1. 新建Qt工程文件



图1 Qt工程文件框架图

2.\*.ui文件中的QWidget对象进行分辨率（geometry参数）的设置

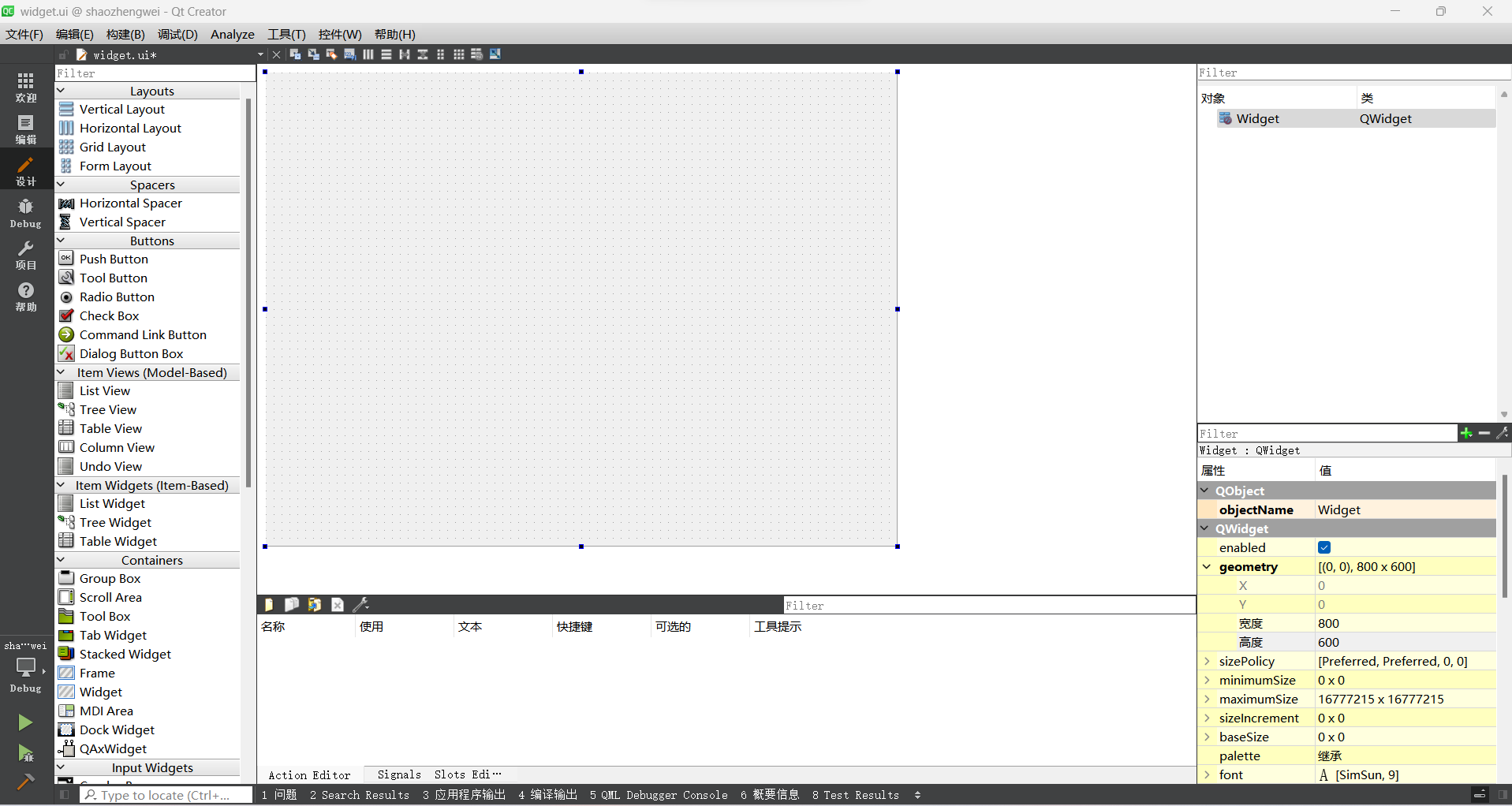


图2 QWidget几何参数设置

3.添加label和push button两个控件，并分别增加文字，修改（重命名）两个对象的名称

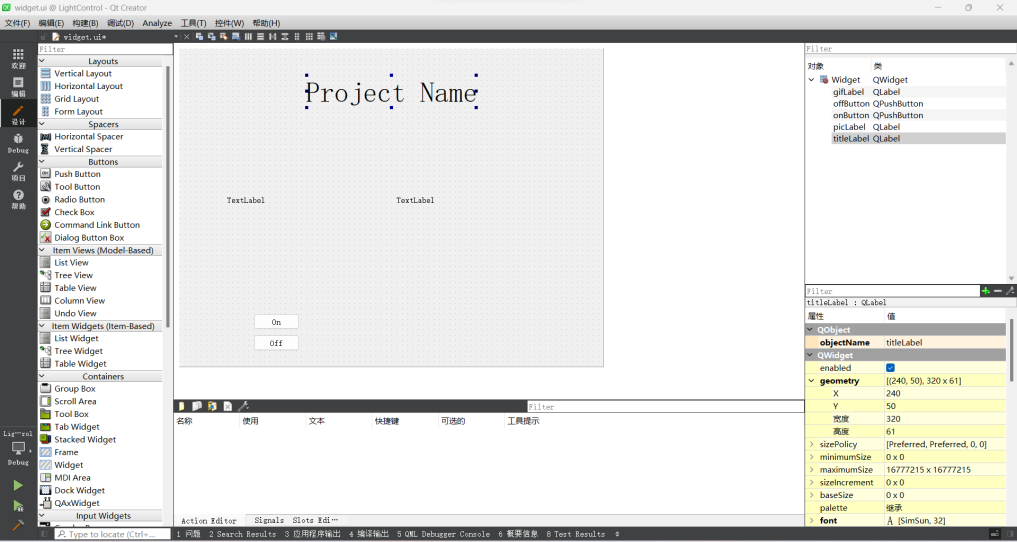


图3 添加控件并修改文字

1. 添加QPushButton对象的相应代码：

#include "widget.h"

#include "ui\_widget.h"

#include <QPixmap>

#include <QMovie>

Widget::**Widget**(QWidget \*parent)

: QWidget(parent)

, ui(new Ui::Widget)

{

ui->setupUi(this);

// label

QFont font;

font.setFamily("Noto Sans CJK SC");

font.setItalic(true);

font.setBold(false);

font.setPixelSize(32);

ui->titleLabel->setFont(font);

ui->titleLabel->setWordWrap(false);

QString string = tr("物联网灯光控制系统");

QString str = ui->titleLabel->fontMetrics().elidedText(string, Qt::ElideRight, 120);

ui->titleLabel->setText(string);

ui->picLabel->setPixmap(QPixmap(":/resources/img/lightoff.png"));

qDebug("显示off已经执行");

QMovie \*movie = new QMovie(":/resources/img/simulation.gif");

ui->gifLabel->setMovie(movie);

movie->start();

}

Widget::~***Widget***()

{

delete ui;

}

void Widget::**on\_onButton\_clicked**()

{

ui->picLabel->setPixmap(QPixmap(":/resources/img/lighton.png"));

qDebug("显示on已经执行");

}

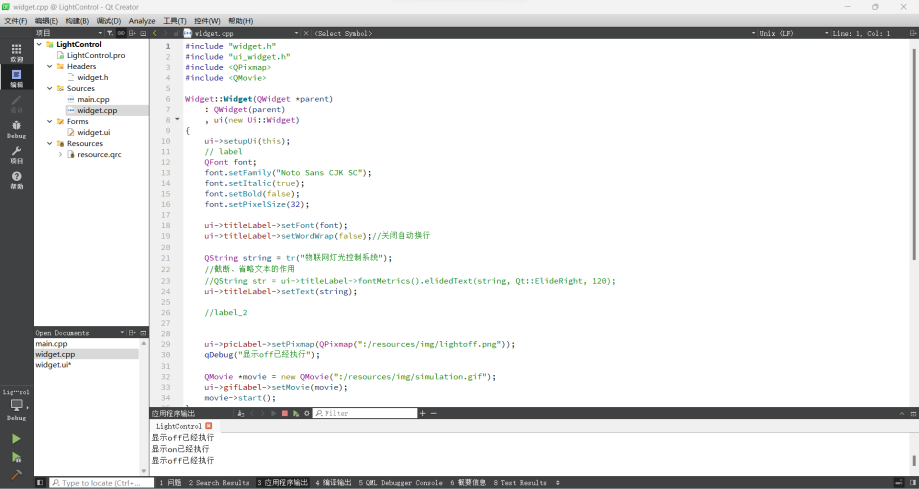
void Widget::**on\_offButton\_clicked**()

{

ui->picLabel->setPixmap(QPixmap(":/resources/img/lightoff.png"));

qDebug("显示off已经执行");

}



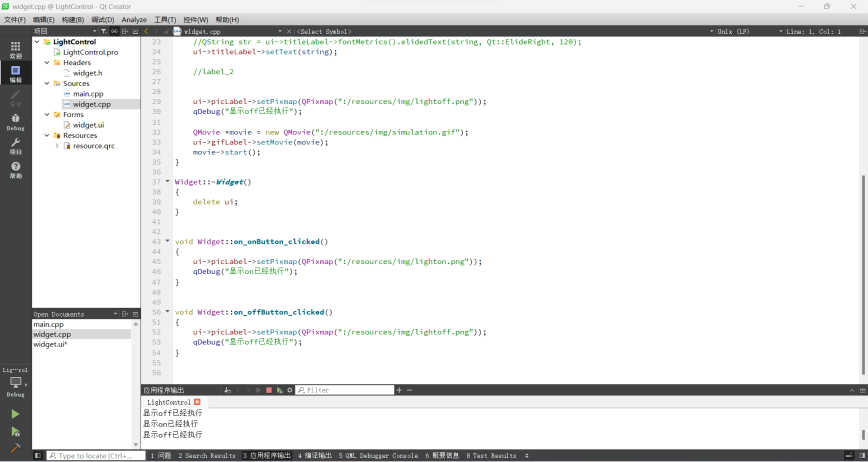


图4 Qt Creator验证测试

