

程序清单



课程名称 软件工程

题目名称 程序代码清单

任课教师 杨 枨

组 号： G18

组 长： 谢子文

组 员： 梁泽生

组 员： 黄 馨

2020 年12月 29 日

目录

**目录**

[一、 引言： 3](#_Toc19167)

[1.1编写目的 3](#_Toc29171)

[1.2背景 3](#_Toc6409)

[1.3定义 4](#_Toc23233)

[1.4参考资料 4](#_Toc3295)

[1.5版本历史 4](#_Toc31785)

[二、 功能模块代码： 5](#_Toc6927)

[2.1 用户认证登录模块： 5](#_Toc16124)

[2.2 专注计时模块： 7](#_Toc15491)

[2.3 统计总览（蒸笼）： 13](#_Toc1657)

[2.4 历史时间轴模块： 15](#_Toc9642)

[2.5 排行榜模块： 17](#_Toc17100)

[2.6 商店模块： 19](#_Toc25974)

[2.7 设置模块： 20](#_Toc4277)

[2.7.1 设置界面 20](#_Toc4342)

[2.7.2 个人中心界面 22](#_Toc5927)

# 引言：

## 1.1编写目的

程序清单的编写目的是为了让想要了解这个小程序的人通过查看代码模块分类与相关程序来粗略知道这个小程序是怎么实现怎么运行的，让浏览者更加清晰的知道整个小程序的构架和框架。

## 1.2背景

说明：

1. 待开发的软件系统的名称：

《专注包子》（又名《FocusBuns》）

1. 本项目的任务提出者、开发者、用户及实现该软件的计算中心或计算机网络；

提出者：

任务发布者：杨枨教授

G18小组组长：谢子文

开发者：

G18小组所有成员：谢子文、黄馨、梁泽生

用户：（分为目标用户和其他用户）

面向自制力不强，需要外力帮助监督不玩手机、或者想要专注学习和工作的人们。

目标用户：建筑1801王宇键、法学1802王新、软工1802陈骁

其他用户：为随机用户，因这部分用户的采访是不固定的，并且这部分用户不想透露个人信息，所以为其他用户。

实现该微信小程序的计算中心：

小组三人个人的计算机：

Huawei matebook14、联想威6 2020、戴尔轻薄笔记本

操作系统：Windows® 10 32 / 64bit / 7 32 / 64bit / XP

处理器：

Intel® Core™ 2 Duo E6600 or AMD Phenom™ X3 8750 processor or better

内存：2 GB RAM or more

显卡：Video card must be 256MB or more

实现该微信小程序的计算机网络：任何网络。

1. 该软件系统同其他系统或其他机构的基本的相互来往关系。

（1）微信小程序官方管理系统

（2）所使用开发软件的所有者机构

（3）github版本管理远程库

## 1.3定义

列出本文件中用到的专门术语的定义和外文首字母组词的原词组。

## 1.4参考资料

[1] 《GB856T——88》国标文档

[2] 《软件工程导论》清华大学出版社，张海藩、牟永敏编著

[3] 代码清单模板

<https://max.book118.com/html/2016/0419/40809642.shtm>

## 1.5版本历史

|  |  |  |  |
| --- | --- | --- | --- |
| 版本号 | 版本修订日期 | 版本修改人 | 是否通过审核 |
| 程序清单v0.1.0 | 2020/12/29 | 黄馨 | 是 |
| 程序清单v0.1.1 | 2020/12/29 | 黄馨 | 是 |
| 程序清单v0.1.2 | 2021/1/3 | 黄馨 | 是 |

# 功能模块代码：

## 2.1 用户认证登录模块：

**app.js:**

//app.js

App({

  onLaunch: function () {

    // 展示本地存储能力

    var logs = wx.getStorageSync('logs') || []

    logs.unshift(Date.now())

    wx.setStorageSync('logs', logs)

    // 登录

    wx.login({

      success: res => {

        // 发送 res.code 到后台换取 openId, sessionKey, unionId

      }

    })

    // 获取用户信息

    wx.getSetting({

      success: res => {

        if (res.authSetting['scope.userInfo']) {

          // 已经授权，可以直接调用 getUserInfo 获取头像昵称，不会弹框

          wx.getUserInfo({

            success: res => {

              // 可以将 res 发送给后台解码出 unionId

              this.globalData.userInfo = res.userInfo

              // 由于 getUserInfo 是网络请求，可能会在 Page.onLoad 之后才返回

              // 所以此处加入 callback 以防止这种情况

              if (this.userInfoReadyCallback) {

                this.userInfoReadyCallback(res)

              }

            }

          })

        }

      }

    })

  },

  globalData: {

    userInfo: null

  }

})

**app.json:**

{

  "pages": [

    "pages/main\_page/main\_page",

    "pages/adminpage/adminpage",

    "pages/timeline/timeline",

    "pages/shop/shop",

    "pages/mycenter/mycenter",

    "pages/setting/setting",

    "pages/steamer/steamer",

    "pages/ranking/ranking",

    "pages/index/index",

    "pages/logs/logs",

    "pages/adminpage/usertbl/usertbl"

  ],

  "window": {

    "backgroundTextStyle": "light",

    "navigationBarBackgroundColor": "#856E56",

    "navigationBarTitleText": "专注包子",

    "navigationBarTextStyle": "white"

  },

  "style": "v2",

  "sitemapLocation": "sitemap.json",

  "plugins": {

    "calendar": {

      "version": "1.1.3",

      "provider": "wx92c68dae5a8bb046"

    }

  }

}

## 2.2 专注计时模块：

**main\_page.wxml:**

<!--pages/main\_page/main\_page.wxml-->

<!-- 备注：wx:if="{{isTiming == 0}}" 代码块的作用是用来判断是否处于计时状态(0不处于计时状态，1处于计时状态)，以此来显示不同的界面 -->

<!-- 打开菜单后半透明灰灰的部分 英文:mask -->

<view wx:if='{{mask\_show}}' class="mask" bindtap="closeSideMenu"></view>

<!-- 侧滑菜单栏的部分 -->

<view wx:if='{{sideMenu\_show}}' class="sideMenu">

  <ul>

    <li>

      <navigator url="../steamer/steamer">

        <image></image><text>蒸笼</text>

      </navigator>

    </li>

    <li>

      <navigator url="../timeline/timeline">

        <image></image><text>时间轴</text>

      </navigator>

    </li>

    <li>

      <navigator url="../ranking/ranking">

        <image></image><text>排行榜</text>

      </navigator>

    </li>

    <li>

      <navigator url="../shop/shop">

        <image></image><text>商店</text>

      </navigator>

    </li>

    <li>

      <navigator>

        <image></image><text>同桌</text>

      </navigator>

    </li>

    <li>

      <navigator url="../setting/setting">

        <image></image><text>设置</text>

      </navigator>

    </li>

  </ul>

</view>

<!-- 主页面主要内容 -->

<view class="page">

  <!-- 打开侧滑菜单栏的按钮（用view仿制的button） -->

  <view class="btn\_showSideMenu" bindtap="showSideMenu" wx:if="{{isTiming==0}}">

    <image src="/image/菜单.png"></image>

    <view>菜单</view>

  </view>

  <!-- 计时模式切换部分 -->

  <view class="modeSwitch\_ui" wx:if="{{isTiming==0}}">

    <view class="tips">左右滑动可以切换其他计时模式</view>

    <swiper class="modeSwiper" bindchange="swiperChange" current="{{currentTab}}">

      <!-- 倒计时模式 -->

      <swiper-item>

        <view class="timingMode">

          <view class="modeName">倒计时</view>

        </view>

      </swiper-item>

      <!-- 定点计时模式 -->

      <swiper-item>

        <view class="timingMode">

          <view class="modeName">定点计时</view>

        </view>

      </swiper-item>

    </swiper>

  </view>

  <!-- 计时进行时显示的界面 -->

  <view class="timing\_ui" wx:if="{{isTiming==1}}">

  </view>

  <!-- 时间显示部分 -->

  <view class="timeset\_ui">

    <view class="tips2" wx:if="{{isTiming==0}}">Tips：点击下方的时间区域进行设置</view>

    <view class="tips2\_timing" wx:if="{{isTiming==1}}">Tips：专注进行中，不要玩手机哦</view>

    <picker class="picker" bindchange="bindPickerChange" value="{{timeindex}}" range="{{timesetArray}}">

        <text class="text\_timeset">{{UI\_Timeset}}</text> <!-- 动态数据 -->

        <text class="text\_minutes">分钟</text>

    </picker>

  </view>

  <!-- 开始按钮部分 -->

  <view class="startButton\_ui">

    <button class="btn\_start btn\_timing" bindtap="startTiming" wx:if="{{isTiming == 0}}">开始</button>

    <button class="btn\_stop btn\_timing" bindtap="stopTiming" wx:if="{{isTiming == 1}}">停止</button>

  </view>

</view>

**main\_page.js:**

// pages/main\_page/main\_page.js

Page({

  data: {

    /\* 侧边菜单栏的data \*/

    mask\_show:false, //遮罩显示与否

    sideMenu\_show: false, //侧边菜单显示与否

    /\* swiper左右滑动模块需要用到的data \*/

    currentTab:0,

    indicatorDots: true,

    vertical: false,

    autoplay: false,

    circular: false,

    interval: 2000,

    duration: 500,

    previousMargin: 0,

    nextMargin: 0,

    /\* 设置时间的data \*/

    timesetArray: [ //picker选择时间用的数组

      '10:00', '15:00', '20:00', '25:00', '30:00', '35:00', '40:00', '45:00',

      '50:00', '55:00', '60:00',  '65:00', '70:00', '75:00', '80:00', '85:00',

      '90:00', '95:00', '100:00', '105:00',  '110:00', '115:00', '120:00'

    ],

    timesetArray\_num: [ //timesetArray对应的number类型的分钟数

      10, 15, 20, 25, 30, 35, 40, 45,

      50, 55, 60, 65, 70, 75, 80, 85,

      90, 95, 100, 105, 110, 115, 120

    ],

    timeindex:0, //picker选择的序号，可以用在 timesetArray 和 timesetArray\_num 数组中作为下标

    UI\_Timeset:'10:00', //初始化为10分钟

    isTiming:0, //是否在计时

    isBindBtnStop:0, //是否按下了停止按钮

    totalsecond:0 //总时间换算成秒为单位

  },

  /\* 点击按钮打开菜单的函数 \*/

  showSideMenu:function(){

    if(!this.data.sideMenu\_show){

      this.setData({

        sideMenu\_show: true,

        mask\_show: true

      })

    }

  },

  /\* 点击其他地方关闭菜单的函数 \*/

  closeSideMenu:function(){

    if(this.data.sideMenu\_show){

      this.setData({

        sideMenu\_show: false,

        mask\_show: false

      })

    }

  },

  swiperChange:function(e){

    this.setData({

      currentTab: e.detail.current,

    })

    var that = this;

    var currentTab = that.data.currentTab

    console.log(currentTab)

  },

  /\* 设置时间后及时更新picker \*/

  bindPickerChange: function(e) {

    var that = this;

    console.log('picker发送选择改变，携带值为', e.detail.value)

    var timeindex = e.detail.value

    var timesetArray = that.data.timesetArray

    this.setData({

      timeindex: timeindex,

      UI\_Timeset: timesetArray[timeindex]

    })

  },

  /\* 开始计时函数 startTiming \*/

  startTiming:function(){

    var that = this;

    that.setData({

      isTiming: 1,

      totalsecond: that.data.timesetArray\_num[that.data.timeindex] \* 60 - 1

    })

    var totalsecond = that.data.totalsecond

    var minute = that.data.timesetArray\_num[that.data.timeindex] //分钟数

    var second = totalsecond % 60 //代表的是总时间去掉分钟的秒数零头

    /\* 设定一个定时器。按照指定的周期（以毫秒计）来执行注册的回调函数 \*/

    var i = setInterval(function(){

      //每次定时器执行时，更新变量

      var totalsecond = that.data.totalsecond

      var isBindBtnStop = that.data.isBindBtnStop

      second = totalsecond % 60

      minute = (totalsecond - second) / 60

      totalsecond--

      //判断 是继续计时并更新界面 还是停止计时

      if( totalsecond>=0 && (isBindBtnStop==0)){

        that.setData({

          totalsecond: totalsecond,

          UI\_Timeset: that.zeroFill(minute.toString(), 2)+":"+ that.zeroFill(second.toString(), 2)

        })

      }else{

        clearInterval(i)

        that.setData({

          UI\_Timeset: that.data.timesetArray[that.data.timeindex],

          isTiming:0

        })

        if(isBindBtnStop==1){

          console.log("弹窗告知用户专注失败")

          that.setData({isBindBtnStop:0}) //刷新状态

        }

      }

    }, 1000)

  },

  /\* 停止计时函数 stopTiming \*/

  stopTiming:function(){

    var that = this;

    wx.showModal({

      title: '警告',

      content: '确定要终止计时吗？这样会增加一条失败的专注记录！',

      success(res){

        if (res.confirm) {

          console.log('用户点击确定')

          that.setData({

            totalsecond:0,

            isBindBtnStop:1,

          })

        } else if (res.cancel) {

          console.log('用户点击取消')

        }

      }

    })

  },

  /\* 计时函数用的补零函数 \*/

  zeroFill(str, n){

    //补零方法，str为数字字符串 n为需要的位数，不够补零

    if (str.length < n){

      str = '0'+str;

    }

    return str;

  },

})

## 2.3 统计总览（蒸笼）：

**steamer.wxml:**

<!--pages/steamer\_v2/steamer\_v2.wxml-->

<!-- 选择时间 picker -->

<view class="tips1">点击下方的日期进行选择:</view>

<view class="chooseDate\_ui">

  <picker mode="date" value="{{date}}" start="2000-09" end="2099-12" fields="month" bindchange="bindDateChange">

    <text>{{date}}</text>

  </picker>

</view>

<view class="movableArea\_ui">

  <movable-area>

    <view class="tips2">按住小球可以进行拖动哦</view>

    <block wx:for="{{bunsList}}">

      <movable-view x="{{300/bunsList\_length\*index}}" y="{{300/bunsList\_length\*index}}" direction="all" out-of-bounds inertia >{{item.minute}}</movable-view>

    </block>

  </movable-area>

</view>

<view class="statistics">

  <view class="totalTime statistics-item">

    <view class="v1">{{totalTime}}</view>

    <view class="v2">分钟</view>

  </view>

  <view class="completeNum statistics-item">

    <view class="v1">{{completeNum}}</view>

    <view class="v2">完成</view>

  </view>

  <view class="failedNum statistics-item">

    <view class="v1">{{failedNum}}</view>

    <view class="v2">失败</view>

  </view>

</view>

**steamer.js:**

// pages/steamer\_v2/steamer\_v2.js

Page({

  data: {

    date: '2020-12',

    totalTime:300, // 总时间

    completeNum: 6, // 完成数

    failedNum: 0, // 失败数

    bunsList:[

      {minute:10},{minute:30},{minute:50},{minute:20},{minute:10},

      {minute:40},{minute:10},{minute:30},{minute:20},{minute:30}

    ],

    bunsList\_length:0, //数组长度

  },

  bindDateChange(e) { // 日期改变函数

    this.setData({

      date: e.detail.value

    })

  },

  /\*生命周期函数--监听页面加载\*/

  onLoad: function (options) {

    var that = this;

    that.setData({

      bunsList\_length: that.data.bunsList.length

    })

  },

})

## 2.4 历史时间轴模块：

**timeline.wxml:**

<!--pages/timeline/timeline.wxml-->

<view class="view\_calendar">

  <calendar calendar-style="calendar" header-style="calendarHeader" board-style="calendarBoard"

    weeks-type="cn" days-color="{{dayStyle}}"

    binddayClick="dayClick" bindnextMonth="nextClick" bindprevMonth="prevClick">

  </calendar>

</view>

<block class="block\_recordList">

  <view class="view\_recordList" wx:for="{{recordList}}" wx:for-index="index" wx:for-item="item" wx:key="record\_id">

    <view class="listItem">

      <view>2020年01月01日</view>

      <view>完成了一个120分钟的包子</view>

    </view>

  </view>

</block>

**timeline.js:**

// pages/timeline/timeline.js

Page({

  /\* 页面的初始数据 \*/

  data: {

    nowYear: new Date().getFullYear(), // 年份

    nowMonth: new Date().getMonth() + 1, // 月份

    nowDay: new Date().getDate(),

    calendarYear: 0, // 年份

    calendarMonth: 0, // 月份

    calendarDay: 0,

    dayStyle: [

      { month: 'current', day: 0, color: 'white', background: '#AAD4F5' }, //今天的日期 dayStyle[0]

      { month: 'current', day: 0, color: 'white', background: '#84E7D0' }, //被选中的日期 dayStyle[1]

    ],

    recordList: [ /\* 记录列表 \*/

      {item:1},{},{},{},{},{},{},{},{},{},{},{},{}

    ]

  },

  dayClick:function (event) { /\* 选中的日期 \*/

    // this.checkDate(event)

  },

  nextClick: function(event){

    console.log(event)

    // this.checkDate(event)

  },

  prevClick:function(event){

    // this.checkDate(event)

  },

  /\* 检查日期 \*/

  checkDate(event){

    if(1){

      console.log("run here")

    }else{

    }

  },

})

## 2.5 排行榜模块：

**ranking.wxml:**

<!--pages/ranking/ranking.wxml-->

<!-- 头部提示部分 -->

<view class="tips-container">

  <view class="tips-text">本回合剩余时间</view>

  <view class="tips-time">06日11时04分</view>

</view>

<!-- 排名列表部分 -->

<view class="ranking-container">

  <view class="ranking-item" wx:for="{{rankingList}}" wx:for-index="index" wx:for-item="user" wx:key="user\_id">

    <view class="user-avatar">

      <view class="u\_avatar"><image></image></view>

    </view>

    <view class="user-name">

      <view class="u\_name u\_text">{{user.username}}</view>

    </view>

    <view class="user-info">

      <view class="u\_info u\_text">{{user.completed}}完成 {{user.failed}}失败</view>

    </view>

    <view class="user-rank">

      <view class="u\_rank">

        {{index + 1}}

      </view>

    </view>

    <view class="user-time">

      <view class="u\_time u\_text">{{user.sumtime}}分钟</view>

    </view>

  </view>

</view>

**ranking.js:**

// pages/ranking/ranking.js

Page({

  data: {

    rankingList:[

      {

        username:"Nanami",

        sumtime:2249,

        completed:116,

        failed:0

      },

      {

        username:"麋鹿",

        sumtime:2249,

        completed:415,

        failed:3

      },

      {

        username:"Teeth",

        sumtime:2249,

        completed:167,

        failed:0

      },

      {

        username:"Nicole Wu",

        sumtime:2249,

        completed:112,

        failed:0

      },

      {

        username:"麦香鱼",

        sumtime:2249,

        completed:148,

        failed:0

      },

      {

        username:"dr.Balaji",

        sumtime:2249,

        completed:115,

        failed:0

      },

      {},

      {},

      {},

      {},

      {},

      {}

    ]

  },

})

## 2.6 商店模块：

**shop.wxml:**

<!--pages/shop/shop.wxml-->

<view class="goods-container">

  <view class="goods-for" wx:for="{{goodsList}}" wx:for-index="index" wx:for-item="item">

    <view class="goods">

      <view class="goods-pic"></view>

      <text class="goods-name">蒸品名称</text>

    </view>

    <view class="iconPrice">

      <view class="icon\_price"></view>

      <!-- goods-price 动态数据 -->

      <view class="goods-price">600</view>

    </view>

  </view>

</view>

**shop.js:**

// pages/shop/shop.js

Page({

  data: {

    goodsList:[

      {},{},{},{},{},{},{},{},{},{},{},{}

    ]

  }

})

## 2.7 设置模块：

### 2.7.1 设置界面

**setting.wxml:**

<!--pages/setting/setting.wxml-->

<view class="container">

  <view class="view\_personalCenter float">

    <navigator url="../mycenter/mycenter">

      <view class="view\_head float">

        <open-data type="userAvatarUrl"></open-data>

      </view>

      <view class="view\_name float">

        <open-data type="userNickName"></open-data>

        <view class="view\_uid">

          userid

        </view>

      </view>

      <view class="view\_arrow">

        <image src='/pages/setting/picture/jiantou.png'></image>

      </view>

    </navigator>

  </view>

  <view class="view\_focusSetting">

    <view class="view\_focusTitle line1 flex">

      <view class="view\_setPicture">

        <image src="/pages/setting/picture/shezhi.png"></image>

      </view>

      <view class="view\_setWord">专注设置</view>

    </view>

    <view class="view\_encourage line1">

      <view class="view\_statement" style="color:{{style\_fontColor}}">

        自定义鼓励语句

      </view>

    </view>

  </view>

  <view class="view\_aboutUs">

    <view class="view\_aboutTitle line2 flex">

      <view class="view\_usPicture">

        <image src="/pages/setting/picture/aboutus.png"></image>

      </view>

      <view class="view\_usWord">关于我们</view>

    </view>

    <view class="view\_developer line2">

      <view class="view\_interface common" style="color:{{style\_fontColor}}">

        开发者界面

      </view>

    </view>

    <view class="view\_userFeedback line2">

      <view class="view\_advice common" style="color:{{style\_fontColor}}">

        用户反馈

      </view>

    </view>

  </view>

</view>

**setting.js:**

// pages/setting/setting.js

Page({

  /\*\*

   \* 页面的初始数据

   \*/

  data: {

    style\_fontColor: "#808080"

  },

})

### 2.7.2 个人中心界面

**mycenter.wxml:**

<!--pages/mycenter/mycenter.wxml-->

<view class="container">

  <view class="view\_module">

    <view class="view\_head">

      <open-data type="userAvatarUrl"></open-data>

    </view>

    <view class="view\_name">

      <open-data type="userNickName"></open-data>

    </view>

    <view class="view\_focusTime">

      3天21小时50分钟（专注时长）

    </view>

    <view class="view\_smallModule">

      <view class="view\_successFailRatio box">

        完成/失败比

        <view class="view\_ratioValue value">86%</view>

      </view>

      <view class="view\_totalFinished box">

        总完成数

        <view class="view\_finishValue value">189</view>

      </view>

      <view class="view\_achivementRatio box">

        成就完成比例

        <view class="view\_achivementValue value">25%</view>

      </view>

    </view>

  </view>

</view>

**mycenter.js:**

// pages/mycenter/mycenter.js

Page({

  /\*\*

   \* 页面的初始数据

   \*/

  data: {

  },

})