

# 一、实验准备

课程主页: [课程主页\(gitee.com\)](#)

实验文档: [lab4文档](#)

实验视频: [lab4视频](#)

# 二、实验目标

- 1、掌握视频API的操作方法;
- 2、掌握如何发送随机颜色的弹幕。

# 三、实验步骤

## 1、项目创建和页面配置

基本流程见前两个lab, 在此不再赘述。

## 2、导航栏设计

编写App.js:

```
{
  "pages": [
    "pages/index/index"
  ],
  "window": {
    "navigationBarTextStyle": "white",
    "navigationBarTitleText": "实验四: 口述校史",
    "navigationBarBackgroundColor": "#24A0ED"
  },
  "style": "v2",
  "componentFramework": "glass-easel",
  "sitemapLocation": "sitemap.json",
  "lazyCodeLoading": "requiredComponents"
}
```

## 3、视图设计

编写index.wxml:

```
<!--index.wxml-->
<!-- 第一个区域: 视频播放器 -->
<video id='myvideo' class="videoClass" controls='true' src='{{src}}' enable-
danmu="true" danmu-btn="true" autoplay="true" ></video>

<!-- 第二个区域: 弹幕区域 -->
<view class="danmuArea">
  <input type='text' placeholder='请输入弹幕内容' bindinput="getDanmu" value="
{{danmuText}}"></input>
  <!-- 新版的button默认较大, 令size='mini'达到缩小按钮的效果 -->
  <button class='minibtn' size='mini' bind:tap="sendDanmu">发送弹幕</button>
```

```

</view>

<!-- 第三个区域：视频列表 -->
<view class='videoList'>
  <view class='videoBar' wx:for='{{list}}' wx:key='video{{index}}' data-
url='{{item.videoUrl}}'
  bind:tap='playVideo'>
    <image src="/images/play.png" ></image>
    <text>{{item.title}}</text>
  </view>
</view>

```

这里将video的属性添加autoplay="true"用来自动播放视频。

新版的button默认较大，将button的属性添加size='mini'达到缩小按钮的效果。

## 编写index.wxss

```

/**index.wxss**/
video{
  width: 100%;
}

.danmuArea{
  display: flex;
  flex-direction: row;
}

input{
  border: 1rpx solid #24A0ED;
  height: 100rpx;
  flex-grow: 1;
}

button{
  color: white;
  background-color: #24A0ED;
}

.minibtn{
  color: white;
  background-color: #24A0ED;
  height: 100rpx;
  text-align: center;
}

.videoList{
  width: 100%;
  min-height: 400rpx;
}

.videoBar{
  width: 95%;
  display: flex;
  flex-direction: row;
  border-bottom: 1rpx solid #24A0ED;
  margin: 10rpx;
}

```

```

image{
  width: 70rpx;
  height: 70rpx;
  margin: 20rpx;
}

text{
  font-size: 45rpx;
  color: #24A0ED;
  margin: 20rpx;
  flex-grow: 1;
}

```

### 3、逻辑实现

编写index.js:

```

// index.js
//随机获得颜色
function getRandomColor() {
  let rgb = []
  for (let i = 0; i < 3; ++i) {
    let color = Math.floor(Math.random() * 256).toString(16)
    color = color.length == 1 ? '0' + color : color
    rgb.push(color)
  }
  return '#' + rgb.join('')
}

Page({
  data: {
    src: '',
    danmuText: '',
    list: [
      {
        id: '001',
        title: '杨国宜先生口述校史实录',
        videoUrl:
'http://arch.ahnu.edu.cn/__local/6/CB/D1/C2DF3FC847F4CE2ABB67034C595_025F0082_AB
D7AE2.mp4?e=.mp4'
      },
      {
        id: '002',
        title: '唐成伦先生口述校史实录',
        videoUrl:
'http://arch.ahnu.edu.cn/__local/E/31/EB/2F368A265E6C842BB6A63EE5F97_425ABEDD_71
67F22.mp4?e=.mp4'
      },
      {
        id: '003',
        title: '倪光明先生口述校史实录',
        videoUrl:
'http://arch.ahnu.edu.cn/__local/9/DC/3B/35687573BA2145023FDAEBAFE67_AAD8D222_92
5F3FF.mp4?e=.mp4'
      }
    ]
  }
})

```

```

    },
    {
      id: '004',
      title: '吴仪兴先生口述校史实录',
      videoUrl:
        'http://arch.ahnu.edu.cn/___local/5/DA/BD/7A27865731CF2B096E90B522005_A29CB142_65
        25BCF.mp4?e=.mp4'
    }
  ]
},

playVideo: function(e) {
  console.log(e);
  this.videoCtx.stop();
  this.setData({
    src: e.currentTarget.dataset.url
  });
  this.videoCtx.play();
},

getDanmu: function(e){
  this.setData({
    danmuText: e.detail.value
  })
},

sendDanmu: function(e){
  let text = this.data.danmuText
  this.videoCtx.sendDanmu({
    text: text,
    color: getRandomColor()
  })
  // 发送弹幕之后应该清空输入框
  this.setData({
    danmuText: ''
  })
},

// 页面加载时调用
onLoad(options) {
  this.videoCtx=wx.createVideoContext('myVideo')
},

// 页面初次渲染完成时调用
onReady: function() {
  this.videoCtx = wx.createVideoContext('myVideo');
  // 自动播放第一个视频
  if (this.data.list.length > 0) {
    this.setData({
      src: this.data.list[0].videoUrl
    }, () => {
      this.videoCtx.play();
    });
  }
},

onShow() {}, // 页面显示时调用

```

```
onHide() {}, // 页面隐藏时调用

onUnload() {}, // 页面卸载时调用

onPullDownRefresh() {}, // 用户下拉动作时调用

onReachBottom() {}, // 页面上拉触底事件的处理函数

onShareAppMessage() {} // 用户点击右上角分享时调用

})
```

## 四、程序运行结果

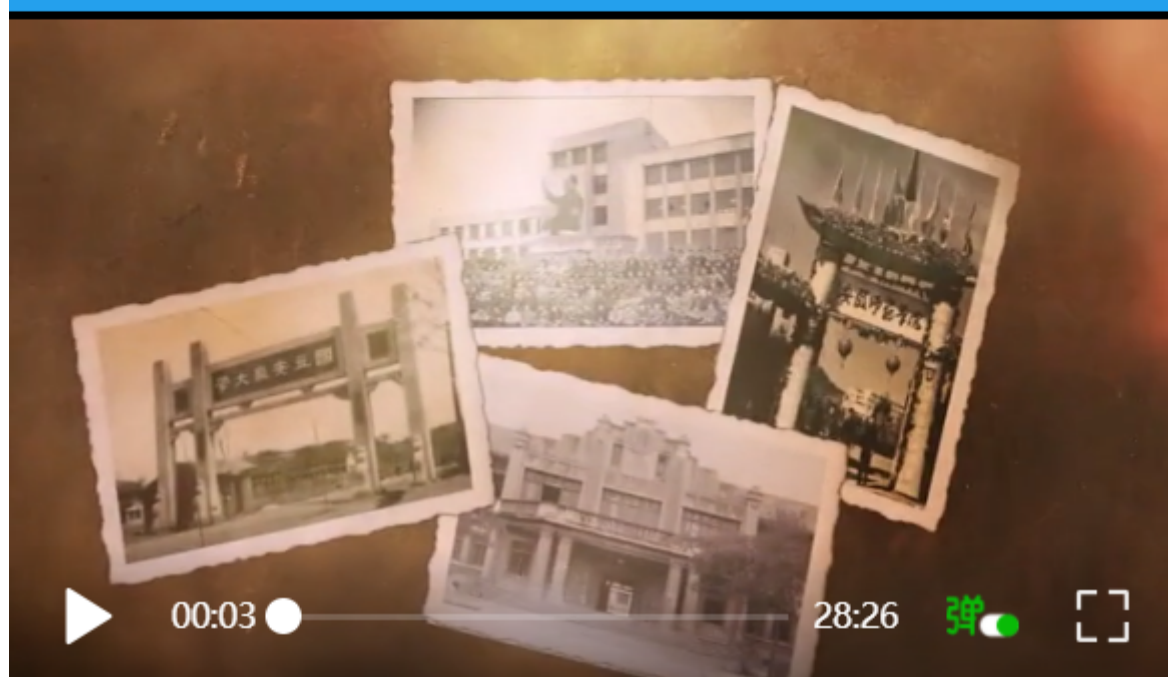
---

(1)首页

9:38

80%

## 实验四：口述校史



请输入弹幕内容

发送弹幕

▶ 杨国宜先生口述校史实录

▶ 唐成伦先生口述校史实录

▶ 倪光明先生口述校史实录

▶ 吴仪兴先生口述校史实录

(2)发送弹幕



请输入弹幕内容

发送弹幕

▶ 杨国宜先生口述校史实录

▶ 唐成伦先生口述校史实录



## 倪光明先生口述校史实录

---



## 吴仪兴先生口述校史实录

---



点击“发送弹幕”按钮之后自动清除输入框的弹幕内容。

## 五、问题总结与体会

---

- 1、问题：本次实验比较简单。可能值得注意的就是“发送弹幕的按钮”比较大，不太美观。
- 2、解决方法：查阅微信官方文档后发现button有size属性，默认为default，可以设置为mini，从而调整按钮大小。



button

基础库 1.0.0 开始支持，低版本需做兼容处理。

微信 Windows 版：支持

微信 Mac 版：支持

渲染框架支持情况：Skyline（使用最新 Nighly 工具调试）、WebView

功能描述

按钮。

通用属性

属性	类型	默认值	必填	说明						
size	string	default	否	按钮的大小						
<table><tr><td>合法值</td><td>说明</td></tr><tr><td>default</td><td>默认大小</td></tr><tr><td>mini</td><td>小尺寸</td></tr></table>					合法值	说明	default	默认大小	mini	小尺寸
合法值	说明									
default	默认大小									
mini	小尺寸									

CSDN @归零者007

3、收获与体会：

- （1）本次实验我掌握了视频API的操作方法和如何发送随机颜色的弹幕。
- （2）本次实验我还增加了遇到问题查阅官方文档的训练，从而更好地解决问题。