

## 《微信小程序综合实践》实验报告六

### 小程序实战 —— 音乐小程序在线播放与云开发

姓名 熊子宇

专业 计算机科学与技术

学号 3200105278

联系方式 18255168425

#### 一、实验目的

1. 在上一次实验的基础上，对小程序进行完善，通过本次实验得到功能完备的小程序并可发布上线；
2. 综合运用小程序所提供的多媒体组件与事件机制，实现音乐小程序核心功能；
3. 掌握父子组件、兄弟组件之间传值的方法，掌握全局数据的用法；
4. 掌握小程序状态同步方法，通过全局数据同步相同组件不同实例之间的数据；
5. 掌握云开发基本环境的配置方法，可以将普通环境增强为云开发环境；
6. 使用云开发提供的接口和云函数，对有用信息持久化存储到云数据库；
7. 熟练掌握对云数据库数据增加、修改、删除、查找等操作，通过云数据库获取数据并对组件状态进行更新；

#### 二、实验内容及要求

**准备工作：**为了方便后续的学习和实践，建议同学们自带电脑，在自己电脑上进行实验内容，并完成实验报告。本次实验为基础内容，认真完成本次实验内容即可掌握开发微信小程序的基本工具使用。

##### （一）实验内容

1. 在上一次实验的基础上，使用 `audio` 组件为音乐小程序添加歌曲播放器；
2. 将歌曲播放器固定在底部栏并在每个页面进行显示；
3. 自定义适合屏幕尺寸的播放器组件；
4. 对播放器歌曲信息进行同步，保证页面切换时当前播放信息不会更改；
5. 添加音乐详情页，能点击音乐详情页进行歌曲播放；
6. 使用云开发提供收藏歌曲功能，保存音乐收藏信息；

## （二）实验步骤（仔细阅读，按照步骤完成实验）

（本次实验所需的静态资源文件可见学在浙大上的附件）

## 三、实验感受及记录

### （一）实验感受（本次实验遇到的问题、主要收获等内容）

本次实验对我而言过于困难，花费十余个小时才大致完成。问题主要有以下几点：

1. 实验 5 和实验 6 累计写了很多文件夹和近千行代码，工程变得复杂，各组件数据传递之间逻辑令我混乱，需要 debug 很久。
2. 我对 js 的语法，尤其是小程序中的特性很不熟悉，所以经常没有办法找到自己的错误，或者无法选用合适的语句完成任务。
3. 数据库的上传和引入部分，仅凭复刻助教给出的代码，仍然无法熟悉操作，导致自己编写的时候处处出错，无法正确地上传和引入文件。
4. 编写个人收藏页面过程中，发现 songItem 组件的渲染逻辑略有问题，比如收藏了李荣浩的 1、2、3 首歌后，打开许嵩的详情页，发现他的 1、2、3 首歌也显示为被收藏（实际上没有收藏），这是由于渲染红心(favor)状态仅仅依据 sid 的缘故。由于这个逻辑错误，导致进一步编写电台详情页和个人收藏页时，可能会导致显示不正确。

通过本次实验，我主要收获了页面和组件间传递参数的方法和逻辑，设置并更新全局变量的方法，以及调用云数据库的基本方法。

### （二）实验记录（实验过程中关键步骤截图记录及文字描述）

- 2.1 完成所有「实验步骤」内容，将报告中的功能和代码添加到经过上次实验所完成一半的小程序中。将之前写在配置文件中的歌曲数据插入到数据库中，并将数据引入方式改为从数据库中引入。将相关代码片段及小程序关键页面在下方进行截图展示。

```
switch (singer) {  
    case "李荣浩":  
        this.getLiRongHao();  
        break;  
}
```

```

getLiRongHao: function() {
    console.log("getLiRongHao")
    const db = wx.cloud.database({
        env: 'cloud1-0gmvwabtf2ca9f59'
    });
    db.collection('liRongHao').get().then(
        res => {
            var n = res.data.length;
            console.log(n);
            let song = new Array(n);
            for (var i=0;i<n;i++) {
                song[i] = res.data[i];
            }
            this.setData({
                dataSource: song
            })
            console.log(this.data.dataSource);
        }
    )
},

```

|                                 |  |
|---------------------------------|--|
| _id: 0448022461b204b7016fb4...  | + 添加字段   |
| _id: 0448022461b2054b016fc61... | "_id": "0448022461b204b7016fb457566d734d"                                |
| _id: c462c81061b20584016b00...  | "_openid": "o-ilk5Uxa55bSkFW42-REsTdaTUE"                                |
|                                 | "id": 1  |
|                                 | "name": "太坦白"  |
|                                 | "poster": "https://y.gtimg.cn/music/photo_new/T002R300x300M000004AhJ..." |
|                                 | "singer": "李荣浩"  |
|                                 | "src": "https://music.163.com/song/media/outer/url?id=27731177.mp3"      |



2.2 实验报告中只对「热歌推荐」页面制作了详情页，请制作「精选电台」页面歌曲的详情页，并使歌曲有收藏功能（可复用报告中提供的 `songItem` 组件）。将相关代码片段及小程序关键页面在下方进行截图展示。

```
<!--pages/broadDetail/broadDetail.wxml-->
<view class='music-detail'>
  <view class='detail-banner' style="background-color:
  {{color}};">
    <image src="{{thumb}}" mode="aspectFill"></image>

  </view>
  <view class='detail-wrap'>
    <view wx:for='{{dataSource}}' wx:for-item="item" wx:for-
    index="index" wx:key="index">
      <songItem item="{{item}}" singer="{{name}}" />
    </view>
  </view>
</view>
<musicPlayer id="player2"/>
```

```
/* pages/broadDetail/broadDetail.wxss */
page {
  height: 100%;
}

.music-detail {
  padding-bottom: 110rpx;
}

.detail-banner {
  width: 100%;
  height: 500rpx;
  line-height: 500rpx;
  position: fixed;
```

```

    top: 0;
    z-index: -1;
  }

.detail-banner image {
  width: 100%;
  height: 500rpx;
}

.detail-banner::after{
  border-radius: 5px;
  position: absolute;
  top:0;
  left:0;
  content:'';
  width:100%;
  height:100%;
  background-color: rgba(0,0,0,.5);
  z-index: 0;
}

.detail-wrap {
  padding-top: 500rpx;
}

// pages/broadDetail/broadDetail.js
import broadSongList from "../../datas/broadSongList.js";
var app = getApp();
Page({
  /**
   * 页面的初始数据
   */
  data: {
    name: null,
    thumb: null,
    dataSource: [],
  },

  /**
   * 生命周期函数--监听页面加载
   */
  onLoad: function (options) {
    var player = this.selectComponent('#player2');
    console.log(player);
    app.globalData.musicPlayer = player;
    const {
      name,
      thumb
    } = options;
    console.log(name, thumb);
    this.setData({
      name: name,
      thumb: thumb,
    })

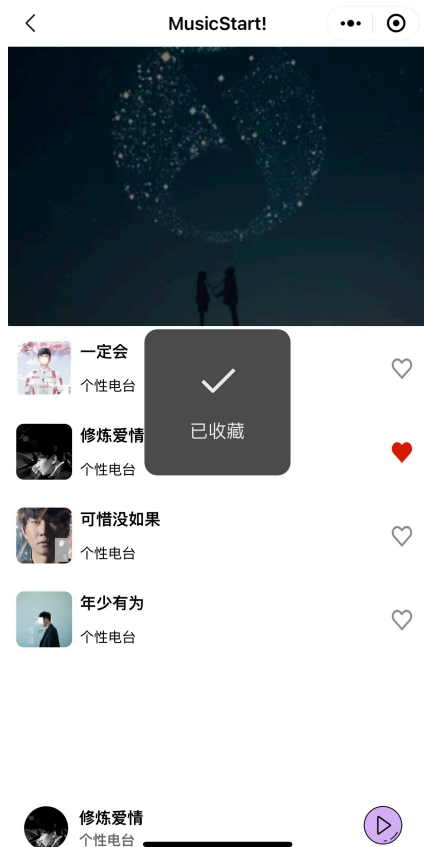
    switch (name) {
      case "个性电台":
        this.setData({
          dataSource: broadSongList.geXingDianTai
        });
    }
  }
})

```

```

        break;
      case "随心听":
        this.setData({
          dataSource: broadSongList.suiXinTing
        });
        break;
    }
  },
})

```



## 2.2 电台详情页

## 2.3 个人收藏页

2.3 在「个人中心」页面提供查看已收藏歌曲列表的功能。将相关代码片段及小程序关键页面在下方进行截图展示。

```

<!--pages/favorDetail/favorDetail.wxml-->
<view style="padding-bottom: 20px"></view>
<view class='detail-wrap'>
  <view wx:for='{{dataSource}}' wx:for-item="item" wx:for-
index="index" wx:key="index">
    <songItem item="{{item}}" singer="{{singer}}" />
  </view>
</view>

```

```
// pages/favorDetail/favorDetail.js
```

```
import songList from "../../datas/songList.js";
import broadSongList from "../../datas/broadSongList.js";

Page({
  data: {
    dataSource: [],
    singer: null,
    favor: "个人收藏"
  },
  onLoad: function (options) {
    const db = wx.cloud.database({
      env: 'cloud1-0gmvwabt2ca9f59'
    });
    db.collection('music_favor').field({
      sid: true,
      _id: true,
      singer: true,
      favor: true,
    }).get({
      success: res => {
        var n=res.data.length;
        let song = new Array(n);
        let singer = new Array(n);
        console.log(n);
        for (var i=0;i<n;i++) {
          singer[i] = res.data[i].singer;
          switch(res.data[i].singer) {
            case "李荣浩":
              song[i]=songList.liRongHao[res.data[i].sid];
              break;
            case "许嵩":
              song[i]=songList.xuSong[res.data[i].sid];
              break;
          }
        }
      }
    });
  }
});
```

```
        case "个性电台":
song[i]=broadSongList.geXingDianTai[res.data[i].sid];
        break;
    }
}
this.setData ({
    dataSource: song
})
console.log(this.data.dataSource);
},
fail: err => {
    wx.showToast({
        icon: 'none',
        title: '查询记录失败',
    })
    console.error('[数据库][查询记录]失败: ', err)
},
})
}
})
```

2.4 上传最终的小程序源代码文件。