

# 乡村志愿服务小程序-实验报告

杨建国-1913149

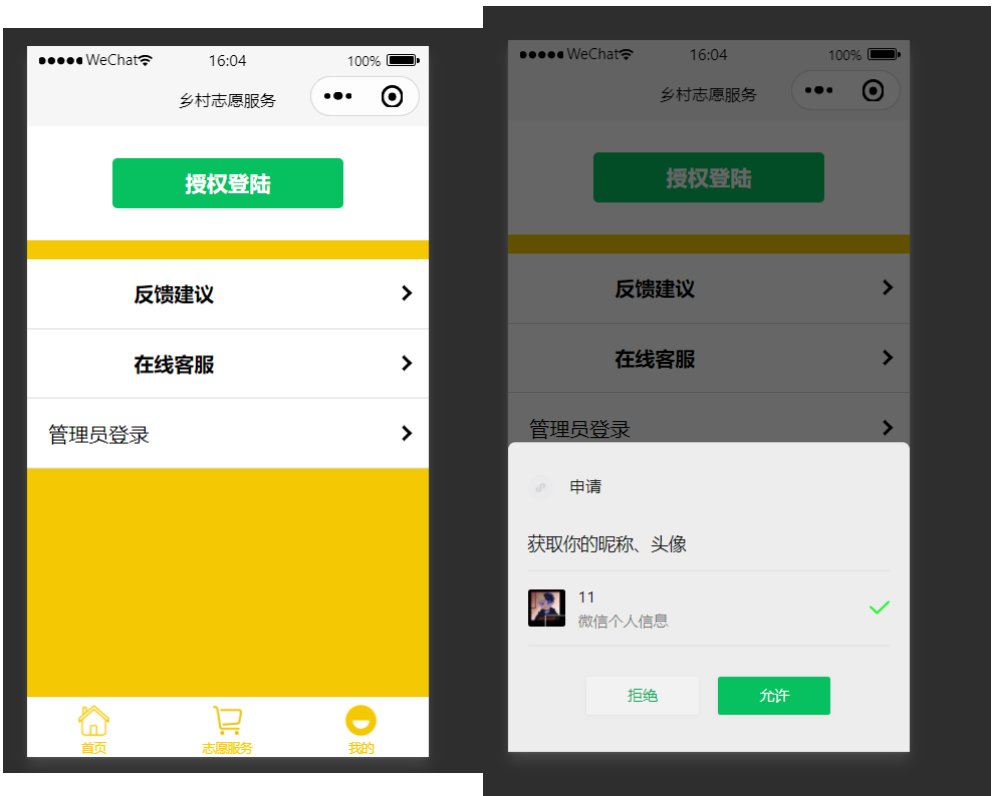
## 一.项目背景

基于目前乡村经济发展热潮以及人口老龄化严重等问题，乡村的一些活动对于志愿者的需求也越来越大，于是，为了满足大家的需求，我决定实现一个农村志愿服务平台，通过互联网构建便捷，规范化的农村志愿服务体系，提高志愿者到乡村服务的积极性，发展农业生产、促进邻里互助、倡导文明风尚、关怀儿童教育、协调矛盾冲突、保护生态环境等。解决由于乡村渠道受限、信息不畅，志愿者招募方式单一、范围小、质量低；受困管理机制的缺陷，提高志愿者热情。

## 二.项目功能分析

项目应包含以下功能：

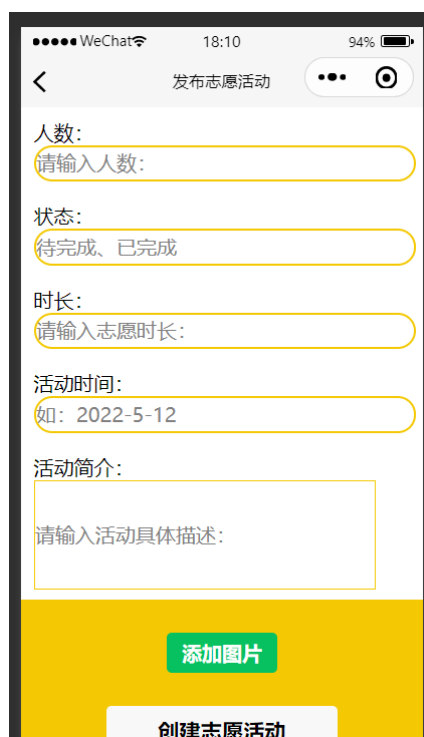
1.用户登录功能：用户使用微信经过授权后可登录



2.发布者登录功能：活动发布者可通过管理员账号进行登录



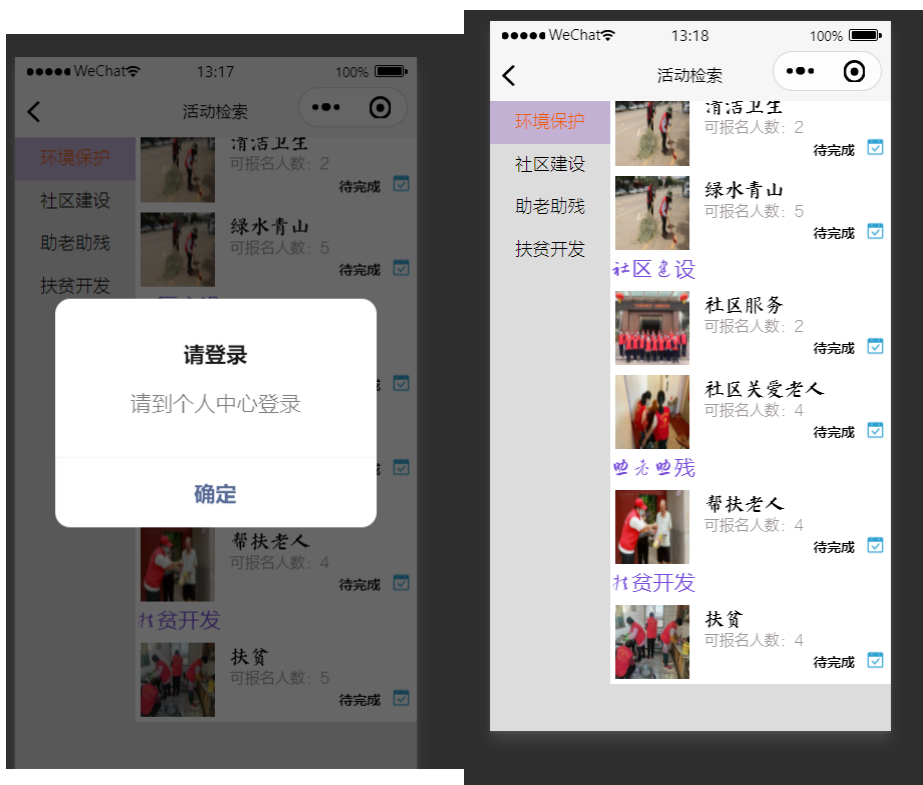
**3.发布志愿信息功能：发布者登录后可发布志愿活动（填写志愿活动具体信息，包括上传图片等功能）**



**4.志愿活动浏览：用户可查看所有未完成的志愿活动（通过活动类别对其进行分类，用户也可通过推荐活动进入志愿活动页面）**



5.用户报名活动：点击报名后会先检测用户是否登录，如登录后才可继续报名，并将该活动所需人数-1



6.签到打卡：用户报名后可在签到打卡页面看到已报名的活动并进行活动签到



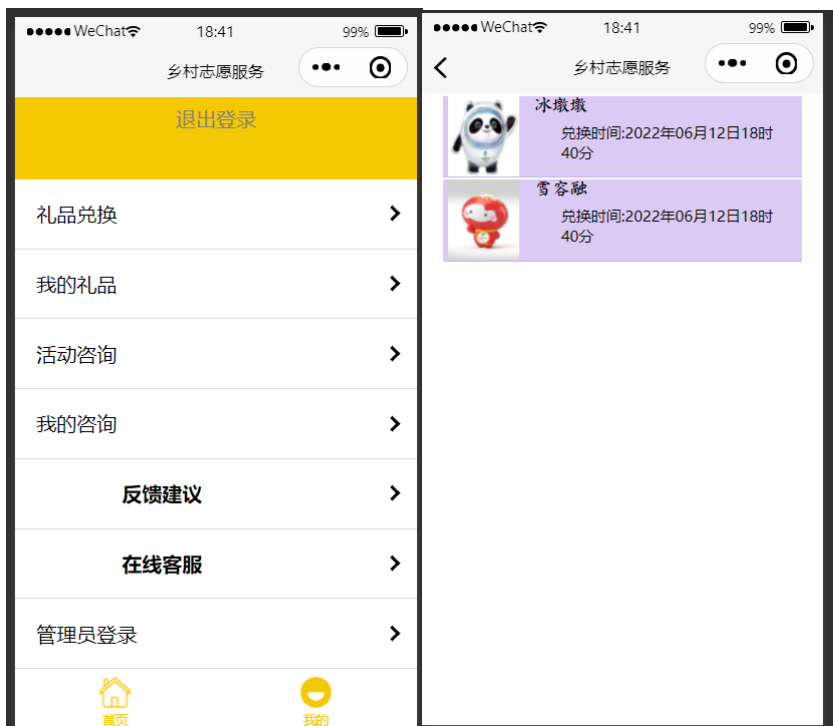
## 7.志愿活动管理：发布者登录后可对志愿活动进行管理（完成活动后用户会增加志愿时长）



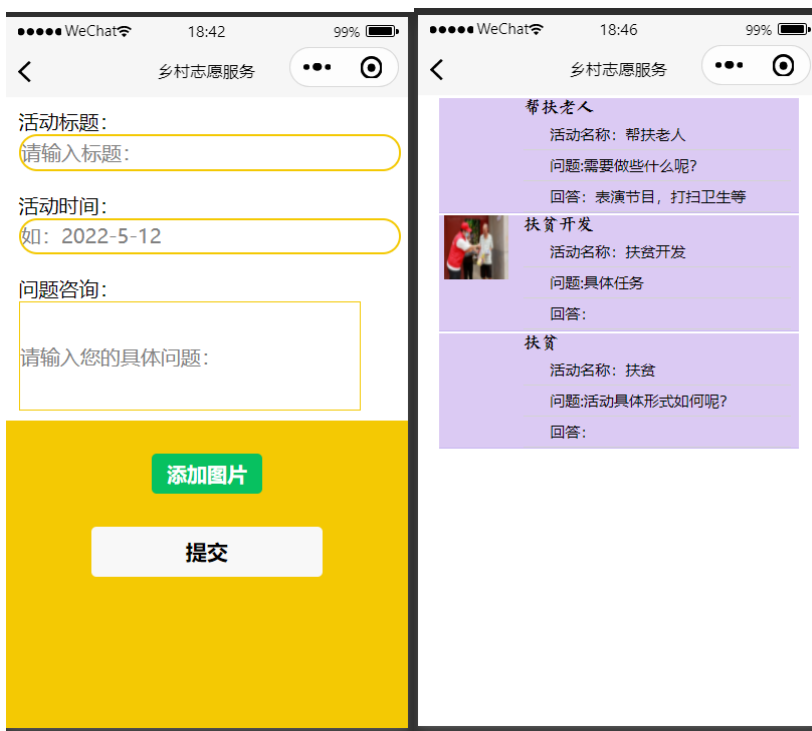


8.礼品兑换：用户可使用自己的志愿时长进行礼品兑换





## 9.问题咨询：用户可就志愿活动的一些问题对发布者进行提问



## 10.发布者回答：发布者登录后可对问题进行回答



## 三.技术及方法

**1.登录功能：**发布者登陆功能，操作者输入用户名及密码，我们检索数据库如有匹配项则登录成功并跳转到发布者页面

```
//登录
login(name, password) {
  db.collection('admin').where({
    name: name,
    password: password
  }).get()
  .then(res => {
    console.log("登陆成功", res)
    if (res.data && res.data.length > 0) {
      this.setData({
        isAdmin: true
      })
      let admin = {
        name: name,
        password: password
      }
      wx.setStorageSync('admin', admin)
    } else {
```

## 2.创建志愿服务

**1.添加及删除图片功能：**使用流量微信开发者文档提供的API实现

```

up: function (e) {
  let that = this
  wx.chooseImage({
    count: 1,
    sizeType: ['original', 'compressed'],
    sourceType: ['album', 'camera'],
    success (res) {
      var timestamp = Date.parse(new Date());
      timestamp = timestamp / 1000;
      // tempFilePath可以作为 img 标签的 src 属性显示图片
      const tempFilePaths = res.tempFilePaths
      wx.cloud.uploadFile({
        cloudPath: 'product/'+timestamp+'.png',
        filePath: tempFilePaths[0], // 文件路径
        success: res => {
          // 返回文件 ID
          console.log(res.fileID)
          that.setData({
            img: that.data.img.concat(res.fileID)
          })
        },
        fail: console.error
      })
    }
  })
},

```

```

delete: function (e) {
  let that = this
  console.log(that.data.img)
  console.log(e.currentTarget.dataset.id)
  var id = e.currentTarget.dataset.id;
  var img = that.data.img;
  img.splice(id, 1)
  that.setData({
    img: img
  })
  wx.cloud.deleteFile({
    fileList: [e.currentTarget.dataset.src],
    success: res => {
      // handle success
      console.log(res.fileList)
    },
    fail: err => {
      // handle error
    },
  })
  console.log(that.data.img)
},

```

**2.创建志愿服务信息：**通过input实现，提交功能通过按钮实现并绑定submit函数，在js文件中实现了submit函数，通过对数据库的增加实现志愿信息发布



```
<view wx:if="{{!isAdd}}" class="container">
  <form bindsubmit="submit">
    <view class="line_lpx" />
    <view class="item_root">
      活动类型:
      <input class="textareaa25px" name="type" placeholder="0: 助教, 1: 助教" />
    </view>
    <view class="line_lpx" />
    <view class="item_root">
      活动主题:
      <input class="textareaa25px" name="name" placeholder="请输入标题: " />
    </view>
    <view class="item_root">
      活动地点:
      <input class="textareaa25px" name="address" placeholder="活动地点: " />
    </view>
    <view class="line_lpx" />
    <view class="item_root">
      人数:
      <input class="textareaa25px" name="number" placeholder="请输入人数: " />
    </view>
    <view class="line_lpx" />
    <view class="item_root">
      状态:
      <input class="textareaa25px" name="state" placeholder="0: 未开始, 1: 进行中, 2: 已完成: " />
    </view>
  </form>
  <button type="submit">提交</button>
</view>

submit: function(e) {
  let that = this
  console.log(e)
  db.collection('v-service').add({
    data: {
      name: e.detail.value.name,
      number: e.detail.value.number,
      duration: e.detail.value.duration,
      state: e.detail.value.state,
      date: e.detail.value.date,
      information: e.detail.value.information,
      type: e.detail.value.type,
      picture: that.data.img,
    },
  }, success: function(res) {
    console.log('增加数据成功', res)
    that.setData({
      isAdd: true
    })
    wx.showToast({
      icon: 'none',
      title: '活动创建成功',
    })
  }, fail: function(res) {
    console.log('增加数据失败', res)
    that.setData({
      isAdd: false
    })
  })
}
```

### 3. 志愿服务浏览：获取所有未完成活动，对数据进行分类

```
let endData = []
dataList.forEach(item => {
  if (tempArr.indexOf(item.type) === -1) {
    endData.push({
      title: item.type,
      list: [item]
    })
    tempArr.push(item.type)
  } else {
    for (let j = 0; j < endData.length; j++) {
      if (endData[j].title === item.type) {
        endData[j].list.push(item)
        break
      }
    }
  }
})
// 过滤数组, 添加id
endData.map((item, index) => {
  item.id = index
})
console.log('过滤后', endData)
this.setData({
  cartlist: cartlist,
  menuArr: endData,
})
this.getTotalPrice()
this.getHeightArr()
}).catch(res => {
  console.log('活动数据请求失败', res)
})

<!-- 菜单区域 -->
<view class="main">
  <scroll-view scroll-y="true" scroll-with-animation="true" bindscroll="rightScrollFn" class="left">
    <view wx:for="{{menuArr}}" data-mid="{{item.id}}" wx:key="index"
      class="scroll-view-item left-tab {{leftActiveNum===item.id ? 'active' : ''}}" bindtap="leftClickFn">{{item.title}}
    </view>
  </scroll-view>
  <scroll-view scroll-y="true" scroll-with-animation="true" bindscroll="rightScrollFn" class="right">
    <scroll-into-view="top"{{tab}}>
      <view wx:for="{{menuArr}}" wx:key="index" id="top{{item.id}}" class="scroll-view-item rightblock">
        <view class="title">{{item.title}}</view>
        <block wx:for="{{item.list}}" wx:key="index">
          <view class="caipin_item">
            <image class="cai_img" src="{{item.picture}}"/>
            <view class="cai_root">
              <view class="cai_title" data-index="{{index}}">{{item.name}}</view>
              <view class="cai_sell">可报名人数: {{item.number}}</view>
              <view class="stepper">
                <view class="value" data-id="{{item_id}}">
                  {{item.state}}</view>
                <view bindtap="signUp" data-item="{{item}}">
                  <image class="jiajian" src="/images/home3.png"/>
                </view>
              </view>
            </view>
          </view>
        </block>
      </view>
    </scroll-into-view>
  </scroll-view>
</view>
```

### 4. 报名志愿活动：用户点击报名按钮后遍历数组找到目标活动并判断是否满员，然后向报名活动表中添加数据，修改志愿活动所需人数

```

signUp(e){
  let item = e.currentTarget.dataset.item;
  let f = false;
  let menuArr = this.data.menuArr
  menuArr.forEach(v=>{
    v.list.forEach(v2 =>{
      if (v2._id == item._id){
        console.log("找到活动了")
        if(v2.number<=0){
          wx.showModal({
            title: '提示',
            content: '活动已满员'
          })
        }else{
          let userInfo = app.globalData.userInfo;
          if (!userInfo || !userInfo.nickName) {
            wx.showModal({
              title: '请登录',
              content: '请到个人中心登录',
              showCancel: false, //去掉取消按钮
              success: function (res) {
                if (res.confirm) {
                  wx.switchTab({
                    url: '../pages/me/me',
                  })
                }
              }
            })
          }
        }
      }
    })
  })

  db.collection("voluntary_activity").where({
    name:v2.name,
  }).update({
    data:{
      number:v2.number
    },success:function(res){
      console.log("人数更新成功",res)
    },fail:function(res){
      console.log("人数更新失败",res)
    }
  })

  db.collection("register_list").add({
    data:{
      avatarUrl:userInfo.avatarUrl,
      activity_name:v2.name,
      name:userInfo.nickName,
      status:0,
      duration:v2.duration,
      picture:v2.picture,
      state:v2.state,
      id:v2._id,
      create_time:new Date().getTime()
    },success:function(res){
      wx.showToast({
        title: '提示',
        content: '报名成功',
      })
    }
  })
}

```

## 5.签到打卡：遍历确认活动，更新报名状态

```

goodList.forEach(v => {
  if(v._id==item._id){
    console.log("确认活动")
    db.collection("register_list").where({
      name:userInfo.nickName,
      avatarUrl:userInfo.avatarUrl,
      activity_name:v.activity_name,
      picture:v.picture
    }).update({
      data:{
        status:1
      },
      success:function(res){
        console.log("更新成功")
        wx.redirectTo({
          url: '../pages/register/register',
        })
      },fail:function(res){
        console.log("更新失败")
      }
    })
  }
})
}

```

## 6.志愿活动管理

### 1.顶部tab切换，获取活动数据

```

//顶部tab切换
navbarTap: function (e) {
  let index = e.currentTarget.dataset.idx;
  this.setData({
    currentTab: index
  })
  if (index == 0) {
    orderStatus = 0;
  } else if (index == 1) {
    orderStatus = 1;
  } else if (index == 2) {
    orderStatus = 2;
  } else {
    orderStatus = 0;
  }
  this.getMyOrderList();
},

let openid = app.__checkOpenid();
if (!openid) {
  return;
}
//请求自己后台获取用户openid
wx.cloud.callFunction({
  name: 'getSignupList',
  data: {
    action: 'admin',
    orderStatus: orderStatus
  }
})
.then(res => {
  console.log("用户报名列表", res)
  this.setData({
    list: res.result.data
  })
})

```

## 2.完成活动：修改活动报名状态，修改活动状态并增加志愿时长

```

let openid = app.globalData.openid;
const _ = db.command
console.log(e.currentTarget.dataset.id)
wx.cloud.callFunction({
  name: 'adminhandle',
  data: {
    id: e.currentTarget.dataset.id
  }
}).then(res => {
  console.log('活动完成ok', res)
  if (res.result && res.result.stats && res.result.stats.updated > 0) {
    wx.showToast({
      title: '修改成功',
    })
    this.getMyOrderList()
  } else {
    wx.showToast({
      icon: 'none',
      title: '提交失败',
    })
  }
})
}

//活动中更新对应活动状态
db.collection('register_list').doc(e.currentTarget.dataset.id)
.get().then(res => {
  db.collection('voluntary_activity').where({name: res.data.activity_name})
  .update({
    data: {
      state: '已完成', //活动完成
    }
  })
})

//用户表中增加相应志愿时长
db.collection('register_list').doc(e.currentTarget.dataset.id)
.get().then(res => {
  let increaceNum = Number(res.data.duration)
  db.collection('user').where({openid: openid}).update({
    data: {
      length: _.inc(Number(increaceNum)),
    }
  })
})

```

## 7.礼品兑换：将礼品添加到礼品数据库中，并减少自己的志愿总时长（该时长来源于user表，登录时自动创建一条数据，存储用户相关信息以及该用户志愿时长）

```

let item = e.currentTarget.dataset.item;
const _ = db.command
console.log("数据是: ",item)
let giftList = this.data.giftList
giftList.forEach(v2 => {
  if(v2._id==item._id){
    console.log("确认礼品")
    if(v2.num<=0){
      wx.showModal({
        title: '提示',
        content: '礼品库存不足'
      })
    }else{
      console.log("兑换礼品")
      let userInfo = app.globalData.userInfo;
      db.collection("mygift").add({
        data:{
          avatarUrl:userInfo.avatarUrl,
          gift_name:v2.name,
          name:userInfo.nickName,
          picture:v2.picture,
          create_time:new Date().getTime()
        },success:function(res){
          wx.showToast({
            title: '提示',
            content: '兑换成功',
          })
        }
      })
    }
  }
})

```

```

<!-- 活动展示 -->
<view wx:if="{{giftList&&giftList.length>0}}">
  <block wx:for="{{giftList}}" wx:key="index">
    <view class="good_item">
      <image class="good_img" src="{{item.picture}}"/>
      <view class="good_right">
        <view class="good_title" data-index="{{index}}">{{item.name}}</view>
        <view class="dibuttxt">所需积分:{{item.score}}</view>
        <view class="dibuttxt">我的积分:{{length}}</view>
        <view class="dibuttxt">是否有存货: {{item.empty}}</view>
        <view data-item="{{item}}" bindtap="Finish">
          <image class="category_item_image" src="/images/home3.png"/>
          <text>兑换</text>
        </view>
      </view>
    </view>
  </block>
</view>

```

## 8.我的礼品：从云端获取该用户所拥有的礼品数据并展示

```

let that = this
let userInfo = app.globalData.userInfo;
if (!userInfo || !userInfo.nickName) {
  wx.showModal({
    title: '请登录',
    content: '请到个人中心登录',
    showCancel: false, //去掉取消按钮
    success: function (res) {
      if (res.confirm) {
        wx.switchTab({
          url: '../pages/me/me',
        })
      }
    }
  })
}
return;
db.collection("mygift").where({
  name:userInfo.nickName,
  avatarUrl:userInfo.avatarUrl
}).get({
  success:function(res){
    console.log("数据查询成功",res)
    that.setData({
      giftList:res.data
    })
  },fail:function(res){
    console.log("数据获取失败",res)
  }
})

```

```

<wx:src="../../utils/filter.wx" module="filters"/>
<!-- 活动展示 -->
<view wx:if="{{giftList&&giftList.length>0}}">
  <block wx:for="{{giftList}}" wx:key="index">
    <view class="good_item">
      <image class="good_img" src="{{item.picture}}"/>
      <view class="good_right">
        <view class="good_title" data-index="{{index}}">{{item.gift_name}}</view>
        <view class="dibuttxt">兑换时间:{{item.create_time?filters.formatDate(item.create_time):''}}</view>
        <!-- <view class="good_price" data-index="{{index}}">{{item.price}}</view> -->
      </view>
    </view>
  </block>
</view>

```

## 9.发起问题：用户可创建问题，将该问题相关信息添加到数据库中（可添加图片），在我的问题页面可通过获取数据库内容展示已提问问题

```

submit:function(e){
  let that = this
  console.log(e)
  if(e.detail.value.name!=""&&e.detail.value.activity_name!=""&&e.detail.value.date){
    db.collection('questions').add({
      data:{
        name:app.globalData.userInfo.nickName,
        activity_name:e.detail.value.activity_name,
        date:e.detail.value.date,
        question:e.detail.value.description,
        status:0,
        question_image:that.data.img,
        _createTime: new Date().getTime()
      },success:function(res){
        wx.showToast({
          title: '提交成功',
        })
        wx.redirectTo({
          url: '../pages/me/me',
        })
      }
    })
  }
}

let that = this
let userInfo = app.globalData.userInfo;
if (!userInfo || !userInfo.nickName) {
  wx.showModal({
    title: '请登录',
    content: '请到个人中心登录',
    showCancel: false, //去掉取消按钮
    success: function (res) {
      if (res.confirm) {
        wx.switchTab({
          url: '../pages/me/me',
        })
      }
    }
  })
  return;
}

db.collection("questions").where({
  name:userInfo.nickName
}).get({
  success:function(res){
    console.log("数据查询成功",res)
    that.setData({
      questionList:res.data
    })
  }
})

```

## 10.管理问题页面：发布者点击回答问题跳转到回答问题页面

```

//请求自己后台获取用户openid
wx.cloud.callFunction({
  name: 'getQuestionList',
  data: {
    action:'admin',
    orderStatus: orderStatus
  }
})
.then(res => {
  console.log("问题列表成功", res)
  this.setData({
    questionList: res.result.data
  })
}).catch(res => {
  console.log("问题列表失败", res)
})
},
completeAnswer(e)
{
  this.setData({
    ID: e.currentTarget.dataset.id
  })
  var that=this
  wx.redirectTo({
    url: '../pages/writeAnswer/writeAnswer'
  })
}

```

## 11.回答问题：发布者回答问题后将该内容更新到questions表的answer中，以便用户可以看到该问题的回答并修改该问题的状态

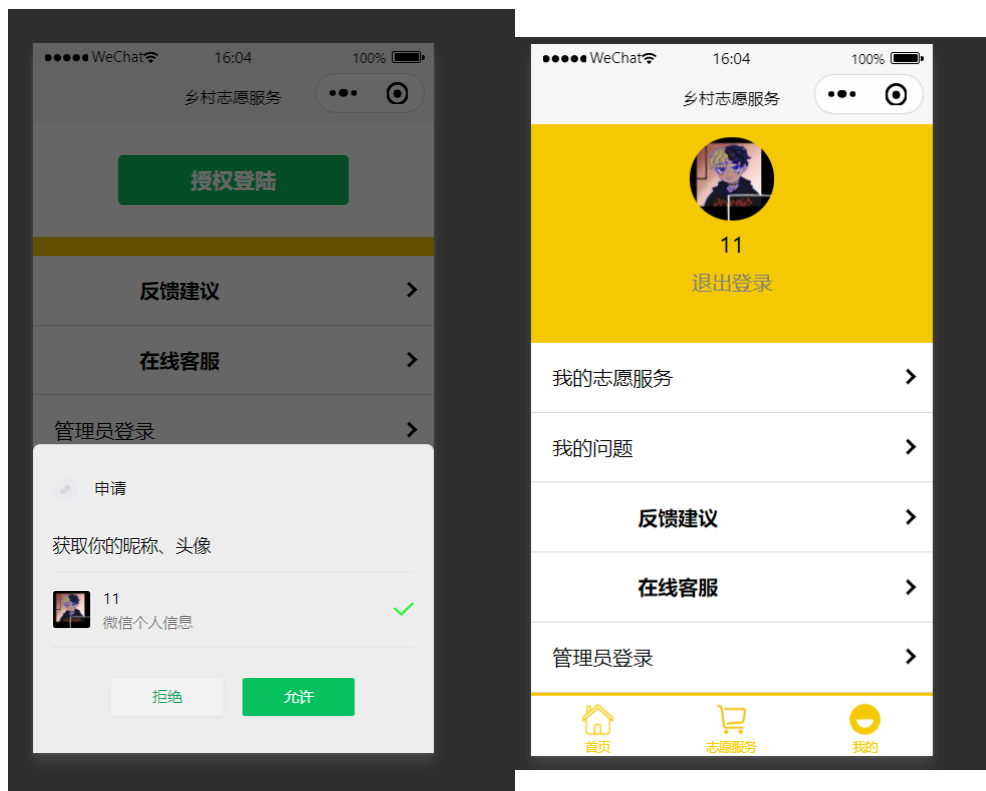
```

submit:function(e){
  let that = this
  console.log(e)
  if(e.detail.value.name!=""&&e.detail.value.activity_name!=""&&
    db.collection('questions').doc(that.data.ID).update({
      data:{
        answer:e.detail.value.description,
        status:1,
        _updateTime: new Date().getTime()
      },success:function(res){
        wx.showToast({
          title: '提交成功',
        })
        wx.redirectTo({
          url: '../questions_control/questions_control',
        })
      }
    })
  })
}else{
  wx.showToast({
    title: '你还有未填信息',
    icon:"none"
  })
}
}
}

```

## 四.产品实现说明

### 1.登录功能：



### 2.发布者登录： 点击管理员登录并输入用户名： nku， 密码： nku666登录：



### 3.发布志愿信息：输入相关信息点击创建志愿活动即可创建

### 4.浏览志愿活动：点击活动浏览按钮或者下方的推荐活动可跳转到所有未完成的活动页面

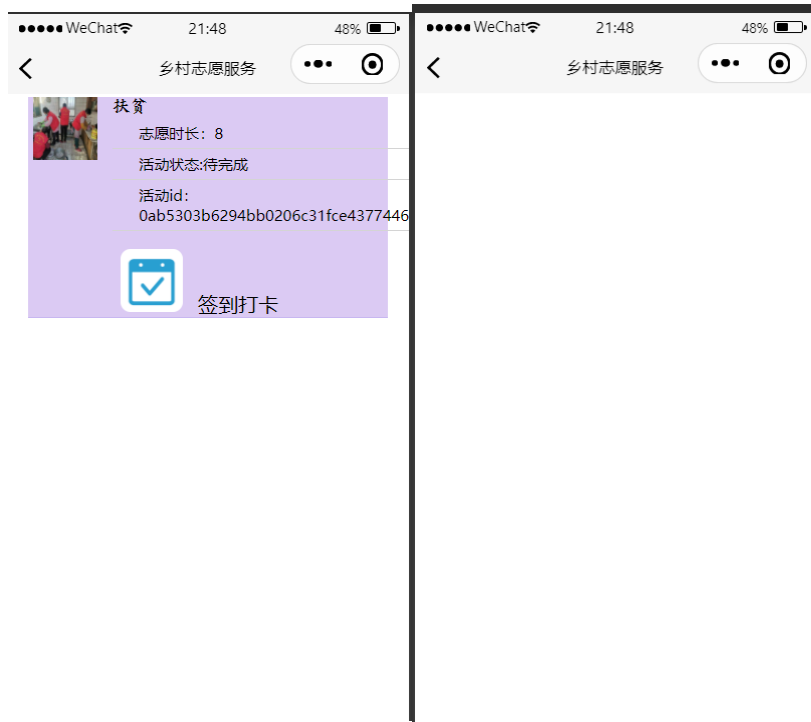


5.报名志愿活动：如未登录，需要登录，点击确定按钮跳转到个人页面，然后返回活动浏览页面可报名活动

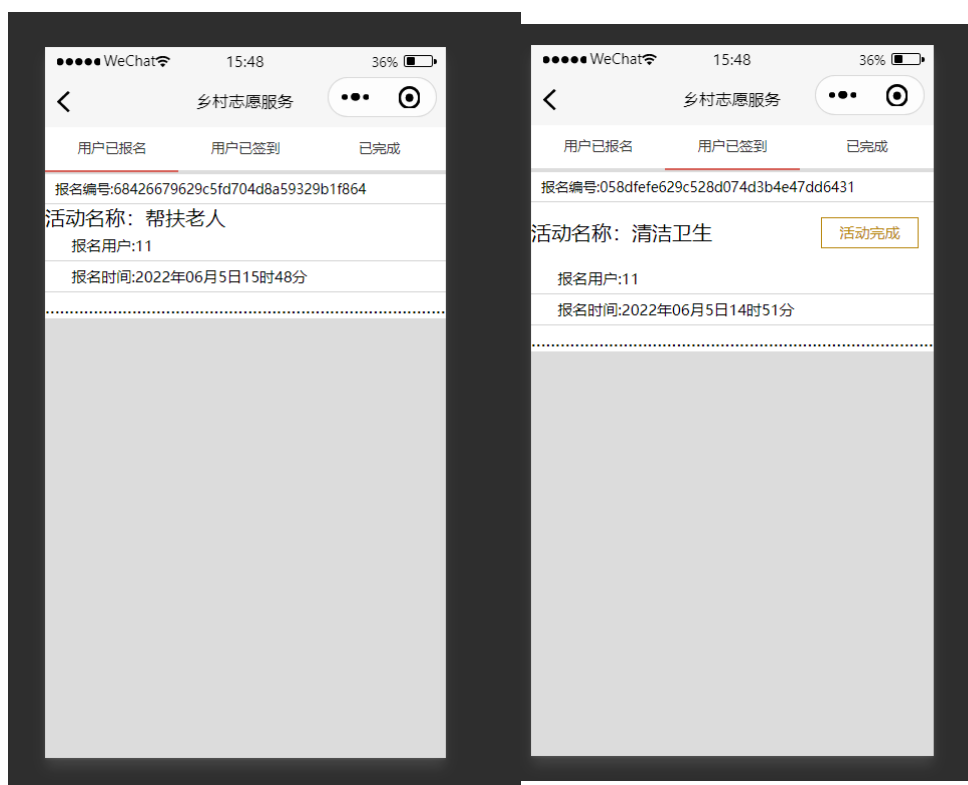


6.签到打卡：点击签到打卡页面进行签到





## 7.活动管理:

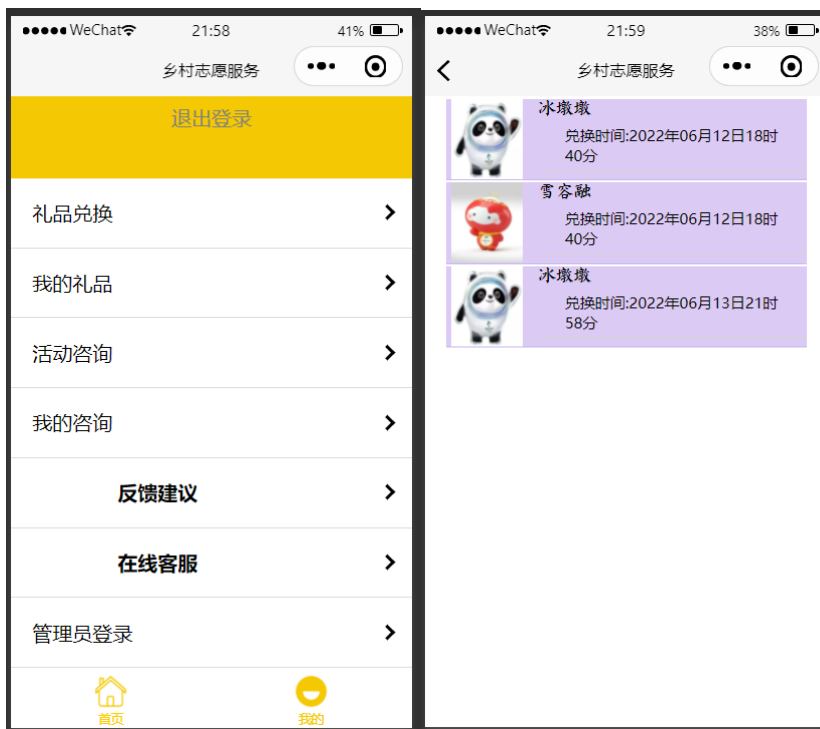




## 8.礼品兑换：点击礼品兑换，选择时长小于自己志愿时长的礼品



## 9.我的礼品：可查看自己已兑换的礼品



## 10.创建问题：用户可创建问题并查看自己的问题



## 11.发布者回答问题：发布者在管理员页面对问题进行管理

