

微信订餐小程序项目实验报告

一、实验目的

在 linux 系统中使用微信开发者工具开发微信订餐小程序

二、安装步骤

1.在 linux 系统中安装微信开发者工具

(1).安装 wine

```
sudo apt-get install wine
```

(2).安装 nwjs-sdk

下载 linux 版 nwjs-sdk

```
wget https://dl.nwjs.io/v0.25.4/nwjs-sdk-v0.25.4-linux-x64.tar.gz
```

解压 nwjs-sdk

```
tar xvf nwjs-sdk-v0.25.4-linux-x64.tar.gz
```

切换到 nwjs-sdk 对应的目录

```
cd nwjs-sdk-v0.25.4-linux-x64
```

启动 nwjs-sdk

```
./nw (测试是否能正常运行,测试正常就关掉)
```

(3)安装微信开发工具包(基于 nwjs-sdk)

获取微信开发工具包

```
git clone https://github.com/cytle/wechat_web_devtools.git
```

切换到 wechat_web_devtools 目录

```
cd wechat_web_devtools
```

复制微信开发工具包到 nwjs-sdk 目录下

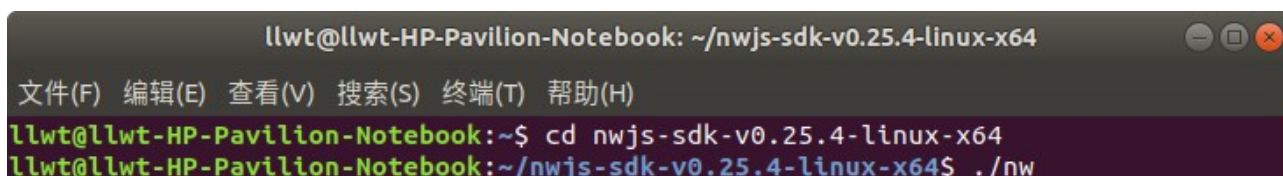
```
cp package.nw ~/development/nwjs-sdk-v0.25.4-linux-x64/ -rf
```

(4)启动

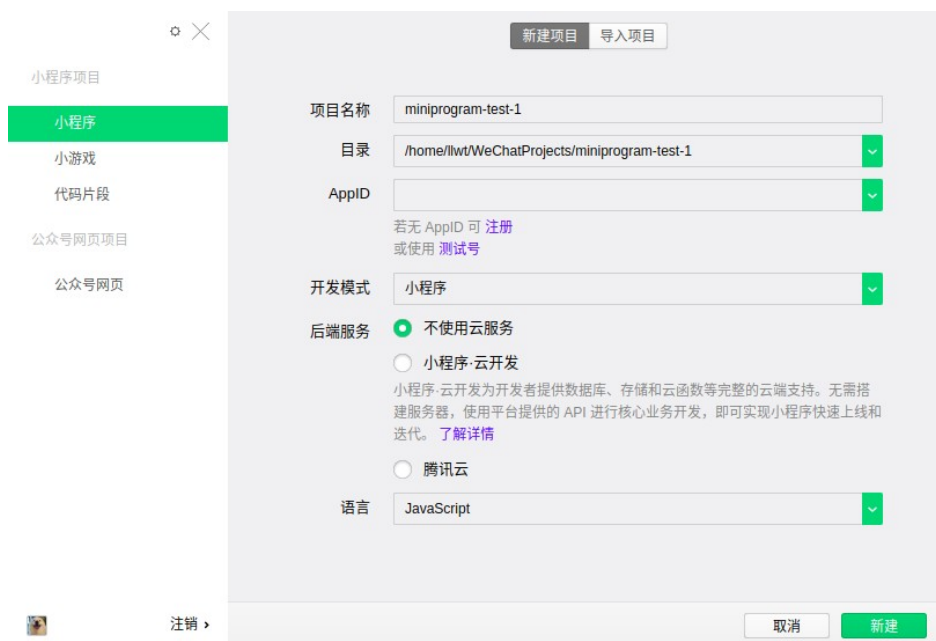
```
./nw
```

三、开发过程及部分代码实现

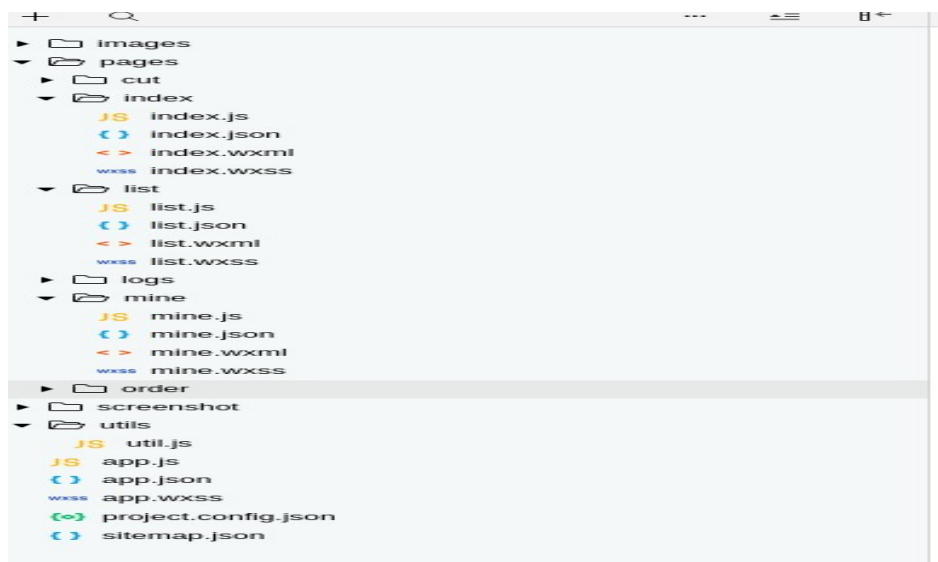
1、在终端输入命令打开微信开发者工具



2、创建新项目



3、页面设计及功能开发



(1)app.js 是小程序的脚本代码,可以在这个文件中监听并处理小程序的生命周期函数、声明全局变量，调用框架

```
1 //app.js
2 App({
3   onLaunch: function () {
4     // 展示本地存储能力
5     var logs = wx.getStorageSync('logs') || []
6     logs.unshift(Date.now())
7     wx.setStorageSync('logs', logs)
8
9     // 登录
10    wx.login({
11      success: res => {
12        // 发送 res.code 到后台换取 openId, sessionKey, unionId
13      }
14    })
15    // 获取用户信息
16    wx.getSetting({
17      success: res => {
18        if (res.authSetting['scope.userInfo']) {
19          // 已经授权，可以直接调用 getUserInfo 获取头像昵称，不会弹框
20          wx.getUserInfo({
21            success: res => {
22              // 可以将 res 发送给后台解码出 unionId
23              this.globalData.userInfo = res.userInfo
24
25              // 由于 getUserInfo 是网络请求，可能会在 Page.onLoad 之后才返回
26              // 所以此处加入 callback 以防止这种情况
27              if (this.userInfoReadyCallback) {
28                this.userInfoReadyCallback(res)
29              }
30            }
31          })
32        }
33      }
34    })
35  },
36  globalData: {
37    userInfo: null
38  }
39 })
```

(2)app.json 是对整个小程序的全局配置，用来对微信小程序进行全局配置，决定页面文件的路径、窗口表现、设置网络超时时间、设置多 tab 等。接受一个数组，每一项都是字符串，来指定小程序由哪些页面组成。

(3)app.wxss 是整个小程序的公共样式表

```
app.json  x  app.js  mine.js  mine.wxmi  mine.wxss  logs.json  mine.json  index.wxss  detail.wxmi
1  {
2    "pages": [
3      "pages/index/index",
4      "pages/logs/logs",
5      "pages/list/list",
6      "pages/order/balance/balance",
7      "pages/mine/mine",
8      "pages/order/list/list",
9      "pages/order/detail/detail"
10   ],
11   "window": {
12     "backgroundTextStyle": "light",
13     "navigationBarBackgroundColor": "#FF9C35",
14     "navigationBarTitleText": "拾饭",
15     "navigationBarTextStyle": "white"
16   },
17   "tabBar": {
18     "color": "#8a8a8a",
19     "selectedColor": "#FF9C35",
20     "borderStyle": "black",
21     "list": [
22       {
23         "selectedIconPath": "images/home_s.png",
24         "iconPath": "images/home.png",
25         "pagePath": "pages/index/index",
26         "text": "首页"
27       },
28       {
29         "selectedIconPath": "images/order_s.png",
30         "iconPath": "images/order.png",
31         "pagePath": "pages/order/list/list"
32       },
33       {
34         "selectedIconPath": "images/mine_s.png",
35         "iconPath": "images/mine.png",
36         "pagePath": "pages/mine/mine",
37         "text": "我的"
38       }
39     ]
40   },
41   "sitemapLocation": "sitemap.json"
42 }
```

/app.json 1.0 KB 行 31, 列 45 JSO

(4)index.js 是页面的脚本文件,可以监听并处理页面的生命周期函数、获取小程序实例，声明并处理数据，响应页面交互事件等。

index.wxml 是页面结构文件

index.wxss 是页面样式表文件

index 最终实现界面



(5) list 为购物车及点单界面

list.js 从 easy-mock 网站中生成模拟数据

```

15     currentIndex: 0,
16     sizeIndex: 0,
17     sugarIndex: 0,
18     temIndex: 0,
19     sugar: ['常规糖', '无糖', '微糖', '半糖', '多糖'],
20     tem: ['常规冰', '多冰', '少冰', '去冰', '温', '热'],
21     size: ['常规', '珍珠', '西米露'],
22     cartList: [],
23     sumMonney: 0,
24     cupNumber: 0,
25     showCart: false,
26     loading: false
27   },
28
29   /**
30    * 生命周期函数--监听页面加载
31    */
32   onLoad: function (options) {
33     var that = this;
34     var sysinfo = wx.getSystemInfoSync().windowHeight;
35     console.log(sysinfo)
36     wx.showLoading({
37       title: '努力加载中',
38     })
39     //将本来的后台换成了easy-mock的接口，所有数据一次请求完 略大。。
40     wx.request({
41       url: 'https://easy-mock.com/mock/59abab95e0dc66334199cc5f/coco/aa',
42       method: 'GET',
43       data: {},
44       header: {
45         'Accept': 'application/json'
46       },
47       success: function (res) {
48         wx.hideLoading();
49         console.log(res)
50         that.setData({
51           listData: res.data,
52           loading: true
53         })
54       }
55     })
56   }
57 }

```

/pages/list/list.js 6.0 KB 行 16, 列 18 JavaS

list 订单最终实现界面

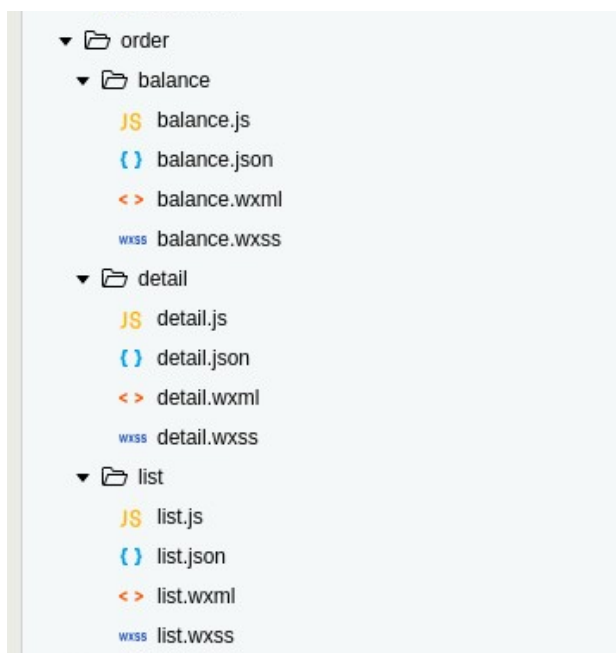


(6) mine 个人信息界面



(7) order 订单详细信息界面

其中包含 balance,detail,list



balance 为订单详情界面



detail 为下单界面

点击上图去支付即可跳转



list 为底部订单按钮界面实现

点击开始点单即可跳转到 list 界面

