

学习和使用uni-app开发可以在手机端运行的应用客户端-2

1.新建登录页面

表单组件见: <https://v1.uviewui.com/components/form.html>

v-model 指令: <https://cn.vuejs.org/guide/essentials/forms.html>

可以简单了解: <https://www.runoob.com/vue2/vue-tutorial.html>

自己目录下创建 login.vue

```
<template>
  <view class="wrap">
    <u-form :model="model" ref="uForm" >
      <u-form-item left-icon="account" label-width="120" label="姓名"
prop="name">
        <u-input placeholder="请输入姓名" v-model="model.zhanghao"
type="text"></u-input>
      </u-form-item>
      <u-form-item left-icon="lock" label-width="120" label="密码"
prop="password">
        <u-input :password-icon="true" type="password" v-
model="model.mima" placeholder="请输入密码"></u-input>
      </u-form-item>
      <!-- 此处switch的slot为right, 如果不填写slot名, 也即<u-switch v-
model="model.remember"></u-switch>, 将会左对齐 -->
      <u-form-item label="记住密码" prop="remember" label-width="150">
        <u-switch v-model="model.remember" slot="right"></u-switch>
      </u-form-item>
    </u-form>
    <u-button @click="submit">提交</u-button>
    <view class="u-demo-wrap">
      <view class="u-demo-area">
        <u-link :color="color" :font-size="fontSize" :under-
line="underLine" :href="href">注册</u-link>
      </view>
    </view>

  </view>
</template>

<script>
  export default {
    data() {
      return {
        model: {
          zhanghao: '',
          mima: ''
        },
        href: 'https://www.uviewui.com', //将来换成注册页面
        underLine: true,
        fontSize: 28,
      }
    }
  }
}
```

```

        color: '#2979ff'
      }
    },
    methods: {
      submit(){
        console.log("提交登录")
        //如果成功，返回
        uni.navigateBack()
      }
    }
  }
</script>

<style>
.u-demo-area{
  cursor: pointer;
}
</style>

```

pages.json修改

```

{
  "path" : "pages/gehm/login",
  "style" :
  {
    "navigationBarTitleText": "登录",
    "enablePullDownRefresh": false
  }
}

```

从我的信息，编写页面跳转

<https://uniapp.dcloud.net.cn/api/router.html#navigateto>

登录处添加事件处理

```

<view class="u-m-t-20">
  <u-cell-group>
    <u-cell-item icon="man-add" title="登录" @click="goLogin()">
  </u-cell-item>
  </u-cell-group>
</view>

```

```
//methods处
methods: {
    goLogin(){
        uni.navigateTo({
            url: '/pages/geom/login'
        })
        /*
        //或者
        this.$u.route("/pages/geom/login")
        */
    }
}
```

关于js函数:

<https://www.liaoxuefeng.com/wiki/1022910821149312/1023020745357888>

2.JavaScript函数

2.1具名函数（函数声明）

```
function qiuhe(a,b){
    console.log(a + b);
    return a + b
}
```

2.2函数表达式

```
let qiuhe = function(a,b){
    return a + b
}
```

2.3箭头函数

```
//2.2中 省略function关键字 参数和函数体之间用箭头分隔
let qiuhe = (a,b)=>{
    return a + b
}
//当函数体仅有一条语句时，可以省略大括号{}，且默认返回这行代码的值
let qiuhe = (a,b)=>a+b
//当函数的参数为0个或多个时，不能省略小括号
//当仅有一个参数时，可以省略小括号，例如：
let square=function(a){
    return a * a
}
let square = a=>a * a
```

2.4回调函数

```
let sum = function(a,b){
    return a+b
}
function dec(a,b){
    return a - b
}
```

```

}
function doit(a,b,fun){
    return fun(a,b)
}
//函数也是变量，也可以当参数传 看上去doit回过去调用sum,所以称为callback，回调
let re = doit(11,12,sum)//23
let re2 = doit(11,12,dec)//-1
let re3 = doit(10,20,function(a,b){
    return a*b;
}); //200

```

3.完成登录功能

3.1登录页面使用uni-app的api访问网络

```

methods: {
    submit(){
        uni.request({
            url: 'http://localhost:8080/doLogin',
            data: this.model,
            method:"POST",
            success: (res) => { //我们返回的Result对象 在 res.data中
                console.log(res);
            }
        });
        console.log("提交登录")
        //如果成功，返回
        uni.navigateBack()
    }
}

```

3.2 登录成功和失败的处理

```

methods: {
    submit(){
        uni.request({
            url: 'http://localhost:8080/doLogin',
            data: this.model,
            method:"POST",
            success: (res) => { //返回的结果（Result）对象
                {"code":200,"result":...} 在res.data中
                if(res.data.code*1 == 200){ //成功登录
                    try {
                        //见：
                        https://uniapp.dcloud.net.cn/api/storage/storage.html#setstoragesync
                        uni.setStorageSync('user', res.data.result); //将
                        用户对象本地存储 以便后续身份识别 权限验证等
                        uni.navigateBack() //登录成功返回 我的个人信息页面
                    } catch (e) {
                        this.$u.toast('身份信息格式异常')
                    }
                }else{
                    this.$u.toast('登录失败，用户名密码错误') //提示框
                }
            }
        });
    }
}

```

```

    }
  });
}
}

```

3.3 完成个人信息页面

3.3.1 每次显示页面检查登录状态

我的信息页面中，onShow()方法中处理

<https://uniapp.dcloud.net.cn/tutorial/page.html#lifecycle>

```

<script>
  export default {
    data() {
      return {
        logged: false
      }
    },
    onLoad() {

    },
    onShow() { //*****
      try { //见
https://uniapp.dcloud.net.cn/api/storage/storage.html#getstoragesync
        const value = uni.getStorageSync('user')
        if (value) {
          this.logged = true
        } else {
          this.logged = false
        }
      } catch (e) {
        // error
      }
    },
    methods: {
      goLogin() {
        uni.navigateTo({
          url: '/pages/gehm/login'
        })
      }
    }
  }
}
</script>

```

3.3.2 页面内容绑定真正的用户数据

见: <https://www.runoob.com/vue2/vue-template-syntax.html>

```

<script>
  export default {
    data() {
      return {
        logged: false,
        yonghu: { mingzi: "游客" } //初始
      }
    }
  }

```

```

    }
  },
  onShow() {
    try { //见
https://uniapp.dcloud.net.cn/api/storage/storage.html#getstoragesync
      const value = uni.getStorageSync('user')
      if (value) {
        if(!this.logged){//从未登录变为登录
          this.yonghu = value
        }
        this.logged = true
      }else{
        this.logged = false
      }
    } catch (e) {
      // error
    }
  },
  methods: {
    goLogin() {
      uni.navigateTo({
        url: '/pages/gehm/login'
      })
    }
  }
}
</script>

```

页面内容：

```

<view class="u-flex-1">
  <view class="u-font-18 u-p-b-20">{{yonghu.xingming}}</view>
  <view class="u-font-14 u-tips-color">粉丝数: 11</view>
</view>

```

3.3.3 退出功能

```

methods: {
  goLogin() {
    uni.navigateTo({
      url: '/pages/gehm/login'
    })
  },
  quit(){//增加
    uni.setStorageSync("user",""); //清空 存储
    this.logged = false //登陆状态
  }
}

```

页面内容，将“设置”改成退出

```

<view class="u-m-t-20">
    <u-cell-group>
        <u-cell-item icon="man-delete" title="退出" @click="quit">
    </u-cell-item>
    </u-cell-group>
</view>

```

4.注册页面

4.1 新建register注册页面

```

<template>
    <view class="wrap">
        <u-form :model="user" ref="uForm">
            <u-form-item label-width="140" label-position="left" label="姓名"
prop="xingming">
                <u-input :border="border" placeholder="请输入姓名" v-
model="user.xingming" type="text"></u-input>
            </u-form-item>
            <u-form-item label-width="140" label-position="left" label="用户名"
prop="zhanghao">
                <u-input :border="border" placeholder="请输入用户名" v-
model="user.zhanghao" type="text"></u-input>
            </u-form-item>
            <u-form-item label-width="140" label-position="left" label="密码"
prop="mima">
                <u-input :password-icon="true" :border="border" type="password"
v-model="user.mima"
                placeholder="请输入密码"></u-input>
            </u-form-item>
            <u-form-item label-width="140" label-position="left" label="确认密码"
prop="rmima">
                <u-input :password-icon="true" :border="border" type="password"
v-model="user.rmima"
                placeholder="再输一次密码"></u-input>
            </u-form-item>
            <u-form-item label-width="140" label-position="left" label="年龄"
prop="age">
                <u-input :border="border" placeholder="请输入年龄" v-
model="user.age" type="text"></u-input>
            </u-form-item>
            <u-form-item label-width="140" label-position="left" label="用户类别"
prop="leibie">
                <radio-group @change="radioChange" class="uni-padding-wrap">
                    <label class="u-radio">
                        <radio value="1" />
                        超级用户
                    </label>
                    <label class="u-radio">
                        <radio value="2" />
                        普通用户
                    </label>
                    <label class="u-radio">
                        <radio value="3" />

```

```

        电影推荐
      </label>
    </radio-group>
  </u-form-item>
</u-form>
<u-button @click="submit">提交</u-button>
</view>
</template>

<script>
  export default {
    data() {
      return {
        border: true,
        user: { }
      }
    },
    methods: {
      submit() {
        console.log(this.user)
      },
      radioChange(e) {
        //console.log(e.detail)
        this.user.leibie = e.detail.value
      }
    }
  }
</script>

<style>

</style>

```

pages.json更改 略.....

4.2 登录页面 点击注册按钮跳转到注册

```
<u-button @click="goRegiter">注册</u-button>
```

```

goRegiter(){
  uni.navigateTo({
    url:"/pages/gehm/register"
  })
}

```

4.3 为submit方法调用后端保存

未完待续