

# BBS 制作文档

姓名： 郑子豪 学号： 2017210218028

## 一、 项目需求

BBS 所需要的功能有：

- 1、登录、注册功能
- 2、浏览已经发出的所有 根帖子
- 3、通过根帖子浏览所有该根帖子所属的子帖
- 4、查看个人信息
- 5、查看个人曾经参与的所有 根帖子
- 6、删除曾经发布过的帖子

## 二、 功能分析

一、 登录、注册功能：

- a) 登录：用户输入账号与密码，然后进入浏览界面
- b) 注册：未有账号的用户可以注册新的账号和设置对应的密码

二、 浏览所有根帖子：

- a) 用户可以直接看到所有以及发布的根 帖子

三、 通过根帖子浏览所有该根帖子所属的子帖：

- a) 用户可以通过点击根帖子浏览其所属的所有子帖

四、 查看个人信息：

- a) 用户可以在在个人信息界面看到自己的用户名与其他信息

五、 查看个人曾经参与的所有 根帖子：

- a) 用户可以在个人信息界面选择历史记录来查看自己曾经参与讨论的根帖子

六、 删除曾经发布过的帖子：

- a) 用户可以删除自己曾经发布过的根帖子

## 三、 总体设计

BBS 的制作需要登录、注册界面以及登录成功后，内部的浏览界面与个人信息界面，其中登录与注册可以放在同一页中完成，浏览与编写放在同一页中。然后浏览界面可以看得到的讯息在点击之后打开，需要一个新的页面，个人界面需要多一个历史记录的面，所以合计要设计五个界面，按次序分别命名为：Login,search, read,mine,history。

## 四、 实现原理

1、登录、注册功能：

登录：通过 input 中输入的数据，通过 API 接口 login 发送请求，确认数据库中是否有该用户信息，有则登录成功，无则失败。

注册：通过 input 中的数据，通过 API 接口 reg 发送请求，将新的账号密码存入数据库中。

2、浏览所有根帖子：

通过 API 接口 browserroot 直接搜索所有 根帖子

3、通过根帖子浏览所有该根帖子所属的子帖：

通过 API 接口 browsereplybyid 直接搜索所有所属的子帖

#### 4、查看个人信息：

将在登录时保留下的用户名与密码存在全局变量中，在个人信息界面展示

#### 5、查看个人曾经参与的所有 根帖子：

通过 API 接口 browserroot 直接搜索所有 根帖子，然后对比所有 根帖子中的用户 id，与当前用户对比，如果温和则输出到屏幕，否则不输出

#### 6、删除曾经发布过的帖子：

利用 API 接口 deleteproblembyid，可以删除用户发出过的根帖子

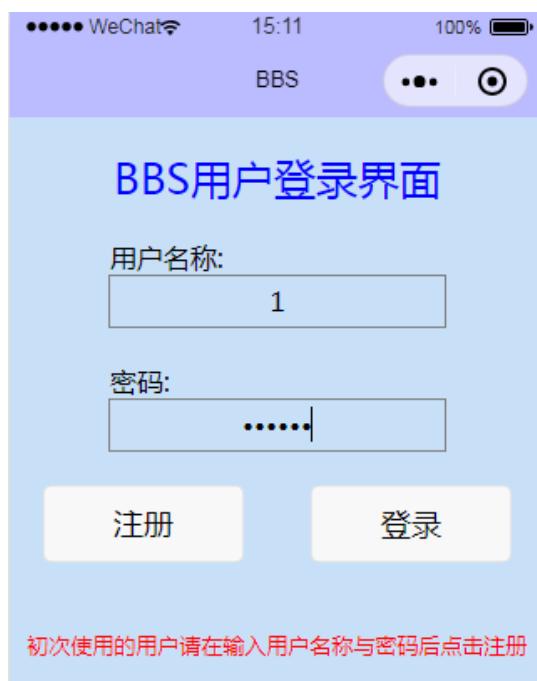
## 五、 模块划分

Login,search, read,mine,history 等五个界面分别代表着登录注册界面，浏览所有根帖子的界面，点击根帖子之后转入查看其所属子帖的界面，个人信息界面以及历史记录界面。本次实验的 BBS 将以上五个模块进行编写。

## 六、 各模块实现过程

### a) 模块一：

- i. 登录注册页面内容为两个输入框，两个文字提示与两个不同的按钮（分别代表登录与注册），在注册成功后会有提示已存在或者注册已成功，登录按钮会返回登录成功（切换页面）或者错误的账号密码。页面如下：



### ii. 各类实现代码

```
// pages/Login/Login.js
var app = getApp();
Page({
  /**
```

```

* 页面的初始数据
*/
data: {
  username: '',
  password: '',
},
bindusername: function (e) {
  console.log(e.detail.value)
  this.setData({
    username: e.detail.value
  })
},
bindpassword: function (e) {
  console.log(e.detail.value)
  this.setData({
    password: e.detail.value
  })
},
clicktemp: function(){
  wx.switchTab({
    url: '/pages/search/search',
  })
},
clickToRegister: function () {
  wx.request({
    url: 'http://172.22.130.33/index.php/Ajaxapi/Ajaxapi/reg/username/' +
this.data.username + '/password/' + this.data.password,
    success: function (res) {
      console.log(res.data.data)
      wx.showToast({
        icon: "none",
        title: '注册成功',
      })
    }
  })
},
clickToLogin: function () {
  var that = this
  wx.request({
    url: 'http://172.22.130.33/index.php/Ajaxapi/Ajaxapi/login/ username/' +
this.data.username + '/password/' + this.data.password,
    success: function (res) {
      console.log(res.data.data)
      app.globalData.username = that.data.username
    }
  })
}

```

```
        console.log("username", app.globalData.username)
        wx.showToast({
            icon: "none",
            title: '登录成功',
        })
        wx.switchTab({
            url: '/pages/search/search'
        })
    }
})
},
/**
 * 生命周期函数--监听页面加载
 */
onLoad: function (options) {

},

/**
 * 生命周期函数--监听页面初次渲染完成
 */
onReady: function () {

},

/**
 * 生命周期函数--监听页面显示
 */
onShow: function () {

},

/**
 * 生命周期函数--监听页面隐藏
 */
onHide: function () {

},

/**
 * 生命周期函数--监听页面卸载
 */
onUnload: function () {
```

```

    },

    /**
     * 页面相关事件处理函数--监听用户下拉动作
     */
    onPullDownRefresh: function () {

    },

    /**
     * 页面上拉触底事件的处理函数
     */
    onReachBottom: function () {

    },

    /**
     * 用户点击右上角分享
     */
    onShareAppMessage: function () {

    }
  })
  <!--pages/Login/Login.wxml-->
<!--pages/result/result.wxml-->
<view class="main">
<text class="title">BBS 用户登录界面\n</text>
  </view>
<view >
<view class="text" >用户名称:</view>
  <view class="main">
    <input style="border:1px solid grey" class="input1" maxlength="10"
bindinput="bindusername" placeholder="请输入用户名称" />
  </view>
<text>\n</text>
  <view class="text">密码:</view>
  <view class="main">
    <input password="true" style="border:1px solid grey" class="input1" maxlength="10"
bindinput="bindpassword" placeholder="请输入密码"/>
  </view >
  <view class="main">
    <text>\n</text>
<button class="button" bindtap="clickToRegister" >注册 </button>
<button class="button" bindtap="clickToLogin" > 登录</button>

```

```

</view>
<text>\n</text>
<view class="main">
<text class="text2">初次使用的用户请在输入用户名称与密码后点击注册</text>
</view>
</view>

/* pages/Login/Login.wxss */
page{
  background:rgb(200, 224, 247)
}
.title{
  font-size:25px;
  color:blue;
  margin: 20px;
}
.main{
  display: flex;
  width: 100%;
  height: 100%;
  align-items: center;
  justify-content: center;
  text-align: center
}
.button{
  margin: 20px;
  width: 140px;
}
.input1{
  height: 30px;
  width: 200px;
}
.text{ margin-left: 60px;
}
.text2{
  font-size: 13px;
  color: red;
  }

```

b) 模块二:

## i. 设计 UI 等

在这个界面里需要一个 input 组件接受外界输入的内容用于编写新的帖子，然后在界面加载时候利用根帖子搜索函数输出所有现在有的、没有父帖子的帖子，可以通过点击帖子的信息转到 read 界面：



利用 browserroot 接口可以得到所有的输出结果，但是很显然在输出的效果上还有很大的提升空间。

## ii. 各类实现代码

```
// pages/search/search.js
var app = getApp();
Page({
  /**
   * 页面的初始数据
   */
})
```

```

data: {
  username: '',
  id: '',
  datas:[],
  data:[],
  word:''
},
bindword:function(e){
  console.log(e.detail.value)
  this.setData({
    word: e.detail.value
  })
},
bindkeyword: function (e) {
  console.log(e.detail.value)
  this.setData({
    pid: e.detail.value
  })
},
clickToread:function(e){ //向数据库申请数据，跳转并将数据保留到下一个页面
  var that=this
  var ids =e.currentTarget.id
  console.log("id:",e.currentTarget.id)
  console.log(that.data.datas[ids])
  that.setData({
    data: that.data.datas[ids]
  })
  wx.setStorageSync('data', that.data.data)
  console.log(that.data.data)
  wx.navigateTo({
    url: '/pages/read/read',
  })
},
clickToCreate:function(){
  wx.request({
    url: 'http://172.22.130.33/index.php/Ajaxapi/Ajaxapi/createproblem/userid/' +
app.globalData.username+'/problem/'+this.data.word,
    success: function (res) {
      wx.showToast({
        icon: "none",
        title: '发布成功',
      })
    }
  })
}

```



```

    },
    /**
     * 生命周期函数--监听页面加载
     */
    onLoad: function (options) {
        console.log(app.globalData.username)
        this.setData({
            username: app.globalData.username
        })
    },

    /**
     * 生命周期函数--监听页面初次渲染完成
     */
    onReady: function () {

    },

    /**
     * 生命周期函数--监听页面显示
     */
    onShow: function () {
        var that = this
        wx.request({
            url: 'http://172.22.130.33/index.php/Ajaxapi/Ajaxapi/browserroot',
            success: function (res) {
                if (res.data.data != false) {
                    console.log("res.data.data:", res.data.data)
                    that.data.datas = res.data.data
                    that.setData({
                        datas: that.data.datas
                    })
                }
                console.log("datas out:", that.data.datas)
            }
        })
    },

    /**
     * 生命周期函数--监听页面隐藏
     */
    onHide: function () {

```

```

    },

    /**
     * 生命周期函数--监听页面卸载
     */
    onUnload: function () {

    },

    /**
     * 页面相关事件处理函数--监听用户下拉动作
     */
    onPullDownRefresh: function () {

    },

    /**
     * 页面上拉触底事件的处理函数
     */
    onReachBottom: function () {

    },

    /**
     * 用户点击右上角分享
     */
    onShareAppMessage: function () {

    }
  })
  <!--pages/search/search.wxml-->
<view>
  <text class="title">所有帖子:</text>
  <view class="main">
    <input style="border:1px solid grey" class="input1" maxlength="100"
bindinput="bindword" placeholder="请输入希望发布的内容" />
    <button class="button" bindtap="clickToCreate" > 发布新帖子 </button>
  </view >
</view >
<!--搜索结果输出，其中需要注意实际上 datas 中的内容可能不是这些，需要联网确认-->
<view>
<view wx:for= "{{datas}}" wx:for-item="item" wx:for-index="idx">
<view id ="{{idx}}" class="image-parent" bindtap="clickToread">

```

```
<text> 发布者 ID: {{item.userid}} ; 正文:{{item.problem}} \n 日期: {{item.createtime}}
\n 父贴 ID:{{item.parentid}}\n</text>
</view>
</view>
</view>

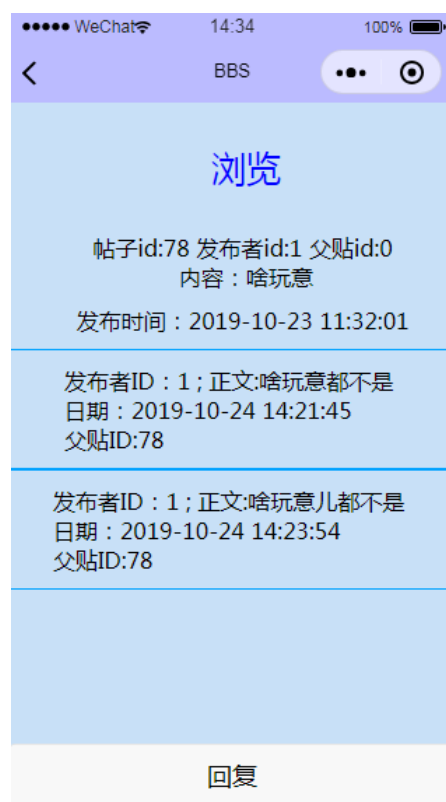
/* pages/search/search.wxss */
.button{
  margin: 10px;
  width: 120px;
  height: 30px;
  font-size: 12px
}
.image-parent{
  height: 200rpx;
  display: flex;
  flex-direction: row;
  align-items: center;
  justify-content: center;
  position: relative;
  border: 2rpx solid rgb(0, 162, 255)
}
page{
  background:rgb(200, 224, 247)
}
.title{
  font-size:25px;
  color:blue;
  margin: 10px;
}
.main{
  display: flex;
  width: 100%;
  height: 100%;
  align-items: center;
  justify-content: center;
  text-align: center;
  margin:10px
}
.input1{
  height: 30px;
  width: 230px;
}
.text{
  margin-left: 30px;
```

```
}  
.text2{  
  font-size: 13px;  
  color: red;  
}
```

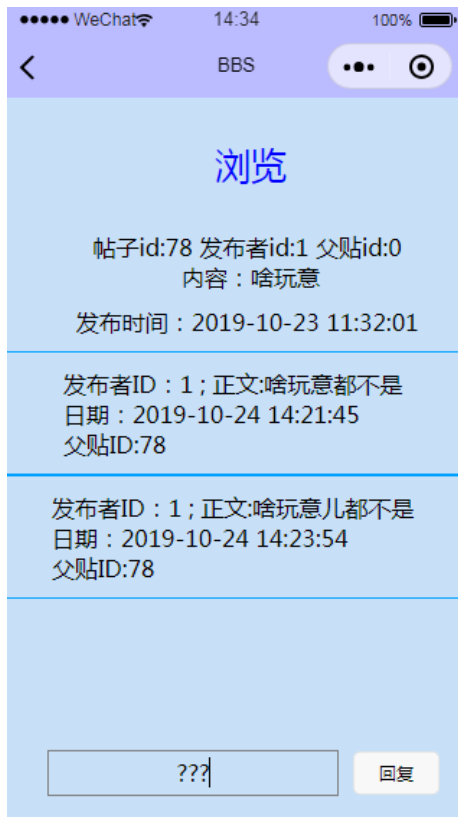
### c) 模块三：

#### i. 设计 UI 等

在查看界面中内容很简单，将 search 页面点击到的帖子信息传递到 read 页，将数值展示出来而已，并且利用根据父帖子搜索的函数 browsereplybyid，在界面下方输出所有的子帖子信息，子帖同样可以点击，重新覆盖这个帖子的内容：



点击回复后会出现输入框：



其中的重点在于 wxml 文件下使用循环读取的相关内容，wx:for，详细可见代码。

## ii. 各类实现代码

```
// pages/read/read.js
var app = getApp()
Page({
  /**
   * 页面的初始数据
   */
  data: {
    pid: '',
    userid: '',
    input_display: false,
    delete_display: false,
    words: '',
    datas: []
  },
  bindword: function (e) {
    console.log(e.detail.value)
    this.setData({
      words: e.detail.value
    })
  },
})
```

```

clickToReply:function(){
    this.setData({
        input_display:true
    })
},
clickToReply2: function () {
    console.log(app.globalData.username, this.data.words, this.data.data.id)
    wx.request({
        url: 'http://172.22.130.33/index.php/Ajaxapi/Ajaxapi/reply/userid/' +
app.globalData.username+'/problem/' + this.data.words+'/parentid/' +
this.data.data.id,
        success: function (res) {
            console.log(res.data.data)
            wx.showToast({
                icon: "none",
                title: '回复成功',
            })
        }
    })
},
clickToread: function (e) { //向数据库申请数据，跳转并将数据保留到下一个页面
    var that = this
    var ids = e.currentTarget.id
    console.log("id:", e.currentTarget.id)
    console.log(that.data.datas[ids])
    that.setData({
        data: that.data.datas[ids]
    })
    wx.setStorageSync('data', that.data.data)
    console.log(that.data.data)
    wx.reLaunch({
        url: '/pages/read/read',
    })
},
clickToDelete:function(){
    wx.request({
        url: 'http://172.22.130.33/index.php/Ajaxapi/Ajaxapi/deleteproblembyid/' +
app.globalData.username+'/pid/'+this.data.data.parentid,
        success: function (res) {
            console.log(res.data.data)
            wx.showToast({
                icon: "none",
                title: '删除成功',
            })
        }
    })
}

```

```

    }
  })
},
/**
 * 生命周期函数--监听页面加载
 */
onLoad: function (options) {
  var that = this
  var datas = wx.getStorageSync('data')
  this.setData({
    data:datas
  })
  console.log("data", this.data.data)
  console.log("userid", this.data.data.userid)
  console.log("username", app.globalData.username)
  if (app.globalData.username == that.data.data.userid) {
    console.log("显示删除按钮")
    this.setData({
      delete_display: true
    })
  }
  wx.request({
    url: 'http://172.22.130.33/index.php/Ajaxapi/Ajaxapi/browsereplybyid/pid/'+
that.data.data.id,
    success: function (res) {
      if (res.data.data != false) {
        console.log("res.data.data:", res.data.data)
        that.data.datas = res.data.data
        that.setData({
          datas: that.data.datas
        })
      }
      console.log("datas out:", that.data.datas)
    }
  })
},
/**
 * 生命周期函数--监听页面初次渲染完成
 */
onReady: function () {
},

```

```
/**
 * 生命周期函数--监听页面显示
 */
onShow: function () {

},

/**
 * 生命周期函数--监听页面隐藏
 */
onHide: function () {

},

/**
 * 生命周期函数--监听页面卸载
 */
onUnload: function () {

},

/**
 * 页面相关事件处理函数--监听用户下拉动作
 */
onPullDownRefresh: function () {

},

/**
 * 页面上拉触底事件的处理函数
 */
onReachBottom: function () {

},

/**
 * 用户点击右上角分享
 */
onShareAppMessage: function () {

}
})
<!--pages/read/read.wxml-->
```



```

<view class="main">
<text class="title">浏览 \n</text>
<view wx:if="{{delete_display}}">
<button class="button" bindtap="clickTodelete" > 删除本帖 </button>
</view></view>
<!--输出帖子内容-->
<view class="main">
<text>帖子 id:{{data.id}} 发布者 id:{{data.userid}} 父贴 id:{{data.parentid}} \n 内容:
{{data.problem}} </text>
</view>
<view class="main">
<text> 发布时间: {{data.createtime}} \n </text>
</view>
<!--输出子贴内容-->
<view>
<view wx:for= "{{datas}}" wx:for-item="item" wx:for-index="idx">
<view id="{{idx}}" class="image-parent" bindtap="clickToread">
<text> 发布者 ID: {{item.userid}}; 正文:{{item.problem}} \n 日期: {{item.createtime}}
\n 父贴 ID:{{item.parentid}}\n</text>
</view>
</view>
</view>
<!--回复-->
<view wx:if="{{!input_display}}">
<button class="button1" bindtap="clickToReply" > 回复 </button>
</view>
<view class="button1" wx:if="{{input_display}}">
<view class="main">
<input style="border:1px solid grey" class="input1" maxlength="100"
bindinput="bindword" placeholder="请输入希望回复的内容" />
<button wx:if="{{input_display}}" class="button" bindtap="clickToReply2" > 回复
</button>
</view>
</view>
/* pages/read/read.wxss */
.button2{
margin: 10px;
width: 60px;
height: 30px;
font-size: 12px;
position: fixed;
bottom: 0;
}
.button1{

```

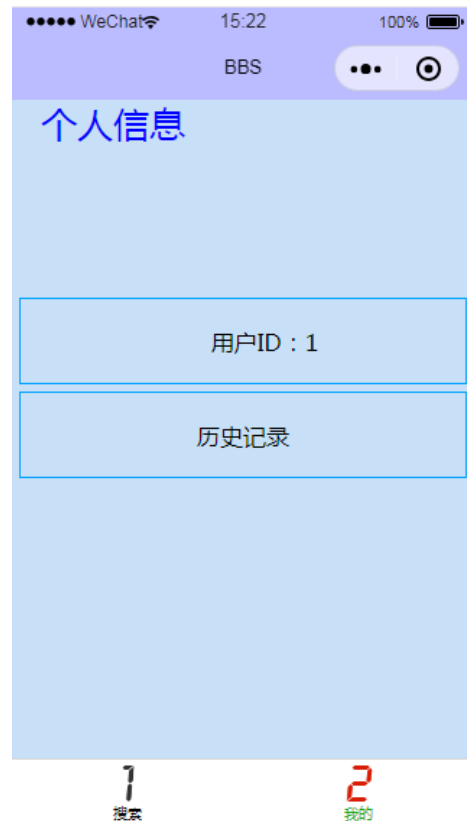
```
    position: fixed;
    bottom: 0;
    width: 100%;
}
.button{
    margin: 10px;
    width: 60px;
    height: 30px;
    font-size: 12px;
}
.image-parent{
    height: 200rpx;
    display: flex;
    flex-direction: row;
    align-items: center;
    justify-content: center;
    position: relative;
    border: 2rpx solid rgb(0, 162, 255)
}
page{
    background:rgb(200, 224, 247)
}
.title{
    font-size:25px;
    color:blue;
    margin: 20px;
}
.main{
    display: flex;
    width: 100%;
    height: 100%;
    align-items: center;
    justify-content: center;
    text-align: center;
    margin:10px
}
.input1{
    height: 30px;
    width: 200px;
}
.text{
    margin-left: 30px;
}
.text2{
```

```
font-size: 13px;
color: red;
}
```

#### d) 模块四：

##### i. 设计 UI 等

Mine 界面并没有什么特殊的，被我简化到仅有用户 ID 展示与历史记录按钮，所以很简单。整体界面如下：



##### ii. 各类实现代码

```
// pages/mine/mine.js
var app = getApp()
Page({

  /**
   * 页面的初始数据
   */
  data: {
    username: app.globalData.username,
  },
  bindhis:function(){
    console.log(this.data.username)
    wx.navigateTo({
```

```
        url: '/pages/history/history',
    })
},
/**
 * 生命周期函数--监听页面加载
 */
onLoad: function (options) {
    this.setData({
        username: app.globalData.username
    })
},

/**
 * 生命周期函数--监听页面初次渲染完成
 */
onReady: function () {

},

/**
 * 生命周期函数--监听页面显示
 */
onShow: function () {

},

/**
 * 生命周期函数--监听页面隐藏
 */
onHide: function () {

},

/**
 * 生命周期函数--监听页面卸载
 */
onUnload: function () {

},

/**
 * 页面相关事件处理函数--监听用户下拉动作
 */
onPullDownRefresh: function () {
```

```

    },

    /**
     * 页面上拉触底事件的处理函数
     */
    onReachBottom: function () {

    },

    /**
     * 用户点击右上角分享
     */
    onShareAppMessage: function () {

    }
  })
  <!--pages/mine/mine.wxml-->
<text class="title">个人信息</text>
<view >
<text class="title">\n\n\n</text>
</view>

<!--从全局变量得到个人 id-->
<view class="image-parent">
<text class="text">用户 ID: {{username}}</text>
</view>

<view class="image-parent" bindtap="bindhis">
<block class="text">历史记录</block>
</view>

```

## e) 模块五：

### i. 设计 UI 等

该界面用于展示用户个人发布过的帖子记录，在搜索所有帖子之后输出 userid 与当前用户 id 相同的结果即可，这个页面的所有结果也可以点击，点击后也会转移至阅读界面，可以进行删除，回复操作，其中删除操作只有用户 id 与帖子发布者 id 相同时才会可见，由于接口只有根帖子遍历，所以编写代码比较繁琐，本次制作中仅制作了对根帖子的删除操作，由于界面本身没有特殊性，不做展示。

### ii. 各类实现代码

```
// pages/history/history.js
```

```

var app = getApp()
Page({

  /**
   * 页面的初始数据
   */
  data: {
    datas:[],
  },
  clickToread:function(e){
    var that = this
    var ids = e.currentTarget.id
    console.log("id:", e.currentTarget.id)
    console.log(that.data.datas[ids])
    that.setData({
      data: that.data.datas[ids]
    })
    wx.setStorageSync('data', that.data.data)
    console.log(that.data.data)
    wx.navigateTo({
      url: '/pages/read/read',
    })
  },
  /**
   * 生命周期函数--监听页面加载
   */
  onLoad: function (options) {
    var that =this
    wx.request({
      url: 'http://172.22.130.33/index.php/Ajaxapi/Ajaxapi/browserroot',
      success: function (res) {
        if (res.data.data != false) {
          console.log("res.data.data:", res.data.data)
          console.log("datas:", that.data.datas)
          console.log("userid:", app.globalData.username)
          for (var a in res.data.data){
            console.log("a:", res.data.data[a])
            if (res.data.data[a].userid == app.globalData.username){
              //将a加入数组datas
              that.data.datas.push(res.data.data[a])
            }
          }
        }
        that.setData({
          datas: that.data.datas
        })
      }
    })
  }
})

```

```
    })
    console.log("datas:", that.data.datas)
  }
}
})
```

//如果想要删除子贴，那么还需要循环跑一遍从各个帖子中和 **userid** 对上的所有讯息，效率太低了，我更推荐改掉接口，所以这里偷个懒

```
},
```

```
/**
 * 生命周期函数--监听页面初次渲染完成
 */
onReady: function () {
```

```
},
```

```
/**
 * 生命周期函数--监听页面显示
 */
onShow: function () {
```

```
},
```

```
/**
 * 生命周期函数--监听页面隐藏
 */
onHide: function () {
```

```
},
```

```
/**
 * 生命周期函数--监听页面卸载
 */
onUnload: function () {
```

```
},
```

```
/**
 * 页面相关事件处理函数--监听用户下拉动作
 */
onPullDownRefresh: function () {
```

```
},
```

```

/**
 * 页面上拉触底事件的处理函数
 */
onReachBottom: function () {

},

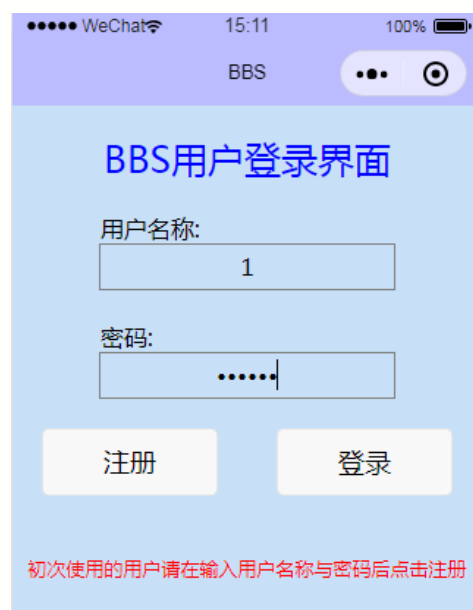
/**
 * 用户点击右上角分享
 */
onShareAppMessage: function () {

}
})
<!--pages/history/history.wxml-->
<view>
<text class="title">历史记录\n </text>
<view wx:for= "{{datas}}" wx:for-item="item" wx:for-index="idx">
<view id="{{idx}}" class="image-parent" bindtap="clickToread">
<text >父贴 ID:{{item.parentid}} \n 正文:{{item.problem}} \n 日期: {{item.createtime}}
\n\n </text>
</view>
</view>
</view>
</view>

```

## 七、 测试

首先是登录

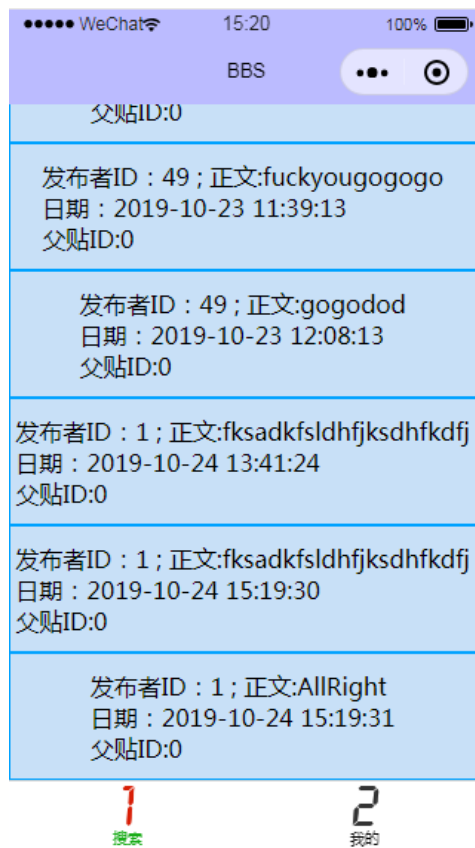


然后进入浏览界面，我们尝试发布新帖子 AllRight:





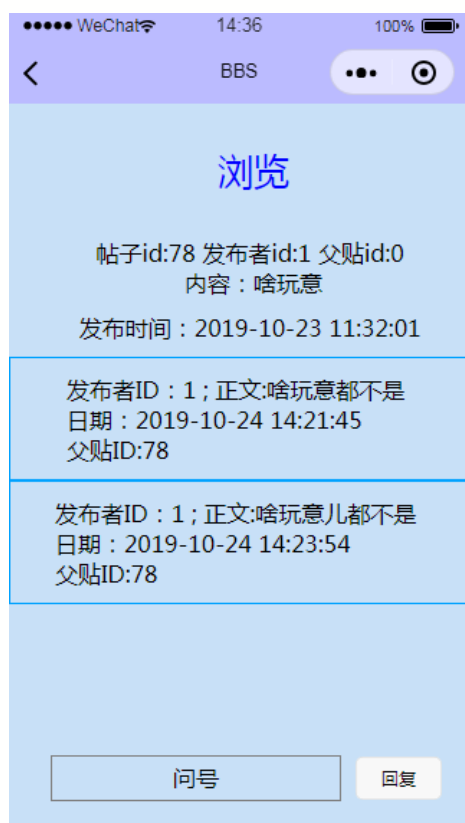
点击发布，查看是否有效



可以看到 AllRight 发布成功，然后是点击其中的某一项进入 read 界面：



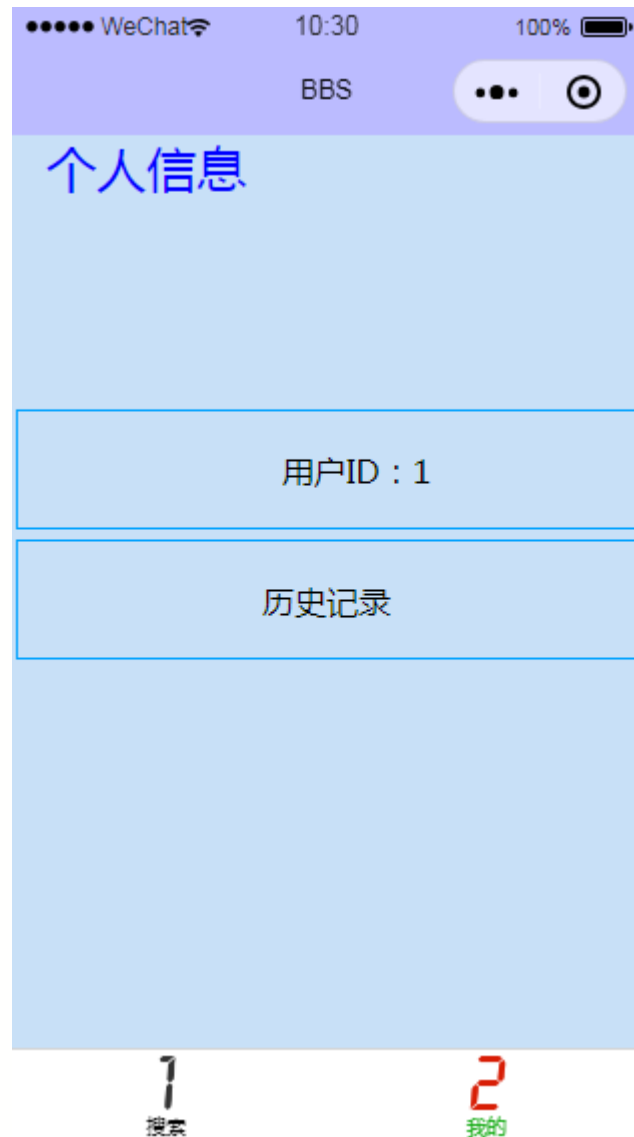
在这个界面输入回复贴，内容为“问号”



然后重新读取之后可以发现结果有效：



个人信息界面，由于本次内容没有过多信息，所以尽量简化



可以点击历史记录进入浏览界面, 之后的效果与之前的搜索界面到 read 界面是类似的, 所以并无展示。

测试基本成功。