摘自 需求分析.doc：



信息采集设计表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 信息采集需填写内容 | 英文 | 检验规则 | 必填 | 备注 |
| 性别 | sex |  | 1 |  |
| 个人联系方式 | personal\_phone | 验证手机号格式 | 1 |  |
| 紧急联系人联系方式 | emergency\_phone | 验证手机号格式 | 1 |  |
| 户籍 | household\_regist | 选择省-市 | 1 |  |
| 家庭地址 | address | 选择省-市 | 1 |  |
| 兴趣爱好 | interest | 多选 | 1 | 唱 跳 rap 篮球 music |
| 个人简介 | profile | 3-140字之间 | 0 |  |
| 报道时间 |  | 选择日期 | 1 | 记录具体报道日期，代码中未显示 |
| 是否报名计算机卓越班 | is\_sign | 单选 | 1 | 是否报名计算机卓越班当人上人 |
| 个人照片 |  |  | 0 | 未定 |

代码：

<template>

<view class="wrap">

<div style="display: flex;background-color: aquamarine; height: 100vh; align-items: center; justify-content: center;">

<h1 style="color: blue; font-family:楷体 ">

你好,欢迎报考重庆大学

</h1>

</div>

<u-form :model="model" :rules="rules" ref="uForm" :errorType="errorType">

<u-form-item :label-position="labelPosition" label="性别" prop="sex">

<u-input :border="border" type="select" :select-open="actionSheetShow" v-model="model.sex" placeholder="请选择性别" @click="actionSheetShow = true"></u-input>

</u-form-item>

<u-form-item :rightIconStyle="{color: '#888', fontSize: '32rpx'}" right-icon="kefu-ermai" :label-position="labelPosition" label="个人联系方式" prop="personal\_phone" label-width="200">

<u-input :border="border" placeholder="请输入手机号码" v-model="model.personal\_phone" type="number"></u-input>

</u-form-item>

<u-form-item :rightIconStyle="{color: '#888', fontSize: '32rpx'}" right-icon="kefu-ermai" :label-position="labelPosition" label="紧急联系人联系方式" prop="emergency\_phone" label-width="200">

<u-input :border="border" placeholder="请输入手机号码" v-model="model.emergency\_phone" type="number"></u-input>

</u-form-item>

<u-form-item :label-position="labelPosition" label="户籍" prop="household\_regist" label-width="150">

<u-input :border="border" type="select" :select-open="pickerShow" v-model="model.household\_regist" placeholder="请选择地区" @click="pickerShow = true"></u-input>

</u-form-item>

<u-form-item :label-position="labelPosition" label="家庭地址" prop="address" label-width="150">

<u-input :border="border" type="select" :select-open="pickerShow\_addr" v-model="model.address" placeholder="请选择地区" @click="pickerShow\_addr = true"></u-input>

</u-form-item>

<u-form-item :label-position="labelPosition" label="兴趣爱好" label-width="150" prop="interest">

<u-checkbox-group @change="checkboxGroupChange" :width="radioCheckWidth" :wrap="radioCheckWrap">

<u-checkbox v-model="item.checked" v-for="(item, index) in checkboxList" :key="index" :name="item.name">{{ item.name }}</u-checkbox>

</u-checkbox-group>

</u-form-item>

<u-form-item :label-position="labelPosition" label="个人简介" prop="profile">

<u-input type="textarea" :border="border" placeholder="请填写简介" v-model="model.profile" />

</u-form-item>

<u-form-item :label-position="labelPosition" label="是否报名计算机卓越班" prop="is\_sign" label-width="150">

<u-radio-group v-model="radio" @change="radioGroupChange" :width="radioCheckWidth" :wrap="radioCheckWrap">

<u-radio shape="circle" v-for="(item, index) in radioList" :key="index" :name="item.name">{{ item.name }}</u-radio>

</u-radio-group>

</u-form-item>

<u-form-item :label-position="labelPosition" label="个人照片" prop="photo" label-width="150">

<u-upload width="160" height="160"></u-upload>

</u-form-item>

<!--

<u-form-item :leftIconStyle="{color: '#888', fontSize: '32rpx'}" left-icon="account" label-width="120" :label-position="labelPosition" label="姓名" prop="name">

<u-input :border="border" placeholder="请输入姓名" v-model="model.name" type="text"></u-input>

</u-form-item>

<u-form-item :label-position="labelPosition" label="性别" prop="sex">

<u-input :border="border" type="select" :select-open="actionSheetShow" v-model="model.sex" placeholder="请选择性别" @click="actionSheetShow = true"></u-input>

</u-form-item>

<u-form-item :label-position="labelPosition" label="简介" prop="profile">

<u-input type="textarea" :border="border" placeholder="请填写简介" v-model="model.profile" />

</u-form-item>

<u-form-item :label-position="labelPosition" label="密码" prop="password">

<u-input :password-icon="true" :border="border" type="password" v-model="model.password" placeholder="请输入密码"></u-input>

</u-form-item>

<u-form-item :label-position="labelPosition" label="确认密码" label-width="150" prop="rePassword">

<u-input :border="border" type="password" v-model="model.rePassword" placeholder="请确认密码"></u-input>

</u-form-item>

<u-form-item :label-position="labelPosition" label="水果品种" label-width="150" prop="interest">

<u-checkbox-group @change="checkboxGroupChange" :width="radioCheckWidth" :wrap="radioCheckWrap">

<u-checkbox v-model="item.checked" v-for="(item, index) in checkboxList" :key="index" :name="item.name">{{ item.name }}</u-checkbox>

</u-checkbox-group>

</u-form-item>

<u-form-item :label-position="labelPosition" label="结算方式" prop="is\_reg" label-width="150">

<u-radio-group v-model="radio" @change="radioGroupChange" :width="radioCheckWidth" :wrap="radioCheckWrap">

<u-radio shape="circle" v-for="(item, index) in radioList" :key="index" :name="item.name">{{ item.name }}</u-radio>

</u-radio-group>

</u-form-item>

<u-form-item :label-position="labelPosition" label="所在地区" prop="address" label-width="150">

<u-input :border="border" type="select" :select-open="pickerShow" v-model="model.address" placeholder="请选择地区" @click="pickerShow = true"></u-input>

</u-form-item>

<u-form-item :label-position="labelPosition" label="商品类型" prop="goodsType" label-width="150">

<u-input :border="border" type="select" :select-open="selectShow" v-model="model.goodsType" placeholder="请选择商品类型" @click="selectShow = true"></u-input>

</u-form-item>

<u-form-item :rightIconStyle="{color: '#888', fontSize: '32rpx'}" right-icon="kefu-ermai" :label-position="labelPosition" label="手机号码" prop="personal\_phone" label-width="150">

<u-input :border="border" placeholder="请输入手机号" v-model="model.personal\_phone" type="number"></u-input>

</u-form-item>

<u-form-item :label-position="labelPosition" label="验证码" prop="code" label-width="150">

<u-input :border="border" placeholder="请输入验证码" v-model="model.code" type="text"></u-input>

<u-button slot="right" type="success" size="mini" @click="getCode">{{codeTips}}</u-button>

</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-position="labelPosition" 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-item :label-position="labelPosition" label="上传图片" prop="photo" label-width="150">

<u-upload width="160" height="160"></u-upload>

</u-form-item>

-->

</u-form>

<view class="agreement">

<u-checkbox v-model="check" @change="checkboxChange"></u-checkbox>

<view class="agreement-text">

勾选代表同意本小组的版权协议

</view>

</view>

<u-button @click="submit">提交</u-button>

<u-action-sheet :list="actionSheetList" v-model="actionSheetShow" @click="actionSheetCallback"></u-action-sheet>

<u-select mode="single-column" :list="selectList" v-model="selectShow" @confirm="selectConfirm"></u-select>

<u-picker mode="region" v-model="pickerShow\_addr" @confirm="regionConfirm\_address"></u-picker>

<u-picker mode="region" v-model="pickerShow" @confirm="regionConfirm\_household\_regist"></u-picker>

<u-verification-code seconds="60" ref="uCode" @change="codeChange"></u-verification-code>

<view class="u-config-wrap">

<view class="u-config-title u-border-bottom">

参数配置

</view>

<view class="u-config-item">

<view class="u-item-title">label对齐方式</view>

<u-subsection :list="['左边', '上方']" @change="labelPositionChange"></u-subsection>

</view>

<view class="u-config-item">

<view class="u-item-title">边框</view>

<u-subsection :current="borderCurrent" :list="['显示', '隐藏']" @change="borderChange"></u-subsection>

</view>

<view class="u-config-item">

<view class="u-item-title">radio、checkbox样式</view>

<u-subsection :list="['自适应', '换行', '50%宽度']" @change="radioCheckboxChange"></u-subsection>

</view>

<view class="u-config-item">

<view class="u-item-title">错误提示方式</view>

<u-subsection :list="['message', 'toast', '下划线', '输入框']" @change="errorChange"></u-subsection>

</view>

</view>

</view>

</template>

<script>

export default {

data() {

let that = this;

return {

model: {

name: '',

sex: '',

interest: '',

profile: '',

is\_reg: '是',

agreement: false,

address: '',

household\_regist: '',

goodsType: '',

personal\_phone: '',

emergency\_phone:'',

code: '',

password: '',

rePassword: '',

remember: false,

photo: ''

},

selectList: [

{

value: '电子产品',

label: '电子产品'

},

{

value: '服装',

label: '服装'

},

{

value: '工艺品',

label: '工艺品'

}

],

rules: {

name: [

{

required: true,

message: '请输入姓名',

trigger: 'blur' ,

},

{

min: 3,

max: 5,

message: '姓名长度在3到5个字符',

trigger: ['change','blur'],

},

{

// 此为同步验证，可以直接返回true或者false，如果是异步验证，稍微不同，见下方说明

validator: (rule, value, callback) => {

// 调用uView自带的js验证规则，详见：https://www.uviewui.com/js/test.html

return this.$u.test.chinese(value);

},

message: '姓名必须为中文',

// 触发器可以同时用blur和change，二者之间用英文逗号隔开

trigger: ['change','blur'],

},

// 异步验证，用途：比如用户注册时输入完账号，后端检查账号是否已存在

// {

// trigger: ['blur'],

// // 异步验证需要通过调用callback()，并且在里面抛出new Error()

// // 抛出的内容为需要提示的信息，和其他方式的message属性的提示一样

// asyncValidator: (rule, value, callback) => {

// this.$u.post('/ebapi/public\_api/index').then(res => {

// // 如果验证出错，需要在callback()抛出new Error('错误提示信息')

// if(res.error) {

// callback(new Error('姓名重复'));

// } else {

// // 如果没有错误，也要执行callback()回调

// callback();

// }

// })

// },

// }

],

sex: [

{

required: true,

message: '请选择性别',

trigger: 'change'

},

],

profile: [

{

required: false,

message: '请填写简介'

},

{

min: 5,

message: '简介不能少于5个字',

trigger: 'change' ,

},

// 正则校验示例，此处用正则校验是否中文，此处仅为示例，因为uView有this.$u.test.chinese可以判断是否中文

{

pattern: /^[\u4e00-\u9fa5]+$/gi,

message: '简介只能为中文',

trigger: 'change',

},

],

interest: [

{

required: false,

message: '请选择您的爱好',

trigger: 'change',

type: 'array',

}

],

is\_reg: [

{

required: true,

message: '请选择',

trigger: 'change',

}

],

address: [

{

required: true,

message: '请选择地区',

trigger: 'change',

}

],

household\_regist: [

{

required: true,

message: '请选择地区',

trigger: 'change',

}

],

goodsType: [

{

required: true,

message: '请选择商品类型',

trigger: 'change',

}

],

personal\_phone: [

{

required: true,

message: '请输入手机号',

trigger: ['change','blur'],

},

{

validator: (rule, value, callback) => {

// 调用uView自带的js验证规则，详见：https://www.uviewui.com/js/test.html

return this.$u.test.mobile(value);

},

message: '手机号码不正确',

// 触发器可以同时用blur和change，二者之间用英文逗号隔开

trigger: ['change','blur'],

}

],

emergency\_phone: [

{

required: true,

message: '请输入手机号',

trigger: ['change','blur'],

},

{

validator: (rule, value, callback) => {

// 调用uView自带的js验证规则，详见：https://www.uviewui.com/js/test.html

return this.$u.test.mobile(value);

},

message: '手机号码不正确',

// 触发器可以同时用blur和change，二者之间用英文逗号隔开

trigger: ['change','blur'],

}

],

code: [

{

required: true,

message: '请输入验证码',

trigger: ['change','blur'],

},

{

type: 'number',

message: '验证码只能为数字',

trigger: ['change','blur'],

}

],

password: [

{

required: true,

message: '请输入密码',

trigger: ['change','blur'],

},

{

// 正则不能含有两边的引号

pattern: /^(?![0-9]+$)(?![a-zA-Z]+$)[0-9A-Za-z]+\S{5,12}$/,

message: '需同时含有字母和数字，长度在6-12之间',

trigger: ['change','blur'],

}

],

rePassword: [

{

required: true,

message: '请重新输入密码',

trigger: ['change','blur'],

},

{

validator: (rule, value, callback) => {

return value === this.model.password;

},

message: '两次输入的密码不相等',

trigger: ['change','blur'],

}

],

},

border: false,

check: false,

selectStatus: 'close',

checkboxList: [

{

name: '唱歌',

checked: false,

disabled: false

},

{

name: '跳舞',

checked: false,

disabled: false

},

{

name: 'rap',

checked: false,

disabled: false

},

{

name: '篮球',

checked: false,

disabled: false

}

],

radioList: [

{

name: '是',

checked: true,

disabled: false

},

{

name: '否',

checked: false,

disabled: false

},

{

name: '不确定',

checked: false,

disabled: false

}

],

radio: '是',

actionSheetList: [

{

text: '男'

},

{

text: '女'

},

{

text: '保密'

}

],

actionSheetShow: false,

pickerShow: false,

pickerShow\_addr: false,

selectShow: false,

radioCheckWidth: 'auto',

radioCheckWrap: false,

labelPosition: 'left',

codeTips: '',

errorType: ['message'],

};

},

onLoad() {

},

computed: {

borderCurrent() {

return this.border ? 0 : 1;

}

},

onReady() {

this.$refs.uForm.setRules(this.rules);

},

methods: {

submit() {

this.$refs.uForm.validate(valid => {

if (valid) {

if(!this.model.agreement) return this.$u.toast('请勾选协议');

console.log('验证通过');

} else {

console.log('验证失败');

}

});

console.log(this.$data)

},

// 点击actionSheet回调

actionSheetCallback(index) {

uni.hideKeyboard();

this.model.sex = this.actionSheetList[index].text;

},

// checkbox选择发生变化

checkboxGroupChange(e) {

this.model.interest = e;

},

// radio选择发生变化

radioGroupChange(e) {

this.model.is\_reg = e;

},

// 勾选版权协议

checkboxChange(e) {

this.model.agreement = e.value;

},

// 选择地区回调

regionConfirm\_household\_regist(e) {

console.log("addrgist",e)

this.model.household\_regist = e.province.label + '-' + e.city.label + '-' + e.area.label;

},

regionConfirm\_address(e) {

console.log("addr",e)

this.model.address = e.province.label + '-' + e.city.label + '-' + e.area.label;

},

// 选择商品类型回调

selectConfirm(e) {

this.model.goodsType = '';

e.map((val, index) => {

this.model.goodsType += this.model.goodsType == '' ? val.label : '-' + val.label;

})

},

borderChange(index) {

this.border = !index;

},

radioCheckboxChange(index) {

if(index == 0) {

this.radioCheckWrap = false;

this.radioCheckWidth = 'auto';

} else if(index == 1) {

this.radioCheckWrap = true;

this.radioCheckWidth = 'auto';

} else if(index == 2) {

this.radioCheckWrap = false;

this.radioCheckWidth = '50%';

}

},

labelPositionChange(index) {

this.labelPosition = index == 0 ? 'left' : 'top';

},

codeChange(text) {

this.codeTips = text;

},

// 获取验证码

getCode() {

if(this.$refs.uCode.canGetCode) {

// 模拟向后端请求验证码

uni.showLoading({

title: '正在获取验证码',

mask: true

})

setTimeout(() => {

uni.hideLoading();

// 这里此提示会被this.start()方法中的提示覆盖

this.$u.toast('验证码已发送');

// 通知验证码组件内部开始倒计时

this.$refs.uCode.start();

}, 2000);

} else {

this.$u.toast('倒计时结束后再发送');

}

},

errorChange(index) {

if(index == 0) this.errorType = ['message'];

if(index == 1) this.errorType = ['toast'];

if(index == 2) this.errorType = ['border-bottom'];

if(index == 3) this.errorType = ['border'];

}

}

};

</script>

<style scoped lang="scss">

.wrap {

padding: 30rpx;

}

.agreement {

display: flex;

align-items: center;

margin: 40rpx 0;

.agreement-text {

padding-left: 8rpx;

color: $u-tips-color;

}

}

</style>