弹框插件使用说明

1. 分类

弹框分为对话类窗口和提示类窗口，首先对话框类窗口是支持输入与点击确认的一类交互窗口，目前对话框支持标签显示、编辑框输入、下拉框输入、按钮点击、时间日期控件的时间选择等，以上控件加入对话框后，使用者可以获取对话框上显示的信息来配合点击事件的校验等；另外一个弹框就是提示类的弹框，包括确认框、提示框与警示框，其中确认框包含确认与取消按钮，在点击确认与取消后可以完成不同的操作，其他两类只有确认按钮，就是提示类框口，不需要在点击确认后执行后续操作。

|  |  |  |
| --- | --- | --- |
| **弹框** | **提示类弹框** | 确认框（确认与取消） |
| 提示框（确认） |
| 警告框（确认，红色警示图标） |
| **对话类弹框** | 对话框（确认与取消，输入显示信息） |

1. 插件代码编译

插件是qt5.6.2编译生成的，用qt创建动态库的方法生成的工程。

插件代码工程目录TipBoxManage的pro文件目录有如下目录：

CONFIG(debug,debug|release) {

DESTDIR = $$PWD/../../bin/

}else {

DESTDIR = $$PWD/../..bin/

}

该目录可以设置动态库生成路径，设置完毕就可以编译代码生成即可。

1. 插件调用

插件是用qt创建动态库的方法使用的，调用的时候在工程中添加外部库即可，具体步骤：右击项目名称->选择外部库->然后点击下一步->在库目录选择编译的库生成路径，在包含路径选择插件的代码目录（包含头文件的目录），平台选择只选择Linux，点击下一步即可。

1. 弹框配置文件的配置

弹框的配置主要是用于获取弹框的标题与提示信息的，PopupConfig.xml配置文件分为四层，从外到里，最外层COMMAND包含其他所有的层级，第二层是功能层级，COMMON是通用的弹框配置，Privilege/LOGIN等是功能模块；第三层是弹框类型，DialogBox是对话框配置，ConfirmBox是确认框配置，PromptBox是提示框的配置，WarningBox是警告框配置；最里层是每条弹框的配置，包括id是序号，每个弹框分类里面序号不要重复，调用的时候根据这个id来的，title是弹框标题，info是弹框要显示的提示信息，effect是附件的方便以后拓展使用，目前可以随便写。

1. 插件使用

首先使用之前以上步骤要完成，按照提示框和对话框两类讲解使用方法；

* 提示框使用方法如下：

1. 首先在使用的类中引用弹框的同文件，#include "PromptBox.h"；
2. 然后实例化，PromptBox \* tipBox = new PromptBox("COMMON","PromptBox",1,this);构造函数的参数按照弹框配置文件中的功能模块、弹框类型、序号以及使用类的this指针，PromptBox 是提示类弹框，支持确认框（ConfirmBox）、提示框（PromptBox）和警告框（WarningBox），不支持其他的；
3. tipBox->SetGeometry(0,0,249);第三个项是设置提示框的高度，提示框要显示文字比较多的话，高度就大一些，可以自己调节，前两个不用看；
4. 可以获取含有#号的提示信息，按照#分割后返回每项内容，方便后续拼接显示，QStringList infolist = tipBox->GetPromptInfo();
5. tipBox->SetInfoShow("内容");重新设置弹框显示信息；
6. 然后显示弹框，tipBox->exec() == QDialog::Accepted点击确认按钮的情况

if(tipBox->exec() == QDialog::Accepted)

{//确认

}else{//取消

}

* 对话框使用方法如下：

1. 首先在使用的类中引用弹框的同文件，#include "PopupWindow.h"；
2. 然后实例化，PopupWindow \*tip = new PopupWindow(2,"COMMON","DialogBox",1,this);构造函数的第一个参数是对话框的模式（可以设置1，2，主要时间日期控件的不同），其他参数按照弹框配置文件中的功能模块、弹框类型、序号以及使用类的this指针，PopupWindow 支持对话框的（DialogBox），不支持其他的；
3. connect(tip,SIGNAL(SendClickSig(bool)),this,SLOT(On\_RecvPopupSig(bool)))这个是获取对话框界面的确认取消按钮的信号，确认按钮是true，取消按钮是false，需要在实用类中添加一个槽函数接受对话框的SendClickSig信号；
4. tip->SetGeometry(0,0,tip->width()+200,tip->height()+300);后两个参数设置对话框的宽高的，前两个不用看；
5. tip->show();对话框显示
6. tip->AddComponent(1,"label\_test",QRect(300,200,80,40)); 添加对话框的控件，第一个参数是控件类型，1是标签，2是输入框，3是下拉框，4.是按钮，5.是时间日期选择控件；第二个参数是控件的objectname需要自己定，不要与其他控件重复；第三个参数控件的坐标，自己调一下；
7. tip->SetComponentText("label\_test","标签"); 设置控件的显示内容，第一个参数objectname，第二个参数显示内容；
8. QString cText = tip->GetComponentText("label\_test");获取控件的显示内容，参数是objectname；
9. QString info = tip->GetTitleName(2);根据索引对话框提示信息；
10. tip->AddComBoxItem("combox\_test","下拉框");为下拉框添加子项；

1. 弹框效果

