 0 modification not allowed = 1 modification allowed	
SecrecyMode	int	= 0 show on keypad	

		= 1 hide on keypad	
LessMode	int	= 0 within upper limit = 1 equal to upper limit	
NumberBegin	int	Starting value of item number for candidate object, which cannot exceed the number of downloaded items.	Valid when Batch mode≠1
NumberEnd	int	End value of item number for candidate object, which cannot exceed the number of downloaded items.	Valid when batch mode=0
FixBallotMode	int	=0 not fixed =1 fixed	
AvoidMode	int	=0 not activate =1 activate	Valid when Mode=1, Notice: call a download method in download objects of avoid lists to download avoid lists.
LimitItemNumber	string	For example, 1-3,5 represent item 1, 2, 3, and 5	Valid when BatchMode=2
Ballots	string	Ballot limits (0-255)，separate with commas, no limit when set up to 0, meaning the maximum votes to each level.	Valid when FixBallotMode = 1
DownloadSuccessKeyIDs	string	= '-2': no valid download is activated. = '-1': BaseConnection is not set. = '0': all downloads are failed. = String represents keypad that is successfully downloaded.	READ ONLY, automatically set up when download stops.
DownloadErrorKeyIDs	string	= '-2': no valid download is activated. = '-1': BaseConnection is not set. =' 0': all downloads are successful. = string represents keypad failed to download.	READ ONLY, automatically set up when download stops.
DownloadSuccessKeySNs	string	= '-2': no valid download is activated. = '-1': BaseConnection is not set. = '0': all downloads are failed. = String represents keypad that is successfully downloaded.	READ ONLY, automatically set up when download stops.
DownloadErrorKeySNs	string	= '-2': no valid download is activated. = '-1': BaseConnection is not set. =' 0': all downloads are successful. = string represents keypad failed to download.	READ ONLY, automatically set up when download stops.
StartMode	int	= 0 continue = 1 clear to restart = 2 restart & resume	The attributes must set the same as the previous ones.

Sample Code

```
BatchEvaluation batchEvaluation = new SunVote.BatchEvaluation();

BaseConnection baseConn = new SunVote.BaseConnection();

batchEvaluation.BaseConnection = baseConn;
```

18.2 Method

18.2.1 Start vote

Method	string Start()
Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully
	When tthe identification mode in additional configuration of base station is Keypad ID, It will generate KeyStatus events.
	When the identification mode in additional configuration of base station is Keypad SN, it will generate KeyStatusSN events.

Sample Code

```
batchEvaluation.NumberBegin = 1;    //Start Number1

batchEvaluation.NumberEnd = 5;//End Number5

batchEvaluation.BatchMode = 0;//continuous numbering

batchEvaluation.LessMode = 1;//LessMode is not activated

batchEvaluation.FixBallotMode = 0;//within upper limit

batchEvaluation.SecrecyMode = 1; //hide on keypad

batchEvaluation.ModifyMode = 0;//Modification not allowed

batchEvaluation.AvoidMode = 0;//AvoidMode is not applied

batchEvaluation.StartMode = 1;//restart

batchEvaluation.Start(); //start vote
```

18.2.2 Stop vote

Method	string Stop()

Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully

Sample Code

```
batchEvaluation.Stop(); //stop vote
```

18.2.2.3 Start to download items based on keypad ID

Method	string StartDownloadItems(string KeyIDs, object DataBuf)
Parameter	KeyIDs: keypad ID to download
	DataBuf:data to be downloaded, two-dimensional array of strings. DataBuf(M,N）. M: number of item (0—X)。X:if the length of item name is less than 7, characters, around 4096 items can be saved. If the average length of item name is less than 32 characters, around 1000 items can be saved. N:2(0= rule type， 1=rule number， 2=character string of item name) Rule type: 1=default evaluation; 2=user-defined evaluation; Rule number: X=corresponding rule type number For example:DataBuf(1,2) , the number of item is 2. DataBuf(0， 0)= "1" DataBuf(0， 1)= "1" DataBuf(0， 2)= "evaluation 1" DataBuf(1， 0)= "1" DataBuf(1， 1)= "1" DataBuf(1， 2)= "evaluation 2"
Return Value	String ='-1', the attribute of BaseConnection is not set. ='0', start successfully = '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate DataDownload event

Sample Code

```
string[,] lArray = new string[1, 3] {{ "2", "1","item1" }}; batchEvaluation.StartDownloadItems('1-10', lArray);
```

18.2.2.4 Start to download items based on keypad SN

Method	string StartDownloadItemsSN (string KeySNs, object DataBuf)
Parameter	KeySNs: keypad SN to be downloaded
	DataBuf:data to be downloaded, two-dimensional array of strings DataBuf(M,N）. M: number of item (0－X). if the length of item name is less than 7, characters, around 4096 items can be saved. If the average length of item name is less than 32 characters, around 1000 items can be saved. N:2(0= rule type， 1=rule number， 2=character string of item name) Rule type: 1=default evaluation; 2=user-defined evaluation; For example:DataBuf(1,2) , the number of item is 2. DataBuf(0， 0)= "1" DataBuf(0， 1)= "1" DataBuf(0， 2)= "evaluation1" DataBuf(1， 0)= "1" DataBuf(1， 1)= "1" DataBuf(1， 2)= "evaluation2"
Return Value	String ='-1', the attribute of BaseConnection is not set. ='0', start successfully = '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate DataDownloadSN event

Sample Code

```
string[,] lArray = new string[1, 3] {{ "2", "1","ITEM1" }};

batchEvaluation.StartDownloadItemsSN('111,222,333', lArray);
```

18.2.2.5 Start download random item based on keypad ID

Method	string StartDownloadRandomItems (string KeyIDs, object DataBuf)
	KeyIDs: keypad ID to be downloaded

Parameter	DataBuf: data to be downloaded， character array DataBuf(M） For example: DataBuf(1) random item number is 2, meaning that item 1 and 3 are selected to vote. DataBuf(0)= "1" DataBuf(1)= "3"
Return Value	String ='-1', the attribute of BaseConnection is not set. ='0', start successfully = '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate DataDownloadevent

Sample Code

```
string[] lArray = new string[2] {"1","3"};

batchEvaluation.StartDownloadRandomItems('1-10', lArray);
```

17.2.2.6 Start to download random items based on keypad SN

Method	string StartDownloadRandomItemsSN (string KeySNs, object DataBuf)
Parameter	KeySNs: keypad SN to be downloaded
	DataBuf: data to be downloaded, character array DataBuf(M ） For example: DataBuf(1) random item number is 2, meaning item 1 and 3 are selected to vote. DataBuf(0)="1" DataBuf(1)="3"
Return Value	String ='-1', the attribute of BaseConnection is not set. ='0', start successfully = '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate DataDownloadSNevent

Sample Code

```
string[] lArray = new string[2] {"1","3"};

batchEvaluation.StartDownloadRandomItems('111,222', lArray);
```

18.2.2.7 Start download evaluation rules based on keypad ID

Method	string StartDownloadRules(string KeyIDs, object DataBuf)
Parameter	KeyIDs: keypad ID to be downloaded
	DataBuf: data to be downloaded, two-dimensional array of strings DataBuf(M,N） M:number of user-defined rules. -1。 N:1(0=default value， 1=name of level)。 Default value: 0—Y(rank)， 0 represents null Level name: “level 1 name, …， level Y name”. The maximum number of level is 8, 4 character in maximum in the name. For example: DataBuf(0,1) the number of rule is 1 DataBuf(0,0)=”1”: default value is excellent DataBuf(0,1)=”excellent, good, medium, poor”
Return Value	String =’-1’, the attribute of BaseConnection is not set. =’0’, start successfully = ‘-2’: invalid keypad ID to be downloaded = ‘-3’, invalid downloaded item or exceed the maximum value
	Will generate DataDownloadevent

Sample Code

```
string[,] lArray = new string[2, 2] { {"3"," excellent, good, general, poor" },{"3","excellent, qualified, general, unqualified"} };

batchEvaluation.StartDownloadRules('1-3', lArray);
```

18.2.2.8 Start download evaluation rules based on keypad SN

Method	string StartDownloadRulesSN(string KeySNs, object DataBuf)
Parameter	KeyIDs: keypad ID to be downloaded
	KeySNs: keypad SN to be downloaded DataBuf: see StartDownloadRules as reference
Return Value	String =’-1’, the attribute of BaseConnection is not set. =’0’, start successfully

	= '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate DataDownloadSN event

Sample Code

```
string[,] lArray = new string[2, 2] { {"3"," excellent, good, general, poor " },{"3"," excellent, qualified, general, unqualified "} };

batchEvaluation.StartDownloadRulesSN('123456,123457,123458', lArray);
```

18.2.2.9 Stop to download

Applied only when stop manually. Generally, it stops automatically when download ends, and no need to call this method.

Method	StopDownload ()
Parameter	Null
Return Value	successful and failed download keypad attribute will be set up when it stops

Sample Code

```
batchEvaluation.StopDownload(); //stop download
```

18.3 Event

18.3.1 Keypad status in keypad ID

When the identification mode in additional configuration of base station is keypad ID, it will return this event.

Event	KeyStatus(string BaseTag, int KeyID, int CommitOK, string KeyValue)
Parameter	**BaseTag:** base station tag
	KeyID: keypad ID
	CommitOK: whether to submit the voting data by pressing "OK" key on the keypad = 0: without pressing "OK" key = 1: pressed "OK" key
	KeyValue: key value of keypad

Sample Code

```
batchEvaluation.KeyStatus += new

IBatchEvaluationEvents_KeyStatusEventHandler(batchEvaluation_KeyStatus);

public void batchEvaluation_KeyStatus(string BaseTag, int KeyID, int CommitOK, string KeyValue)
```

```
{
    string str = string.Format("BaseTag:{0},KeyID:{1},CommitOK:{0},KeyValue:{1}", BaseTag, KeyID,
CommitOK, KeyValue);
    MessageBox.Show(str);
}
```

18.3.2 Keypad status in keypad SN

When the identification mode in additional configuration of base station is keypad SN, it will return this event.

Event	KeyStatusSN(string BaseTag, string KeySN, int CommitOK, string KeyValue)
Parameter	**BaseTag:** base station tag
	KeySN keypad SN
	CommitOK: whethe to submit the voting data by pressing "OK" key on the keypad = 0: without pressing "OK" key = 1: pressed "OK" key
	KeyValue: key value of keypad

Sample Code

```
batchEvaluation.KeyStatusSN += new
IBatchEvaluationEvents_KeyStatusSNEventHandler(batchEvaluation_KeyStatusSN);

public void batchEvaluation_KeyStatusSN(string BaseTag, string KeySN, int CommitOK, string KeyValue)

{
    string str = string.Format("BaseTag:{0},KeySN:{1},CommitOK:{0},KeyValue:{1}", BaseTag, KeySN,
CommitOK, KeyValue);
    MessageBox.Show(str);
}
```

18.2.3 Download data on the keypad based on keypad ID

Generated when download method (ID) is called. Download ends when DataDownload(0,0,'STOP') appears. The attribute of keypads that successfully and unsuccessfully downloaded are available to read.

Event	DataDownload(int KeyID, int DownloadStatus, string DownloadInfo)
Parameter	**KeyID: keypad ID**
	DownloadStatus: download status

	=1. Download successfully =-1, download failure =0, download is finished
	DownloadInfo: download information (BEGIN、current page/total page、STOP)

Sample Code

```
DataDownload (2,1, '1/4');//keypad 2 downloaded page 1 successfully

DataDownload (3,-1, '1/4');//keypad 3 failed to download page 1

DataDownload (2,1, 'STOP') //download of keypad 2 finished

DataDownload (0,0, 'STOP') //download of all keypads finished
```

18.2.4 Download data on the keypad based on keypad SN

Generated when download method (SN) is called. Download ends when DataDownload(0,0,'STOP') appears. The attribute of keypads that successfully and unsuccessfully downloaded are available to read.

Event	DataDownloadSN(string KeySN, int DownloadStatus, string DownloadInfo)
Parameter	**KeySN: keypad SN**
	DownloadStatus: download status =1. Download successfully =-1, download failure =0, download is finished
	DownloadInfo: download information (BEGIN、current page/total page、STOP)

Sample Code

```
DataDownloadSN('111',1,'1/4');//SN'111'keypad downloaded page1 successfully

DataDownloadSN('112',-1,'1/4');// SN'112'keypad failed to download page1

DataDownloadSN('113',1,'STOP') // SN'113' keypad downloaded successfully

DataDownloadSN(0,0,'STOP') //All keypad finished downloading
```

19.MultipleAssess

19.1 Attribute description

Attribute Name	Type	Parameter Description	Note
BaseConnection	Object		nitialize the BaseConnection object before setting attribute, and then call a method after setting attribute.

Mode	int	= 0 name mode（real-name registration） = 1 number mode（Anonymity）	
DisplayFormat	int	= 0 Y axis represents objects to be assessed; X axis represents indexes = 1 Y axis represents indexes; X axis represents objects to be assessed = 2 Y axis represents objects to be assessed using partial indexes = 3 X axis represents objects to be assessed using partial indexes.	
ModifyMode	int	= 0 not allow to modify = 1 modified	
SecrecyMode	int	= 0 show on keypad = 1 hide on keypad	
LessMode	int	= 0 less mode activated = 1 less mode not activated	
NumberBegin	int	Start value of assessment item ID number which cannot exceeds the number of downloaded items.	
NumberEnd	int	End value of assessment item ID number which cannot exceed the number of downloaded items.	
RuleBegin	int	Should not exceed the number of downloaded rules.	Valid when Mode>1
RuleEnd	int	Should not exceed the number of downloaded rules.	Valid when Mode>1
EnterMove	int	=0 no move =1 automatic downward =2 automatic right	
AvoidMode	int	=0 not activated =1 activated	Valid when Mode=1, Notice: call a download method in download objects on avoid lists to download avoid lists.
ColWidthFirst	int	In units of one English character	
ColWidthOther	int	In units of one English character	
DownloadSuccessKeyIDs	string	= '-2': no valid downloadis is activated. = '-1': BaseConnection is not set. = '0': all downloads are failed. = String represents keypad that is successfully downloaded.	READ ONLY, automatically set up when download stops.

DownloadErrorKeyIDs	string	= '-2': no valid download is activated. = '-1': BaseConnection is not set. =' 0': all downloads are successful. = string represents keypad failed to download.	READ ONLY, automatically set up when download stops.
StartMode	int	= 0 continue = 1 clear to restart = 2 restart & resume	The attributes must set the same as the previous ones.

Sample Code

```
MultipleAssess multipleAssess = new SunVote.MultipleAssess();

BaseConnection baseConn = new SunVote.BaseConnection();

multipleAssess.BaseConnection = baseConn;
```

19.2 Method

19.2.1 Start vote

Method	string Start()
Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully
	When tthe identification mode in additional configuration of base station is Keypad ID, It will generate KeyStatus events.
	When the identification mode in additional configuration of base station is Keypad SN, it will generate KeyStatusSN events.

Sample Code

```
multipleAssess.Mode = 0; //real-name registration with name

multipleAssess.DisplayFormat = 0; //assessment objects as Y axis, indexes as X axis

multipleAssess.NumberBegin = 1; //start number

multipleAssess.NumberEnd = 6; //end number

multipleAssess.EnterMove = 1; //press enter automatic downward

multipleAssess.LessMode = 1; //LessMode not activated

multipleAssess.ColWidthFirst = 10; //width of first line is 10
```

```
multipleAssess.ColWidthOther = 14; //width of other lines is 14

multipleAssess.StartMode = 1; //restart

multipleAssess.SecrecyMode = 1; //secrecy

multipleAssess.ModifyMode = 1;//modification allowed

multipleAssess.AvoidMode = 1; //Avoid Mode activated, avoid lists are required.

multipleAssess.Start(); //assessment starts
```

19.2.2 Stop vote

Method	string Stop()
Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully

Sample Code

```
multipleAssess.Stop(); //stop vote
```

19.2.3 Start to download items based on keypad ID

Method	string StartDownloadItems(string KeyIDs, object DataBuf)
Parameter	KeyIDs: keypad ID to download
	DataBuf:data downloaded, one-dimensional array of strings DataBuf(M) M:0-X, X:X:if the length of item name is less than 7, characters, around 4096 items can be saved. If the average length of item name is less than 32 characters, around 1000 items can be saved. For example: DataBuf(1), number of time is 2 DataBuf(0)= "item1"
Return Value	String ='-1', the attribute of BaseConnection is not set. ='0', start successfully = '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate DataDownload event

Sample Code

```
string[] lArray = new string[2] {"item 1","item 2"};

multipleAssess.StartDownloadItems('1-10', lArray);
```

19.2.4 Start download assessment rules

Method	string StartDownloadAssessRules (string KeyIDs, object DataBuf)
Parameter	KeyIDs: keypad ID to be downloaded
	DataBuf: downloaed data, two-dimensional array of strings DataBuf(M,N) . M: number of item name (1 - X)。X:about 55 pieces of index name can be savedwhen the lengh of index name is less than 7 characters. N:2(0=rule type，1=rule ID，2=character string of item name) ，see following list. For example: DataBuf(1,2) DataBuf(0,0)=”3”: user-defined assessment mode. DataBuf(0,1)=”1”: user-defined assessment rule No. 1 DataBuf(0,2)=”rule 1”: rule name. DataBuf(1,0)=”4”: user-defined assessment mod. DataBuf(1,1)=”2”: user-defined assessment rule No.2. DataBuf(1,2)=”rule 2” : rule name.
Return Value	String =’-1’, the attribute of BaseConnection is not set. =’0’, start successfully = ‘-2’: invalid keypad ID to be downloaded = ‘-3’, invalid downloaded item or exceed the maximum value
	Will generate DataDownloadevent

Sample Code

```
string[,] lArray = new string[2, 3] {{ "3", "1","business capability"},{ "4", "1","working attitude"}};

multipleAssess.StartDownloadAssessRules('1-10', lArray);
```

19.2.5 Start to download score rules based on keypad ID

Method	string StartDownloadScoreRules (string KeyIDs, object DataBuf)
Parameter	KeyIDs: keypad ID to download
	downloaded data, Two-dimensional character array DataBuf(M， N)。 M:0-7， the number of scoring rules (1—8)。 N:5(0=unit， 1= lower limit of score， 2= upper limit of score， 3=default value，4=integer place， 5= decimal place)。

	Value can be inputted no more than 8 places ， decimal point is deemed to one place。 Unit:no more than 2 chinese charater Lower limit of score:minimum value of digital input, showing ” -1" when it is not used. Upper limit of score:maxmum value of digital input , showing ” -1" when it is not used. Default value:giving defaut value automatically when no value is inputted, showing ” -1" when it is not used. Integer place:demical place of digital input’ Decimal place:decimal place of digital input。 e.g. DataBuf(1,5)the number of scoring rules is 2 DataBuf(0,0)=“”: The unit is empty。 DataBuf(0,1)=“0”: The lower limit of score is 0。 DataBuf(0,2)=“100”: The upper limit of score is 100 DataBuf(0,3)=“0”: The default value is 0。 DataBuf(0,4)=“3”:The Integer number is 3。 DataBuf(0,5)=“0”: The decimal number is 0. DataBuf(1,0)=“score”:The unit is score。 DataBuf(1,1)=“0”: The lower limit of score is 0。 DataBuf(1,2)=“10”: The upper limit of score is 10 DataBuf(1,3)=“0”: The default value is 0. DataBuf(1,4)=“2”: The Integer number is 3. DataBuf(1,5)=“1”: The decimal number is 1.
Return value	String =’-1’, the attribute of BaseConnection is not set. =’0’, start successfully = ‘-2’: invalid keypad ID to be downloaded = ‘-3’, invalid downloaded item or exceed the maximum value
	Will generate DataDownload event

Sample Code

```
string[,] lArray = new string[1, 6] {{"Score","0","10","0","2","1"}};

batchScore.StartDownloadRules('1-10', lArray);
```

19.2.6 Start download evaluation rules

Method	string StartDownloadEvaluationRules (string KeyIDs, object DataBuf)
Parameter	KeyIDs: keypad ID to be downloaded
	DataBuf: data to be downloaded, two-dimensional array of strings DataBuf(M,N） M:0 (one user-defined rule). N:1(0=default value， 1=name of level)。 Default value: 0—Y(rank)， 0 represents null Level name: “level 1 name, …， level Y name”. The maximum number of level is 8, 4 character in maximum in the name. For example: DataBuf(0,1) the number of rule is 1 DataBuf(0,0)=”1”: default value is excellent DataBuf(0,1)=”excellent, good, medium, poor”
Return Value	String =’-1’, the attribute of BaseConnection is not set. =’0’, start successfully = ‘-2’: invalid keypad ID to be downloaded = ‘-3’, invalid downloaded item or exceed the maximum value
	Will generate DataDownloadevent

Sample Code

```
string[,] lArray = new string[2, 2] { {"3"," excellent, good, general, poor " },{"3"," excellent, qualified, general, unqualified "} };

multipleAssess.StartDownloadEvaluationRules('1-10', lArray);
```

19.2.7 Stop to download

Applied only when stop manually. Generally, it stops automatically when download ends, and no need to call this method.

Method	StopDownload ()
Parameter	Null
Return Value	successful and failed download keypad attribute will be set up when it stops

Sample Code

```
multipleAssess.StopDownload(); //stop to vote
```

19.3 Event

19.3.1 Keypad status in keypad ID

When the identification mode in additional configuration of base station is keypad ID, it will return this event.

Event	KeyStatus(string BaseTag, int KeyID, int CommitOK, string KeyValue)
Parameter	**BaseTag:** base station tag
	KeyID: keypad ID
	CommitOK: whether to submit the voting data by pressing "OK" key on the keypad = 0: without pressing "OK" key = 1: pressed "OK" key
	KeyValue: key value of keypad

Sample Code

```
multipleAssess.KeyStatus += new
IMultipleAssessEvents_KeyStatusEventHandler(multipleAssess_KeyStatus);

public void multipleAssess_KeyStatus(string BaseTag, int KeyID, int CommitOK, string KeyValue)

{

    string str = string.Format("BaseTag:{0},KeyID:{1},CommitOK:{0},KeyValue:{1}", BaseTag, KeyID,
CommitOK, KeyValue);

    MessageBox.Show(str);

}
```

19.2.2 Download data on the keypad

Generated when download method (ID) is called. Download ends when DataDownload(0,0,'STOP') appears. The attribute of keypads that successfully and unsuccessfully downloaded are available to read.

Event	DataDownload(int KeyID, int DownloadStatus, string DownloadInfo)
Parameter	**KeyID: keypad ID**
	DownloadStatus: download status =1. Download successfully =-1, download failure =0, download is finished

	DownloadInfo: download information (BEGIN、current page/total page、STOP)

Sample Code

```
DataDownload (2,1, '1/4');//keypad No.2 successfully downloaded page1

DataDownload (3,-1, '1/4');//keypad No.3 faild to download page1

DataDownload (2,1, 'STOP') //keypad No.2 finished downloading
```

20. Election

20.1 Attribute description

Attribute Name	Type	Parameter Description	Note
BaseConnection	Object		Initialize the BaseConnection object before setting attribute, and then call a method after setting attribute.
Mode	int	= 0 all election ID = 1 specify election ID = 2 limit election ID	
UIMode	int	= 0 use check mode = 1 use approval, disapproval mode = 2 use accumulative voting = 3 null = 4 ID number = 5 recommend mode1: input name only = 6 recommend mode2: input name and note	
ModifyMode	int	= 0 not allowed to modify = 1 modified	
SecrecyMode	int	= 0 show on keypad = 1 hide on keypad	
LessMode	int	= 0 within upper limit = 1 equal to upper limit	
ItemStartID	int	Start value of election item ID number which cannot exceed the number of downloaded items	Valid when Mode≠0
ItemEndID	int	End value of election item ID number which cannot exceed the number of downloaded items	Valide when Mode=1
NumberOptions	int		Valide when UIMode=4
Selects	int		
OthersMode	int	=0 not allowed	Valid when UIMode≠4

		=1 allowed（choose from name list） =2 allowed（additional input）	
EquityMode	int	=0 plus minus 1 =1 plus minus 1% =2 plus minus 0.1%	Valid when UIMode =2
LimitItemNumber	string	For example, 1-3,5 represent item No.1,2,3 and 5	
DownloadSuccessKeyIDs	string	= '-2': no valid downloadis is activated. = '-1': BaseConnection is not set. = '0': all downloads are failed. = String represents keypad that is successfully downloaded.	READ ONLY, automatically set up when download stops
DownloadErrorKeyIDs	string	= '-2': no valid downloading is activated. = '-1': the attribute of BaseConnection is not set. =' 0': all downloads successful = string keypad failed to download	READ ONLY, automatically set up when download stops
DownloadSuccessKeySNs	string	= '-2': no valid downloading is activated. = '-1': the attribute of BaseConnection is not set. = '0': all downloads failed = String represents keypad that is successfully downloaded	READ ONLY, automatically set up when download stops
DownloadErrorKeySNs	string	= '-2': no valid download is activated. = '-1': BaseConnection is not set. =' 0': all downloads are successful. = string represents keypad failed to download.	READ ONLY, automatically set up when download stops
StartMode	int	= 0 continue = 1 clear to restart = 2 restart & resume	The attributes must set the same as the previous ones.

Sample Code

```
Election election = new SunVote.Election();

BaseConnection baseConn = new SunVote.BaseConnection();

election.BaseConnection = baseConn;
```

20.2 Method

20.2.1 Start vote

Method	string Start()

Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully
	When tthe identification mode in additional configuration of base station is Keypad ID, It will generate KeyStatus events.
	When the identification mode in additional configuration of base station is Keypad SN, it will generate KeyStatusSN events.
	Elector event will be generated with input of additional name.
	Referrer event will be generated with input of additionoal name and note

Sample Code

```
election.Mode = 1; //specify elevtion ID number
election.UIMode = 1; //approval and disapproval mode applied
election.Selects = 4; //pick up 4 people
election.OthersMode = 0;//additional candidate is not allowed
election.LessMode = 0; //less mode allowed
election.StartMode = 1; //restart
election.SecrecyMode = 1; //secrecy mode
election.ModifyMode = 0; //modification not allowed
election.ItemStartID = 1;
election.ItemEndID = 10; //limit ID number 1-10
election.Start();//election starts
```

20.2.2 Stop vote

Method	string Stop()
Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully

Sample Code

```
election.Stop(); //stop vote
```

20.2.3 Start to download items based on keypad ID

Method	string StartDownloadItems(string KeyIDs, object DataBuf)
Parameter	KeyIDs: keypad ID to download
	DataBuf: data to be downloaded, one-dimensional array of strings DataBuf(M） M:0-X, X:about 4096 items can be saved when the lengh of item name is shorter than 7 Chinese characters. About 1000 items can be saved if the average length of item is shorter than 32 Chinese character. For example: DataBuf(1),number of item is 2 DataBuf(0)= "item1" DataBuf(1)= "item2"
Return Value	String ='-1', the attribute of BaseConnection is not set. ='0', start successfully = '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate DataDownload event

Sample Code

```
string[] lArray = new string[2] {"candidate1","candidate2"};

election.StartDownloadItems('1-10', lArray);
```

20.2.4 Start to download items based on keypad SN

Method	string StartDownloadItemsSN (string KeySNs, object DataBuf)
Parameter	KeySNs: keypad SN to download
	DataBuf: data to be downloaded, one-dimensional array of strings DtaBuf(M） M:0-X, X:about 4096 items can be saved if the average length of item name is shorter than 32 Chinese characters. For example:DataBuf(1), the number of item is 2. DataBuf(0)= "item1" DataBuf(1)= "item2"
Return Value	String ='-1', the attribute of BaseConnection is not set.

	='0', start successfully = '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate DataDownloadSN event

Sample Code

```
string[] lArray = new string[2] {"candidate1","candidate2"};

election.StartDownloadItemsSN('111,222', lArray);
```

20.2.5 Start download random items based on keypad ID

Method	string StartDownloadRandomItems (string KeyIDs, object DataBuf)
Parameter	KeyIDs: keypad ID to be downloaded
	DataBuf: data to be downloaded, character array DataBuf(M） For example: DataBuf(1) random item ID is 2， meaning that item No.1 and 3 are selected to be voted DataBuf(0)= "1" DataBuf(1)= "3"
Return Value	String ='-1', the attribute of BaseConnection is not set. ='0', start successfully = '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate DataDownloadevent

Sample Code

```
string[] lArray = new string[2] {"1","3"};

election.StartDownloadRandomItems('1-10', lArray);
```

20.2.6 Start download random items based on keypad SN

Method	string StartDownloadRandomItemsSN (string KeySNs, object DataBuf)
Parameter	KeySNs: keypad SN to be downloaded
	DataBuf: data to be downloaded, character array DataBuf(M） For example: DataBuf(1) random item ID is 2， meaning that item No.1 and 3 are selected to be voted DataBuf(0)= "1"

	DataBuf(1)= "3"
Return Value	String ='-1', the attribute of BaseConnection is not set. ='0', start successfully = '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate DataDownloadSNevent

Sample Code

```
string[] lArray = new string[2] {"1","3"};

election.StartDownloadRandomItemsSN('1-10', lArray);
```

20.2.7 Start download ballots based on keypad ID

Method	string StartDownloadBallots(string KeyIDs, string Ballots)
Parameter	KeyIDs: keypad ID to be downloaded
	Ballots: accumulative number of votes for accordingly keypads, separated by"," For example "10,20,30"
Return Value	String ='-1', the attribute of BaseConnection is not set. ='0', start successfully = '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate BallotesDownload event

Sample Code

```
election.StartDownloadBallots("1-10", "10,20,30");
```

20.2.8 Start download ballots based on keypad SN

Method	string StartDownloadBallotsSN(string KeySNs, string Ballots)
Parameter	KeySNs: keypad SN to be downloaded
	Ballots: accumulative number of votes for accordingly keypads, separated by"," For example "10,20,30"
Return Value	String ='-1', the attribute of BaseConnection is not set. ='0', start successfully

	= '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate BallotesDownloadSN event

Sample Code

```
election.StartDownloadBallotsSN("111,222", "10,20,30");
```

20.2.9 Stop to download

Applied only when stop manually. Generally, it stops automatically when download ends, and no need to call this method.

Method	StopDownload ()
Parameter	Null
Return Value	successful and failed download keypad attribute will be set up when it stops

Sample Code

```
election.StopDownload(); //STOP TO DOWNLOAD
```

20.3 Event

20.3.1 Keypad status in keypad ID

When the identification mode in additional configuration of base station is keypad ID, it will return this event.

Event	KeyStatus(string BaseTag, int KeyID, int CommitOK, string KeyValue)
Parameter	**BaseTag:** base station tag
	KeyID: keypad ID
	CommitOK: whether to submit the voting data by pressing "OK" key on the keypad = 0: without pressing "OK" key = 1: pressed "OK" key
	KeyValue: key value of keypad

Sample Code

```
election.KeyStatus += new IElectionEvents_KeyStatusEventHandler(election_KeyStatus);

public void election_KeyStatus(string BaseTag, int KeyID, int CommitOK, string KeyValue)

{
```

```
        string str = string.Format("BaseTag:{0},KeyID:{1},CommitOK:{2},KeyValue:{3}",
            BaseTag, KeyID, CommitOK, KeyValue);
        MessageBox.Show(str);
    }
```

20.3.2 Keypad status in keypad SN

When the identification mode in additional configuration of base station is keypad SN, it will return this event.

Event	KeyStatusSN(string BaseTag, string KeySN, int CommitOK, string KeyValue)
Parameter	**BaseTag:** base station tag
	KeySN keypad SN
	CommitOK: whethe to submit the voting data by pressing "OK" key on the keypad = 0: without pressing "OK" key = 1: pressed "OK" key
	KeyValue: key value of keypad

Sample Code

```
election.KeyStatusSN += new IElectionEvents_KeyStatusSNEventHandler(election_KeyStatusSN);
public void election_KeyStatusSN(string BaseTag, string KeySN, int CommitOK, string KeyValue)
{
    string str = string.Format("BaseTag:{0},KeySN:{1},CommitOK:{2},KeyValue:{3}",
        BaseTag, KeySN, CommitOK, KeyValue);
    MessageBox.Show(str);
}
```

20.3.3 Download data on the keypad based on keypad ID

Generated when downloading starts after a user-defined rule method (ID) is called.

Event	DataDownload(int KeyID, int DownloadStatus, string DownloadInfo)
Parameter	**KeyID: keypad ID**
	DownloadStatus: download status =1. Download successfully =-1, download failure =0, download is finished
	DownloadInfo: download information (BEGIN、current page/total page、STOP)

Sample Code

```
DataDownload (2,1, '1/4');// keypad No.2 successfully downloaded page1

DataDownload (3,-1, '1/4');// keypad No.3 faild to download page1

DataDownload (2,1, 'STOP') // keypad No.2 finished downloading

DataDownload (0,0, 'STOP') //all keypads finished downloading
```

20.3.4 Download data on the keypad based on keypad SN

Generated when downloading starts after a user-defined rule method (SN) is called.

Event	DataDownloadSN(string KeySN, int DownloadStatus, string DownloadInfo)
Parameter	**KeySN: keypad SN**
	DownloadStatus: download status =1. Download successfully =-1, download failure =0, download is finished
	DownloadInfo: download information (BEGIN、current page/total page、STOP)

Sample Code

```
DataDownloadSN('111',1,'1/4');//keypad SN'111' successfully downloaded page1.

DataDownloadSN('112',-1,'1/4');// keypad SN'112' failed to download page1

DataDownloadSN('113',1,'STOP') // keypad SN'113' downloaded successfully.

DataDownloadSN(0,0,'STOP') // all keypads finished downloading
```

20.3.5 Download ballots based on keypad ID

When the identification mode in additional configuration of base station is keypad ID, it will return this event.

Event	BallotesDownload(int KeyID, int DownloadStatus, string DownloadInfo)
Parameter	**KeyID: keypad ID**
	DownloadStatus: download status =1. Download successfully =-1, download failure =0, download is finished
	DownloadInfo: download information (BEGIN、current page/total page、STOP)

Sample Code

```
BallotesDownload (2,1, '1/4');// keypad No.2 successfully downloaded page1

BallotesDownload (3,-1, '1/4');// keypad No.3 faild to download page1
```

```
BallotesDownload (2,1, 'STOP') // keypad No.2 finished downloading

BallotesDownload (0,0, 'STOP') // all keypads finished downloading
```

20.3.6 Download ballots based on keypad SN

When the identification mode in additional configuration of base station is keypad SN, it will return this event.

Event	BallotesDownloadSN(string KeySN, int DownloadStatus, string DownloadInfo)
Parameter	**KeySN: keypad SN**
	DownloadStatus: download status =1. Download successfully =-1, download failure =0, download is finished
	DownloadInfo: download information (BEGIN、current page/total page、STOP)

Sample Code

```
BallotesDownloadSN('111',1,'1/4');// keypad SN'111' successfully downloaded page1.

BallotesDownloadSN('112',-1,'1/4');// keypad SN'112' failed to download page1

BallotesDownloadSN('113',1,'STOP') // keypad SN'113' downloaded successfully.

BallotesDownloadSN(0,0,'STOP') // all keypads finished downloading.
```

20.3.7 Choose other candidate

Event	Elector(string BaseTag, int KeyID, int NameID, string Name)
Parameter	**BaseTag: base station tag**
	KeyID: keypad ID
	NameID: NAME ID
	Name: name that input

Sample Code

```
election.Elector += new IElectionEvents_ElectorEventHandler(election_Elector);

public void election_Elector(string BaseTag, int KeyID, int NameID, string Name)

{

    string str = string.Format("BaseTag:{0},KeyID:{1},NameID:{2},Name:{3}",

            BaseTag, KeyID, NameID, Name);

    MessageBox.Show(str);

}
```

20.3.8 Choose other candidate&note

Event	Referrer(string BaseTag, int KeyID, int CommitOK, int NameID, string Name, string Memo)
Parameter	**BaseTag: base station tag**
	KeyID: keypad ID
	CommitOK: whether to submit
	NameID: NAME ID
	Name: name that input
	Memo: note

Sample Code

```
election.Referrer += new IElectionEvents_ReferrerEventHandler(election_Referrer);

public void election_Referrer(string BaseTag, int KeyID, int CommitOK, int NameID, string Name, string Memo)

{

    string str = string.Format("BaseTag:{0},KeyID:{1},CommitOK:{2},NameID:{3},Name:{4},Memo:{5}",

        BaseTag, KeyID,CommitOK, NameID, Name, Memo);

    MessageBox.Show(str);

}
```

21.Message

21.1 Attribute description

Attribute Name	Type	Parameter Description	Note
BaseConnection	Object		Initialize the base connection object and call the method before set the attributes.
DownloadSuccessKeyIDs	string	= '-2': not start a valid download. = '-1': The attribute of BaseConnection is not set = '0': All failed to download. = String: indicates the keypads download successful.	Read-Only attribute, auto-set when the download is stopped.
DownloadErrorKeyIDs	string	= '-2': not start a valid download. = '-1': The attribute of BaseConnection is not set =' 0': All success download.	Read-Only attribute, auto-set when the download is stopped.

		= string: indicates the keypads download failed.	
DownloadSuccessKeySNs	string	= '-2': not start a valid download. = '-1': The attribute of BaseConnection is not set = '0': All failed to download. = String: indicates the keypads download successful.	Read-Only attribute, auto-set when the download is stopped.
DownloadErrorKeySNs	string	= '-2': not start a valid download. = '-1': The attribute of BaseConnection is not set =' 0': All succeed download. = string: indicates the keypads download failed.	Read-Only attribute, auto-set when the download is stopped.

Sample Code

```
SunVote.Message message = new SunVote.Message();

BaseConnection baseConn = new SunVote.BaseConnection();

message.BaseConnection = baseConn;
```

21.2 Method

21.2.1 start to send message based on keypad ID

Method	string Start(string KeyIDs, string Text)
Parameter	KeyIDs: The keypad ID to be downloaded.
	Text: send message content
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully = '-2': the downloaded keyIDs string is invalid = '-3': invalid data
	will generate KeyStatus events.

Sample Code

```
message.Start('1-10', 'Test');
```

21.2.2 start to send message based on keypad SN

Method	string StartSN(string KeySNs, string Text)
Parameter	KeySNs: The keypad SN to be downloaded.
	Text: send message content
	String:

Return Value	= '-1': the attribute of BaseConnection is not set = '0': start successfully = '-2': the downloaded keyIDs string is invalid = '-3': invalid data
	will generate KeyStatus events.

Sample Code

```
message.StartSN('111,211', 'Test');
```

21.2.3 Stop to download

Applied only when stop manually. Generally, it stops automatically when download ends, and no need to call this method.

Method	StopDownload ()
Parameter	Null
Return Value	successful and failed download keypad attribute will be set up when it stops

Sample Code

```
message.Stop(); /STOP TO DOWNLOAD
```

21.2.4 Show fixed information on keypad based on keypad ID

Method	DisplayFixed(int KeyID, int FixedType)
Parameter	KeyIDs: LONG, keypad ID
	FixedType: Long =1 input error prompts =2 please input ASAP
Return Value	Null, will generate DisplayFixedStatus events.

Sample Code

```
message.DisplayFixed(2,1); //Prompt keypad ID 2 , input ASAP.
```

21.2.5 Show fixed information on keypad based on keypad SN

Method	DisplayFixedSN(string KeySN, int FixedType)
Parameter	KeySNs: STRING, keypad SN
	FixedType: Long =1 input error prompts =2 please input ASAP

Return Value	Null, will generate DisplayFixedStatusSN events.

Sample Code

```
message.DisplayFixed('111',1); // Prompt SN=111,input ASAP
```

21.3 Event

21.3.1 Keypad status in keypad ID

Event	KeyStatus(int KeyID, int SendStatus)
Parameter	KeyID: keypad ID
	SendStatus:Download status = 1: download successfully = -1: download failed = 0: download completed

Sample Code

```
KeyStatus(3,-1);// means the keypad 3 failed to download.

KeyStatus(2,1) // means the keypad 2 has been downloaded successfully.

KeyStatus(0,0) // means all the keypads complete download.
```

21.3.2 Keypad status in keypad SN

Event	KeyStatusSN(string KeySN, int SendStatus)
Parameter	KeySN: keypad SN
	SendStatus:Download status = 1: download successfully = -1: download failed = 0: download completed

Sample Code

```
KeyStatusSN('111',-1);// means keypad SN=111 failed to download

KeyStatusSN('112',1) // means keypad SN=112 has been downloaded successfully.
```

21.3.3 Fixed prompt message based on keypad ID

Event	DisplayFixedStatus (int KeyID, int Status)
Parameter	KeyID: keypad ID
	Status: Prompt status = 1: success = 0: not support

Sample Code

```
DisplayFixedStatus(2,1) // means keypad ID= 2 prompt success
```

21.3.4 Fixed prompt message based on keypad SN

Event	DisplayFixedStatusSN(string KeySN, int Status)
Parameter	KeySN: keypad SN
	Status: Prompt status = 1: success = 0: not support

Sample Code

```
DisplayFixedStatusSN('111',0);// means keypad SN=111 not support prompt message
```

21.3.5 Received message

Event	ReceiveMessage(int KeyID, string Text)
Parameter	KeyID: keypad ID
	Text: The message sent by keypad

Sample Code

```
ReceiveMessage(1,'test');//means keypad id= 1 send a message of "test"
```

22. Download

22.1 File Download

22.1.1 Attribute description

Attribute Name	Type	Parameter Description	Note
BaseConnection	Object		Initialize the base connection object and call the method before set the attributes.
DownloadSuccessKeyIDs	string	= '-2': not start a valid download. = '-1': The attribute of BaseConnection is not set = '0': All failed to download. = String: indicates the keypads download successful.	Read-Only attribute, auto-set when the download is stopped.
DownloadErrorKeyIDs	string	= '-2': not start a valid download. = '-1': The attribute of BaseConnection is not set =' 0': All successes download. = string: indicates the keypads download failed.	Read-Only attribute, auto-set when the download is stopped.

Sample Code

```
SunVote.FileDownload fileDownload = new SunVote.FileDownload();

BaseConnection baseConn = new SunVote.BaseConnection();

fileDownload.BaseConnection = baseConn;
```

22.1.2 Method

22.1.2.1 start download

Method	string StartDownload (string KeyIDs, string FileName)
Parameter	KeyIDs: keypad ID to download
	FileName: the path to the file to be downloaded
Return Value	String ='-1', the attribute of BaseConnection is not set. ='0', start successfully = '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate DataDownload event

Sample Code

```
fileDownload.StartDownload ('1-10', 'd:\123.txt');
```

22.1.2.2 Stop to download

Applied only when stop manually. Generally, it stops automatically when download ends, and no need to call this method.

Method	StopDownload ()
Parameter	Null
Return Value	successful and failed download keypad attribute will be set up when it stops

Sample Code

```
fileDownload.StopDownload(); //stop to download
```

22.1.3 Event

22.1.3.1 Download data on the keypad

Event	DataDownload(int KeyID, int DownloadStatus, string DownloadInfo)
Parameter	**KeyID: keypad ID**
	DownloadStatus: download status =1. Download successfully =-1, download failure =0, download is finished
	DownloadInfo: download information (BEGIN、current page/total page、STOP)

Sample Code

```
DataDownload (2,1, '1/4');//means keypad 2 has downloaded page1 data successful

DataDownload (3,-1, '1/4');//means keypad 3 has failed to download page 1 data

DataDownload (2,1, 'STOP') // means keypad 2 stopped download

DataDownload (0,0, 'STOP') //means all the keypads stopped download
```

22.2 Avoid Items Download

22.2.1 Attribute description

Attribute Name	Type	Parameter Description	Note
BaseConnection	Object		Initialize the base connection object and call the method before set the attributes.
DownloadSuccessKeyIDs	string	= '-2': not start a valid download. = '-1': The attribute of BaseConnection is not set = '0': All failed to download. = String: indicates the keypads download successful.	Read-Only attribute, auto-set when the download is stopped.
DownloadErrorKeyIDs	string	= '-2': not start a valid download. = '-1': The attribute of BaseConnection is not set =' 0': All successes download.	Read-Only attribute, auto-set when the download is stopped

		= string: indicates the keypads download failed.	
DownloadSuccessKeySNs	string	= '-2': not start a valid download. = '-1': The attribute of BaseConnection is not set = '0': All failed to download. = String: indicates the keypads download successful.	Read-Only attribute, auto-set when the download is stopped
DownloadErrorKeySNs	string	= '-2': not start a valid download. = '-1': The attribute of BaseConnection is not set =' 0': All succeed download. = string: indicates the keypads download failed.	Read-Only attribute, auto-set when the download is stopped.

Sample Code

```
SunVote.AvoidItemsDownload avoidItemsDownload = new SunVote.AvoidItemsDownload();
BaseConnection baseConn = new SunVote.BaseConnection();
avoidItemsDownload.BaseConnection = baseConn;
```

22.2.2 Method

22.2.2.1 Start download avoid item based on keypad ID

Method	string StartDownload (string KeyIDs, object DataBuf)
Parameter	KeyIDs: keypad ID to download

	DataBuf: downloaded data, two-dimensional character array DataBuf(M,N）
	M: M (avoid rule number)
	N: 1(0=the keypad need to avoid, 1=the item number avoided
	eg. DataBuf(1,1) the number of avoid rule is 2
	DataBuf(0,0)="1" // keypad 1
	DataBuf(0,1)="3,5" // you cannot vote on item 3 and item 5
	DataBuf(0,0)="2" //keypad ID=1
	DataBuf(0,1)="1" // you cannot vote on item 1

	String

Return Value	='-1', the attribute of BaseConnection is not set. ='0', start successfully = '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate DataDownload event

Sample Code

```
string[,] lArray = new string[2, 2] { {"3","3,5" },{"3","1"} };

avoidItemsDownload.StartDownload('1-10', lArray);
```

22.2.2.2 Start download avoid item based on keypad SN

Method	string StartDownloadSN(string KeySNs, object DataBuf)
Parameter	KeySNs: keypad SN to download

	DataBuf: downloaded data, two-dimensional character array DataBuf(M,N）
	M: M (avoid rule number)
	N: 1(0=the keypad need to avoid, 1=the item number avoided
	eg. DataBuf(1,1) the number of avoid rule is 2
	DataBuf(0,0)="1" // keypad 1
	DataBuf(0,1)="3,5" // you can not vote on item 3 and item 5
	DataBuf(0,0)="2" //keypad 1
	DataBuf(0,1)="1" // you can not vote on item 1

Return Value	String ='-1', the attribute of BaseConnection is not set. ='0', start successfully = '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate DataDownloadSN event

Sample Code

```
string[,] lArray = new string[2, 2] { {"3","3,5" },{"3","1"} };

avoidItemsDownload.StartDownloadSN('123456,123457,123458', lArray);
```

22.2.2.4 Stop to download

Applied only when stop manually. Generally, it stops automatically when download ends, and no need to call this method.

Method	StopDownload ()
Parameter	Null
Return Value	successful and failed download keypad attribute will be set up when it stops

Sample Code

```
avoidItemsDownload.StopDownload(); //STOP TO DOWNLOAD
```

22.2.3 Event

22.2.3.1 Download data on the keypad based on keypad ID

Generated when downloading starts after download method (ID) is called.

Event	DataDownload(int KeyID, int DownloadStatus, string DownloadInfo)
Parameter	**KeyID: keypad ID**
	DownloadStatus: download status =1. Download successfully =-1, download failure =0, download is finished
	DownloadInfo: download information (BEGIN、current page/total page、STOP)

Sample Code

```
DataDownload (2,1, '1/4');//means keypad 2 has downloaded page1 data successful

DataDownload (3,-1, '1/4');//means keypad 3 has failed to download page 1 data

DataDownload (2,1, 'STOP') // means keypad 2 stopped download

DataDownload (0,0, 'STOP') //means all the keypads stopped download
```

22.2.3.2 Download data on the keypad based on keypad SN

Generated when downloading starts after download method (SN) is called.

Event	DataDownloadSN(string KeySN, int DownloadStatus, string DownloadInfo)
Parameter	**KeySN: keypad SN**
	DownloadStatus: download status =1. Download successfully =-1, download failure =0, download is finished
	DownloadInfo: download information (BEGIN、current page/total page、STOP)

Sample Code

```
DataDownloadSN('111',1,'1/4');// means SN  "111" has downloaded page1 data successful

DataDownloadSN('112',-1,'1/4');// means SN "112" has failed to download page 1 data

DataDownloadSN('113',1,'STOP') // means SN "113" has downloaded successful

DataDownloadSN(0,0,'STOP') // means all the keypads stopped download
```

22.3 Explanation of score rule

22.3.1 Attribute description

Attribute Name	Type	Parameter Description	Note
BaseConnection	Object		Initialize the base connection object and call the method before set the attributes.
DownloadSuccessKeyIDs	string	= '-2': not start a valid download. = '-1': The attribute of BaseConnection is not set = '0': All failed to download. = String: indicates the keypads download successful.	Read-Only attribute, auto-set when the download is stopped.
DownloadErrorKeyIDs	string	= '-2': not start a valid download. = '-1': The attribute of BaseConnection is not set =' 0': All successes download. = string: indicates the keypads download failed.	Read-Only attribute, auto-set when the download is stopped.

Sample Code

```
ScoreRuleExplain scoreRuleExplain = new SunVote.ScoreRuleExplain();

BaseConnection baseConn = new SunVote.BaseConnection();

scoreRuleExplain.BaseConnection = baseConn;
```

22.3.2 Method

22.3.2.1 Start download score rules based on keypad ID

Method	string StartDownload (string KeyIDs, object DataBuf)
Parameter	KeyIDs: keypad ID to download
	DataBuf: Downloaded data, one-dimensional character array DataBuf(M）

	M: 0-X, X: if the rule description is below 7 Chinese characters, you can save about 4096 at the same time. If the rule shows an average of less than 32 Chinese characters, about 1000 can be saved. eg. DataBuf(1), the number of rules is 2. DataBuf(0)= "work ability 0-25" DataBuf(1)= "business ability 0-30"
Return Value	String ='-1', the attribute of BaseConnection is not set. ='0', start successfully = '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate DataDownload event

Sample Code

```
string[] lArray = new string[2] {"work ability 0-25","business ability 0-30"};

scoreRuleExplain.StartDownload('1-10', lArray);
```

22.3.2.2 Stop to download

Applied only when stop manually. Generally, it stops automatically when download ends, and no need to call this method.

Method	StopDownload ()
Parameter	Null
Return Value	successful and failed download keypad attribute will be set up when it stops

Sample Code

```
scoreRuleExplain.StopDownload(); //stop to download
```

22.3.3 Event

22.3.3.1 Download data to the keypad

Event	DataDownload(int KeyID, int DownloadStatus, string DownloadInfo)
Parameter	**KeyID: keypad ID**
	DownloadStatus: download status =1. Download successfully =-1, download failure =0, download is finished

	DownloadInfo: download information (BEGIN、current page/total page、STOP)

Sample Code

```
DataDownload (2,1, '1/4');// means keypad 2 has downloaded page1 data successful

DataDownload (3,-1, '1/4');// means keypad 3 has failed to download page 1 data

DataDownload (2,1, 'STOP') //means keypad 2 stopped download.

DataDownload (0,0, 'STOP') // means all the keypads stopped download
```

22. 4 Explanation of Evaluation rule

22.4.1 Attribute description

Attribute Name	Type	Parameter Description	Note
BaseConnection	Object		Initialize the base connection object and call the method before set the attributes.
DownloadSuccessKeyIDs	string	= '-2': not start a valid download. = '-1': The attribute of BaseConnection is not set = '0': All failed to download. = String: indicates the keypads download successful.	Read-Only attribute, auto-set when the download is stopped.
DownloadErrorKeyIDs	string	= '-2': not start a valid download. = '-1': The attribute of BaseConnection is not set =' 0': All successes download. = string: indicates the keypads download failed.	Read-Only attribute, auto-set when the download is stopped.

Sample Code

```
EvaluationRuleExplain evaluationRuleExplain = new SunVote.EvaluationRuleExplain();

BaseConnection baseConn = new SunVote.BaseConnection();

evaluationRuleExplain.BaseConnection = baseConn;
```

22.4.2 Method

22.4.2.1 Start to download evaluation rule

Method	string StartDownload (KeyIDs:String; DataBuf: OleVariant)
Parameter	KeyIDs: keypad ID to download
	DataBuf: Downloaded data, one-dimensional character array DataBuf(M) M: 0-X, X: if the rule description is below 7 Chinese characters, you can save about 4096 at the same time. If the rule shows an average of less than 32 Chinese characters, about 1000 can be saved. eg. DataBuf(1), the number of rules is 2. DataBuf(0)= "work ability excellent /good/general/poor " DataBuf(1)= "Business ability Good/general/poor"
Return Value	String ='-1', the attribute of BaseConnection is not set. ='0', start successfully = '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate DataDownload event

Sample Code

```
string[] lArray = new string[2] {"work ability    excellent/good/general/poor","business ability
good/general/poor"};
evaluationRuleExplain.StartDownload('1-10', lArray);
```

22.4.2.2 Stop to download

Applied only when stop manually. Generally, it stops automatically when download ends, and no need to call this method.

Method	StopDownload ()
Parameter	Null
Return Value	successful and failed download keypad attribute will be set up when it stops

Sample Code

```
evaluationRuleExplain.StopDownload(); //stop to download
```

22.4.3 Event

22.4.3.1 Download data to the keypad

Event	DataDownload(int KeyID, int DownloadStatus, string DownloadInfo)
Parameter	**KeyID: keypad ID**
	DownloadStatus: download status =1. Download successfully =-1, download failure =0, download is finished
	DownloadInfo: download information (BEGIN、current page/total page、STOP)

Sample Code

```
DataDownload (2,1, '1/4');//means keypad 2 has downloaded page 1 successful.

DataDownload (3,-1, '1/4');// means keypad 3 has failed to download page 1 data

DataDownload (2,1, 'STOP') // means keypad 2 stopped download

DataDownload (0,0, 'STOP') // means all the keypads stopped download
```

22.5 Chinese Characters self-made wordstock download

22.5.1 Attribute description

Attribute Name	Type	Parameter Description	Note
BaseConnection	Object		Initialize the base connection object and call the method before set the attributes.
DownloadSuccessKeyIDs	string	= '-2': not start a valid download. = '-1': The attribute of BaseConnection is not set = '0': All failed to download. = String: indicates the keypads download successful.	Read-Only attribute, auto-set when the download is stopped.
DownloadErrorKeyIDs	string	= '-2': not start a valid download. = '-1': The attribute of BaseConnection is not set =' 0': All successes download. = string: indicates the keypads download failed.	Read-Only attribute, auto-set when the download is stopped.

Sample Code

```
WordStock wordStock = new SunVote.WordStock();
```

```
BaseConnection baseConn = new SunVote.BaseConnection();

wordStock.BaseConnection = baseConn;
```

22.5.2 Method

22.5.2.1 Start to download items

Method	string StartDownloadItems(string KeypadMode, string KeyIDs, string Words)
Parameter	KeypadMode: the download keypad model, currently only support G20,V30
	KeyIDs: keypad ID to download
	Words: the Chinese Character to be downloaded.
Return Value	String ='-1', the attribute of BaseConnection is not set. ='0', start successfully = '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate DataDownload event

Sample Code

```
wordStock.StartDownload ('G20','1-10','旻珺玥');
```

22.5.2.2 Stop to download

Applied only when stop manually. Generally, it stops automatically when download ends, and no need to call this method.

Method	StopDownload ()
Parameter	Null
Return Value	successful and failed download keypad attribute will be set up when it stops

Sample Code

```
wordStock.StopDownload(); //stop to download
```

22.5.3 Event

22.5.3.1 Download data to the keypad

Event	DataDownload(int KeyID, int DownloadStatus, string DownloadInfo)

Parameter	**KeyID: keypad ID**
	DownloadStatus: download status =1. Download successfully =-1, download failure =0, download is finished
	DownloadInfo: download information (BEGIN、current page/total page、STOP)

Sample Code

```
DataDownload (2,1, '1/4');// means keypad 2 has downloaded page 1 data successful

DataDownload (3,-1, '1/4');// means keypad 3 has failed to download page 1 data

DataDownload (2,1, 'STOP') // means keypad 2 stopped download

DataDownload (0,0, 'STOP') // means all the keypads stopped download
```

23.Examination

23.1 Attribute description

Attribute Name	Type	Parameter Description	Note
BaseConnection	Object		Initialize the BaseConnection object before setting attribute, and then call a method after setting attribute.
ExamMode	int	= 0 the last batch of results = 1 the result of each question = 2 Not send results when stopped = 3 Specify the type of the question	
ExamNumber	int		
ExamName	string		
ReportProgress	int	= 0 not report = 1 report progress	
SpecifyQuestionTypes	string	Format: Question 1 type - question 1 option, Question 2 type - question 2 option ..., Question n type – question n option, 1 ≤ n ≤ 30 Question type: 0 = single, 1 = multiple choice Number of options: 1-8	Valid when ExamMode=3
StartMode	int	= 0 continue = 1 clear to restart = 2 restart & resume	The attributes must set the same as the previous ones.

Sample Code

```
Examination examination = new SunVote.Examination();

BaseConnection baseConn = new SunVote.BaseConnection();

examination.BaseConnection = baseConn;
```

23.2 Method

23.2.1 Start exam

Method	string Start()
Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully

Sample Code

```
examination.ExamMode = 0; //the last batch results

examination.ExamNumber = 10; // total 10 questions

examination.ReportProgress = 0; //not report progress

examination.ExamName = 'classroom examination'; //Examination name

examination.StartMode =1; //empty and re-start.

examination.Start();
```

23.2.2 Stop exam

The keypad will upload the test result after stop if it is in the last batch of result in examination mode.

Method	string Stop()
Parameter	Null
Return Value	String, ='-1', the attribute of BaseConnection is not set ='0', stop successfully

Sample Code

```
examination.Stop(); //stop exam
```

23.2.3 Quit exam

The keypad cannot submit examination result after quit it.

Method	string Quit()
Parameter	Null
Return Value	String, ='-1', the attribute of BaseConnection is not set ='0', stop successfully

Sample Code

```
examination.Quit(); // QUITE THE EXAM
```

23.2.4 Start to download tests table

Method	string StartDownload (string KeyIDs, string ExamName, object DataBuf)
Parameter	KeyIDs: he keypad ID to be downloaded
	ExamName: the examination name
	DataBuf: downloaded data, character array DataBuf(M） M: Max 14 character, support Chinese words.
Return Value	String ='-1', the attribute of BaseConnection is not set. ='0', start successfully = '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate BallotesDownload event

23.2.5 Start to download tests answers based on keypad ID

Method	string StartDownloadAnswers(string KeyIDs, int Score, int DataType, object DataBuf)
Parameter	KeyIDs: the keypad ID to be downloaded
	Score:Total score of the examination
	DataType: data type, currently only support 0-BCD format
	DataBuf: downloaded data, two-dimensional character array DataBuf(M,N） M: number of questions N:2(0=question type， 1=score value， 2=answer)。

	Question Type: =1 single choice, max 10 options with letter ABCDEFGHJ only =2 multiple choice, max 10 options, can input 14 characters at most (S50 is 10) =3 Ranking, max 10 options, can input 14 characters at most =4 Gap filling, max input 14 characters; =5 numeric, =6 Judge, 1, yes/no 2, True/false
	Score : 0-255
	Answer: right answer

	For example: DataBuf(0,3) one answer
	DataBuf(0,0)=”1” //single choice
	DataBuf(0,1)=”2” // This question 2 points
	DataBuf(0,2)='A' //Right answer is A

Return Value	String ='-1', the attribute of BaseConnection is not set. ='0', start successfully = '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate DataDownload event

Sample Code

```
string[,] lArray = new string[2, 3] {{ "1","2","A"},{"2","3","AB"}};

examination. StartDownloadAnswers('1-3', lArray);
```

23.2.6 Start to download tests answers based on keypad SN

Method	string StartDownloadAnswersSN(string KeySNs, int Score, int DataType, object DataBuf)
Parameter	KeySNs: the keypad SN to be downloaded
	Score:Total score of the examination
	DataType: data type, currently only support 0-BCD format
	DataBuf: downloaded data, two-dimensional character array DataBuf(M,N） M: number of questions N:2(0=question type， 1=score value， 2=answer)。 Question Type: =1 single choice, max 10 options with letter ABCDEFGHJ only =2 multiple choice, max 10 options, can input 14 characters at most (S50 is 10) =3 Ranking, max 10 options, can input 14 characters at most =4 Gap filling, max input 14 characters; =5 numeric, =6 Judge, 1, yes/no 2, True/false
	Score : 0-255

	Answer: right answer

	For example:
	DataBuf(0,3) one answer
	DataBuf(0,0)="1" //single choice
	DataBuf(0,1)="2" // This question 2 points
	DataBuf(0,2)='A' //Right answer is A

Return Value	String ='-1', the attribute of BaseConnection is not set. ='0', start successfully = '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate DataDownloadSN event

Sample Code

```
string[,] lArray = new string[2, 3] {{ "1","2","A"},{"2","3","AB"}};

examination.StartDownloadAnswersSN("123456,123457,123458",lArray);
```

23.2.7 Stop to download

Method	StopDownload ()
Parameter	Null
Return Value	Null, successful and failed download keypad attribute will be set up when it stops

Sample Code

```
examination.StopDownload(); //stop to download
```

22.2.8 Get exam result of single keypad

Method	GetExamResult (int KeyID)
Parameter	KeyID: Keypad ID
Return Value	Null, will generate examination result event.

Sample Code

```
examination.GetExamResult(1); // get the exam result of keypad ID=1
```

22.2.9 Get exam result of batch keypads

Method	GetExamResults(string KeyIDs)
Parameter	KeyIDs: Keypad IDs
Return Value	Null, will generate examination result event.

Sample Code

```
examination.GetExamResult('1-4'); // get the exam result of keypad ID from 1 to 4
```

23.3 Event

23.3.1 Download data on the keypad based on keypad ID

Call the download method (ID) to start the download to generate this event. DataDownload (0, 0, 'STOP') means that the download is stopped and can read the keypads attribute of these successful and failed download.

Event	DataDownload(int KeyID, int DownloadStatus, string DownloadInfo)
Parameter	**KeyID: keypad ID**
	DownloadStatus: download status =1. Download successfully =-1, download failure =0, download is finished
	DownloadInfo: download information (BEGIN、current page/total page、STOP)

Sample Code

```
DataDownload (2,1, '1/4');//means keypad 2 download page 1 data success.

DataDownload (3,-1, '1/4');// means keypad 3 failed to download page 1 data.

DataDownload (2,1, 'STOP') // means keypad 2 stopped download.

DataDownload (0,0, 'STOP') // means all the keypad stopped download.
```

23.3.2 Download data on the keypad based on keypad SN

Call the download method (SN) to start the download to produce this event. DataDownload (0, 0, 'STOP') means that the download is stopped and can read the keypads attribute of these successful and failed download.

Event	DataDownloadSN(string KeySN, int DownloadStatus, string DownloadInfo)

Parameter	**KeySN: keypad SN**
	DownloadStatus: download status =1. Download successfully =-1, download failure =0, download is finished
	DownloadInfo: download information (BEGIN、current page/total page、STOP)

Sample Code

```
DataDownloadSN('111',1,'1/4');//keypad SN'111' download page 1data success.

DataDownloadSN('112',-1,'1/4');// keypad SN'112' failed to download page 1 data.

DataDownloadSN('113',1,'STOP') // keypad SN'113' download success.

DataDownloadSN(0,0,'STOP') // means all the keypad stopped download.
```

23.3.3 Examination progress based on keypad ID

Event	ExamProgress(string BaseTag, int KeyID, int QuestionNumber, int Done, int ChangeTimes, int Seconds)
Parameter	BaseTag: Base station tag
	KeyID: keypad ID
	QuestionNumber: the number of current question
	Done:number of questions completed
	ChangeTimes: the change times of current questions
	Seconds: the total time spent on the current question, unit (second)

Sample Code

```
ExamProgress('1',1,5,2,1,150);// keypad 1 has complete 2 questions, cost 150 seconds
```

23.3.4 Examination progress based on keypad SN

Event	ExamProgressSN(string BaseTag, string KeySN, int QuestionNumber, int Done, int ChangeTimes, int Seconds)
Parameter	BaseTag: Base station tag
	KeySN: keypad SN
	QuestionNumber: the number of current question
	Done:number of questions completed
	ChangeTimes: the change times of current questions
	Seconds: the total time spent on the current question, unit (second)

Sample Code

```
ExamProgressSN('1','123',5,2,1,150);// keypad SN=123has compelted 2 questions, cost 150S.
```

23.3.5 Exam result based on keypad ID

Event	ExamResult(int KeyID, int UploadState, string ExamName, object ExamData)
Parameter	KeyID: keypad ID
	UploadState: =1:successful =0:Failed
	ExamName: Exam name
	ExamData: Exam data. One-dimensional character array ExamData (M) M: question number of the exam (1-X). For example: ExamData (3), means the number is 4 ExamData (0)= "A" ExamData (1)= "B" ExamData (2)= "12" ExamData (3)= "1234"

Sample Code

```
ExamResult(1,1,'TEST',('1','A','B'));
```

23.3.6 Exam result based on keypad SN

Event	ExamResultSN(string KeySN, int UploadState, string ExamName, object ExamData)
Parameter	KeySN: keypad SN
	UploadState: =1:successful =0:Failed
	ExamName: Exam name
	ExamData: Exam data. One-dimensional character array ExamData (M) M: question number of the exam (1-X). For example: ExamData (3), means the number is 4 ExamData (0)= "A" ExamData (1)= "B" ExamData (2)= "12" ExamData (3)= "1234"

Sample Code

```
ExamResultSN('112',1,'TEXT',('1','A','B'));
```

23.3.7 Exam result for single question based on keypad ID

Event	OneQustionResult(string BaseTag, int KeyID, int QuestionNumber, string KeyValue, double KeyTime)
Parameter	BaseTag: Base station tag
	KeyID: Keypad ID

	QuestionNumber: current question number
	KeyValue: Answer
	KeyTime: The time spent on current question

Sample Code

```
OneQustionResult('1',1,10,'A',10.3);// question 10, submitted A, cost 10.3S
```

23.3.8 Exam result for single question based on keypad sn

Event	SubmittedDataSN(string BaseTag, string KeySN, int ItemNO, string KeyValue, double KeyTime)
Parameter	BaseTag: Base station tag
	KeySN: Keypad SN
	ItemNO: current question number
	KeyValue: Answer
	KeyTime: The time spent on current question

Sample Code

```
SubmittedDataSN('1','123',10,'A',10.3);// question 10,    submitted A, cost 10.3S
```

24.Homework

24.1 Attribute description

Attribute Name	Type	Parameter Description	Note
BaseConnection	Object		Initialize the BaseConnection object before setting attribute, and then call a method after setting attribute.
SubmitMode	int	= 0 is not specified = 1 Specifies the job name = 2 Specify the job number	
HomeworkName	string		
HomeworkID	int		
StartMode	int	= 0 continue = 1 clear to restart = 2 restart & resume	The attributes must set the same as the previous ones.

Sample Code

```
Homework homework = new SunVote.Homework();

BaseConnection baseConn = new SunVote.BaseConnection();
```

```
homework.BaseConnection = baseConn;
```

24.2 Method

24.2.1 Start receive the homework

Method	string StartReceive()
Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': start successfully

Sample Code

```
homework.HomeworkName = "Chinese";

homework.SubmitMode = 2;

homework.HomeworkID = 1;

homework.StartReceive();
```

24.2.2 Stop receive the homework

Method	string StopReceive()
Parameter	Null
Return Value	String: = '-1': the attribute of BaseConnection is not set = '0': stop successfully

Sample Code

```
homework.StopReceive();
```

.24.2.3 Start to download homework

Method	string StartDownload (string KeyIDs, int HomeworkID, string HomeworkName, object DataBuf)
Parameter	KeyIDs: he keypad ID to be downloaded
	HomeworkID: The number of homework
	HomeworkName: Homework Name
	DataBuf: downloaded data, character array DataBuf(M）

	M: Max 14 character, support Chinese
Return Value	String ='-1', the attribute of BaseConnection is not set. ='0', start successfully = '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate BallotesDownload event

24.2.4 Start to download homework answers based on keypad ID

Method	string StartDownloadAnswers(string KeyIDs, int HomeworkID, int Score, int DataType, object DataBuf)
Parameter	KeyIDs: the keypad ID to be downloaded
	HomeworkID: The number of the homework
	Score: total score of the homework
	DataType: data type, only support 0-BCD format at the moment
	DataBuf:downloaded data, two-dimensional character array DataBuf(M,N） M:the number of questions N:2(0=question type, 1=score value, 2=answer). Question type: =1 single choice, max 10 options with letter ABCDEFGHJ only =2 multiple choice, max 10 options, can input 14 characters at most (S50 is 10) =3 Ranking, max 10 options, can input 14 characters at most =4 Text, max input 14 characters; =5 numeric, =6 Judge, 1, yes/no 2, True/false
	Score : 0-255
	Answer: right answer

	For example: DataBuf(0,3) one answer
	DataBuf(0,0)="1" //single choice
	DataBuf(0,1)="2" //this question values 2 points
	DataBuf(0,2)='A' // Right answer is A

Return Value	String ='-1', the attribute of BaseConnection is not set. ='0', start successfully = '-2': invalid keypad ID to be downloaded = '-3', invalid downloaded item or exceed the maximum value
	Will generate DataDownload event

Sample Code

```
string[,] lArray = new string[2, 3] {{ "1","2","A"},{"2","3","AB"}};

homework. StartDownloadAnswers('1-3', lArray);
```

24.2.5 Start to download homework answers based on keypad SN

Method	string StartDownloadAnswersSN(string KeySNs, int HomeworkID, int Score, int DataType, object DataBuf)
Parameter	KeySNs: the keypad SN to be downloaded
	HomeworkID: The number of the homework
	Score: total score of the homework
	DataType: data type, only support 0-BCD format at the moment
	DataBuf:downloaded data, two-dimensional character array DataBuf(M,N） M:the number of questions N:2(0=question type, 1=score value, 2=answer). Question type: =1 single choice, max 10 options with letter ABCDEFGHJ only =2 multiple choice, max 10 options, can input 14 characters at most (S50 is 10) =3 Ranking, max 10 options, can input 14 characters at most =4 Text, max input 14 characters; =5 numeric, =6 Judge, 1, yes/no 2, True/false

	For example: DataBuf(0,3) one answer
	DataBuf(0,0)=”1” //single choice
	DataBuf(0,1)=”2” //this question values 2 points
	DataBuf(0,2)=’A’ // Right answer is A

Return Value	String =’-1’, the attribute of BaseConnection is not set. =’0’, start successfully = ‘-2’: invalid keypad ID to be downloaded = ‘-3’, invalid downloaded item or exceed the maximum value
	Will generate DataDownloadSN event

Sample Code

```
string[,] lArray = new string[2, 3] {{ "1","2","A"},{"2","3","AB"}};

homework.StartDownloadAnswersSN('123456,123457,123458', lArray);
```

24.2.6 Stop to download

Method	StopDownload ()
Parameter	Null

Return Value	Null, successful and failed download keypad attribute will be set up when it stops

Sample Code

```
homework.StopDownload(); //stop to download
```

24.2.7 Get homework result of single keypad based on keypad ID

Method	GetHomeworkResultByID (int KeyID, int HomeworkID)
Parameter	KeyID: Keypad ID
	HomeworkID:the number of homework
Return Value	Null, will generate HomeworkResult event.

Sample Code

```
homework.GetHomeworkResultByID(1,1); // get number 1 homework result of keypad id=1
```

24.2.8 Get homework result of single keypad based on keypad SN

Method	GetHomeworkResultBySN(string KeySN, int HomeworkID)
Parameter	KeySN: Keypad SN
	HomeworkID:the number of homework
Return Value	Null, will generate HomeworkResultSN event.

Sample Code

```
homework.GetHomeworkResultBySN('111',1); // get Nnumber 1 homework result of keypad SN=111
```

24.3 Event

24.3.1 Download data on the keypad based on keypad ID

Call the download method (ID) to start the download to generate this event. DataDownload (0, 0, 'STOP') means that the download is stopped and can read the keypads attribute of these successful and failed download.

Event	DataDownload(int KeyID, int DownloadStatus, string DownloadInfo)
Parameter	**KeyID: keypad ID**
	DownloadStatus: download status =1. Download successfully =-1, download failure

	=0, download is finished
	DownloadInfo: download information (BEGIN、current page/total page、STOP)

Sample Code

```
DataDownload (2,1, '1/4');//means keypad 2 download page 1 data succeed

DataDownload (3,-1, '1/4');//means keypad 3 failed to download page 1 data

DataDownload (2,1, 'STOP') // means keypad 2 stopped download

DataDownload (0,0, 'STOP') //means all the keypads stopped download.
```

24.3.2 Download data on the keypad based on keypad SN

Call the download method (SN) to start the download to produce this event. DataDownload (0, 0, 'STOP') means that the download is stopped and can read the keypads attribute of these successful and failed download.

Event	DataDownloadSN(string KeySN, int DownloadStatus, string DownloadInfo)
Parameter	**KeySN: keypad SN**
	DownloadStatus: download status =1. Download successfully =-1, download failure =0, download is finished
	DownloadInfo: download information (BEGIN、current page/total page、STOP)

Sample Code

```
DataDownloadSN('111',1,'1/4');//keypad SN'111' succeed to download page 1 data

DataDownloadSN('112',-1,'1/4');// keypad SN'112' failed to download page 1 data.

DataDownloadSN('113',1,'STOP') // keypad SN'113' succeed to download.

DataDownloadSN(0,0,'STOP') //means all the keypad stopped download.
```

24.3.3 Homework result based on keypad ID

Event	HomeworkResult(int KeyID, int UploadState, string HomeworkName, object HomeworkData)
Parameter	KeyID: keypad ID
	UploadState: =1:successful =0:Failed
	HomeworkName: the name of the homework
	HomeworkData: Homework data, one-dimensional array HomeworkData (M） M: the number of questions of homework For example: HomeworkData (3),means the number is 4.

	HomeworkData (0)= "A" HomeworkData (1)= "B" HomeworkData (2)= "12" HomeworkData (3)= "1234"

Sample Code

```
HomeworkResult(1,1,'homework',('1','A','B'));
```

24.3.4 Homework result based on keypad SN

Event	HomeworkResultSN(string KeySN, int UploadState, string HomeworkName, object HomeworkData)
Parameter	KeySN: keypad SN
	UploadState: =1:successful =0:Failed
	HomeworkName: the name of the homework
	HomeworkData: Homework data, one-dimensional array HomeworkData (M） M: the number of questions of homework For example: HomeworkData (3),means the number is 4. HomeworkData (0)= "A" HomeworkData (1)= "B" HomeworkData (2)= "12" HomeworkData (3)= "1234"

Sample Code

```
HomeworkResultSN('112',1,'hoemwork',('1','A','B'));
```

25.Self Exercise

25.1 Attribute description

Attribute Name	Type	Parameter Description	Note
BaseConnection	Object		Initialize the BaseConnection object before setting attribute, and then call a method after setting attribute.

Sample Code

```
SelfExercise selfExercise = new SunVote.SelfExercise();

BaseConnection baseConn = new SunVote.BaseConnection();

selfExercise.BaseConnection = baseConn;
```

25.2 Method

25.2.1 Start to download exercise question list

Method	string StartDownload (string KeyIDs, int ExerciseID, string ExerciseName, object DataBuf)
Parameter	KeyIDs: he keypad ID to be downloaded
	ExerciseID: The number of the exerciser
	ExerciseName: exercise name
	DataBuf: downloaded data, two-dimensional character array DataBuf(M,N ）. M: the number of the exerciser (1 - X). X: max to 100 N:3(0=question type, 1=parameter, 2=score value, 3=right answer), please refer below for details. **For example:** DataBuf(2,3), means 3 questions DataBuf(0,0)= ”1”:question type= judge DataBuf(0,1)=”2” :Display Yes/No ”1”= True/False ”2”= Yes/No DataBuf(0,2)= ”1”: score =1 DataBuf(0,3)= ”A”:right answer =A DataBuf(0,0)= ”2”:question type= single choice DataBuf(0,1)= ”3” :options=3(max 8) DataBuf(0,2)= ”1”: score =1 DataBuf(0,3)= ”B”:right answer=B DataBuf(0,0)= ”3”: question type=multiple choice DataBuf(0,1)= ”4” : options =3(max 8) DataBuf(0,2)= ”1”: score=1 DataBuf(0,3)= ”ABC”:right answer =ABC。
Return Value	String =’-1’, the attribute of BaseConnection is not set. =’0’, start successfully = ‘-2’: invalid keypad ID to be downloaded = ‘-3’, invalid downloaded item or exceed the maximum value
	Will generate BallotesDownload event

Sample Code

```
string[,] lArray = new string[1, 4] {{"3","4","1","ABC"}};

selfExercise.StartDownload('1-10',1,'exercise1',lArray);
```

25.2.3 Stop to download

Method	StopDownload ()
Parameter	Null
Return Value	Null, successful and failed download keypad attribute will be set up when it stops

Sample Code

```
selfExercise.StopDownload(); //stop to download
```

25.3 Event

25.3.1 Download data on the keypad based

Call the download method (ID) to start the download to generate this event. DataDownload (0, 0, 'STOP') means that the download is stopped and can read the keypads attribute of these successful and failed download.

Event	DataDownload(int KeyID, int DownloadStatus, string DownloadInfo)
Parameter	**KeyID: keypad ID**
	DownloadStatus: download status =1. Download successfully =-1, download failure =0, download is finished
	DownloadInfo: download information (BEGIN、current page/total page、STOP)

Sample Code

```
DataDownload (3,-1, '1/4');//means keypad 3 failed to downloadpage 1 data

DataDownload (2,1, 'STOP') //means keypad 2 stopped download.

DataDownload (0,0, 'STOP') //means all the keypad stopped download.
```

26.Appendix