一、 编写 node 路由，连接后台服务

1. 连接代理

```
const TAF = require('@taf/taf-rpc').Communicator.New();
const Test = require('./UserProxy').Test;

let servant = 'Test.UserZjServer.UserZjObj';
if (!process.env.TAF_CONFIG) {
    servant += '@tcp -h 127.0.0.1 -p 14001 -t 60000';
}
const proxy = TAF.stringToProxy(Test.UserProxy, servant);
exports.Test = Test;
exports.proxy = proxy;
```

2. 编写路由

   1) 登录

```
router.post('/loginForAd', (req, res) => {
    const { username, password } = req.body;
    const stReq = new Test.LoginReq();
    stReq.readFromObject({
        username,
        password,
    });
    proxy.loginForAd(stReq).then(ret => {
        jsonWrite(res, ret.response.arguments.stRsp.toObject());
    }, ret => {
        console.log('error');
    });
});
```

   2) 添加用户

```
// 增加用户接口
router.post('/saveUser', (req, res) => {
    const { username, password, tel, gender, age } = req.body;
    const stReq = new Test.SaveUserInfoReq();
    stReq.readFromObject({
        userinfo: {
            username,
            password,
            gender,
            tel,
            age
        }
    });
    proxy.saveUser(stReq).then(ret => {
        jsonWrite(res, ret.response.arguments.stRsp.toObject());
    }, ret => {
        console.log('error');
    });
});
```

3) 获取用户列表

```
// 获取用户列表
router.get('/getUserListForAd', (req, res) => {
    const { pageNum, pageSize } = req.query;
    const stReq = new Test.PageInfoReq();
    stReq.readFromObject({
        pageNum,
        pageSize
    });
    proxy.getUserListForAd(stReq).then(ret => {
        jsonWrite(res, ret.response.arguments.stRsp.toObject());
    }, ret => {
        console.log('error');
    });
});
```

4) 删除用户

```
// 删除用户
router.get('/delUser', (req, res) => {
    const { uid } = req.query;
    const stReq = new Test.QueryIdReq();
    stReq.readFromObject({
        uid
    });
    proxy.delUser(stReq).then(ret => {
        jsonWrite(res, ret.response.arguments.stRsp.toObject());
    }, ret => {
        console.log('error');
    });
});
```

5) 修改用户

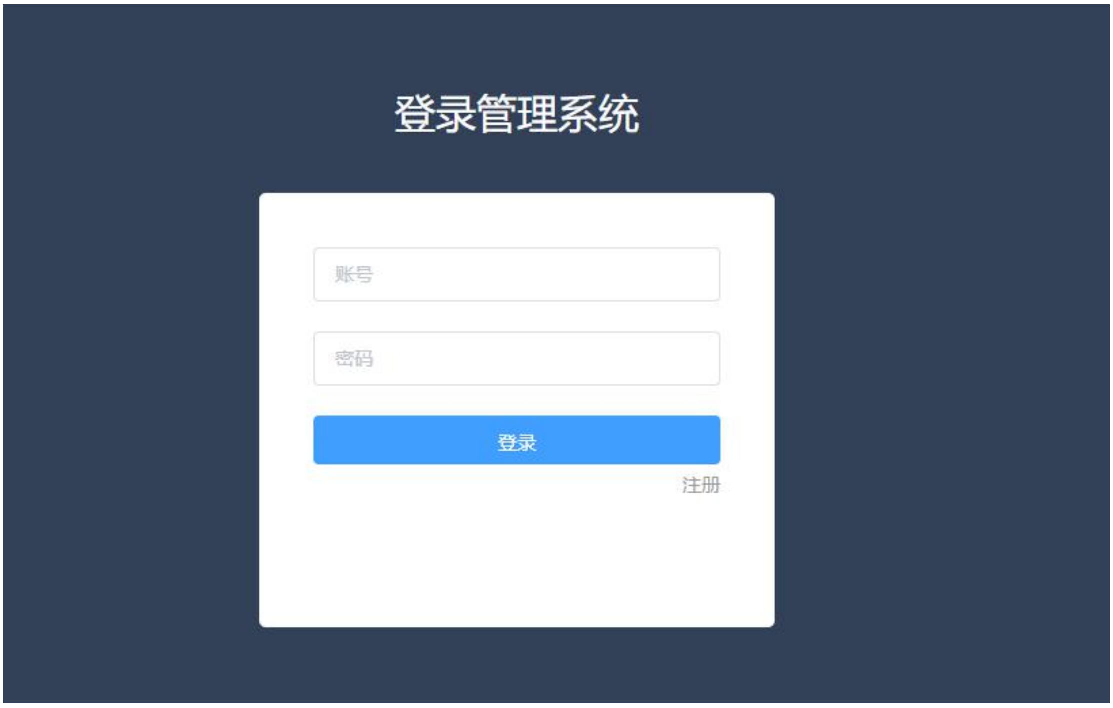
```
// 修改用户
router.post('/updateUser', (req, res) => {
    const { uid, username, password, tel, gender, age } = req.body;
    const stReq = new Test.SaveUserInfoReq();
    stReq.readFromObject({
        userinfo: {
            uid,
            username,
            password,
            gender,
            tel,
            age
        }
    });
    proxy.saveUser(stReq).then(ret => {
        jsonWrite(res, ret.response.arguments.stRsp.toObject());
    }, ret => {
        console.log('error');
    });
});
```

6) 用户详细信息

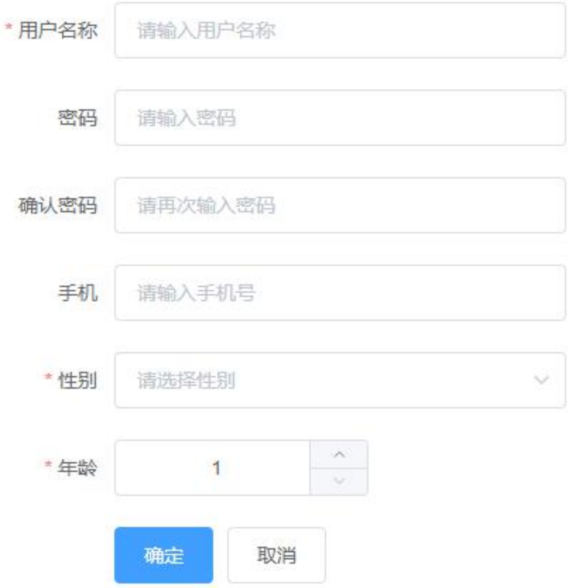
```
// 用户详细
router.get('/getUserDetailForAd', (req, res) => {
    const { uid } = req.query;
    const stReq = new Test.QueryIdReq();
    stReq.readFromObject({
        uid
    });
    proxy.getUserDetailForAd(stReq).then(ret => {
        jsonWrite(res, ret.response.arguments.stRsp.toObject());
    }, ret => {
        console.log('error');
    });

});
```

二、 设计界面，编写页面逻辑请求数据

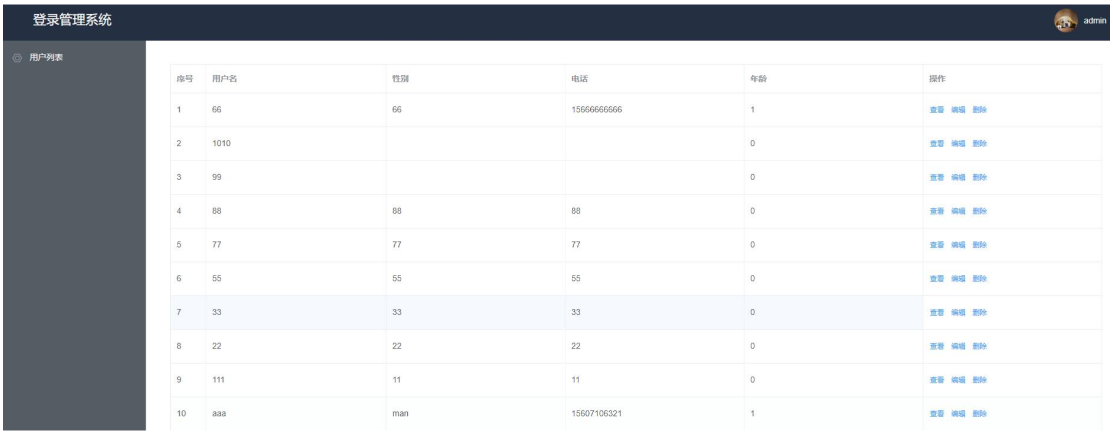
1. 登录


登录管理系统
账号
密码
登录
注册

2. 注册

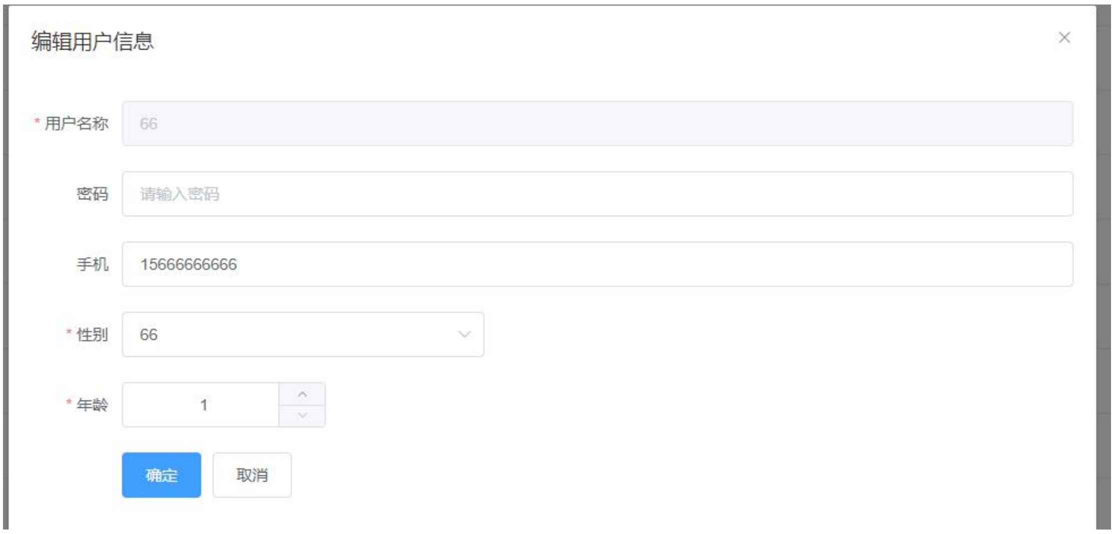

* 用户名称
请输入用户名称
密码
请输入密码
确认密码
请再次输入密码
手机
请输入手机号
* 性别
请选择性别
* 年龄
1
确定
取消

3. 用户列表



序号	用户名	性别	电话	年龄	操作
1	66	66	15666666666	1	查看 编辑 删除
2	1010			0	查看 编辑 删除
3	99			0	查看 编辑 删除
4	88	88	88	0	查看 编辑 删除
5	77	77	77	0	查看 编辑 删除
6	55	55	55	0	查看 编辑 删除
7	33	33	33	0	查看 编辑 删除
8	22	22	22	0	查看 编辑 删除
9	111	11	11	0	查看 编辑 删除
10	aaa	man	15607106321	1	查看 编辑 删除

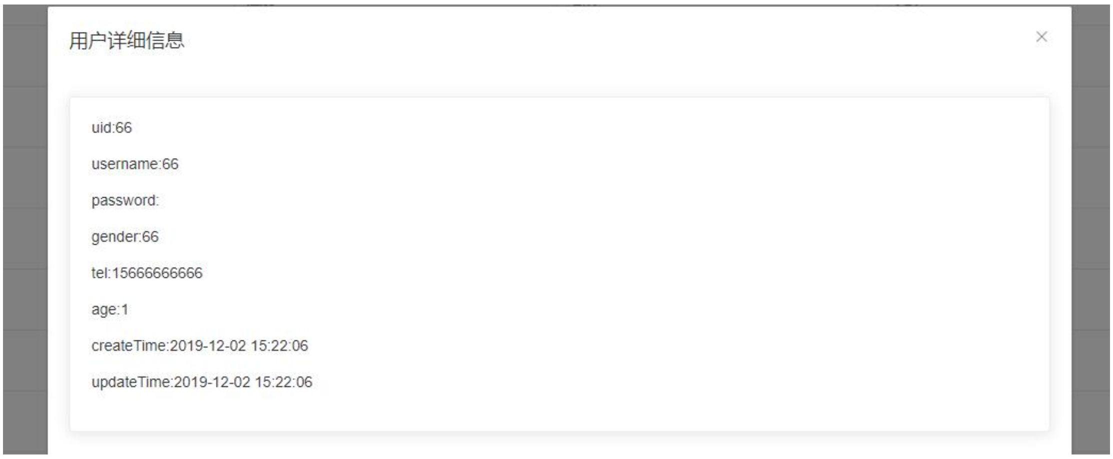
4. 编辑



编辑用户信息

* 用户名称	66
密码	请输入密码
手机	15666666666
* 性别	66
* 年龄	1

5. 详情



用户详细信息

uid:66

username:66

password:

gender:66

tel:15666666666

age:1

createTime:2019-12-02 15:22:06

updateTime:2019-12-02 15:22:06

三、 明日计划

熟悉 dcache api,完成 dcache 操作