

# 前端开发需求说明

---

## 前端开发需求说明

须知

网站主页

登录页

下载中心页

上传中心页

用户个人页面

用户消息页面

---

## 须知

所有静态文件的路径均已"/staticFile"为开头，由后端解析实际地址更换为此路径使用。

前端调用后端返回数据解释：

若后端给前端返回一个Msg对象，含有a、b两个成员，若要使用对应成员的值只需要`{{ .a }}`、`{{ .b }}`即可。后面所有的后端返回数据均用这种方法调用。

有合适的方案可以对前后端交互接口重新设计。

---

## 网站主页

`[GET] handle: "/indexPage"`

需求：实现主站的首页展示效果，可以在此处添加一些样式效果等，最主要的功能是实现一个注册页面的post请求，将用户输入的数据提交到后台。

需要提供的注册信息有：

- 姓名
- 密码
- 年龄
- 学号
- 邮箱
- 宿舍号

其中姓名密码以及学号需要非空验证。

密码是6-18位字符。

邮箱需要正则判断格式，年龄也需要验证。

后端接收的接口：`[POST] action="/sign"`

提交完成后后端会对数据进行验证，如有不合法学号/邮箱会给予一个反馈，调用方法如下：

后端会返回一个对象，该对象可能包含若干个成员。定义如下：

`alert.stuNoErr = "学号已被注册过"`

```
alert.stuEmailErr = "邮箱已经被注册过"
```

根据返回的数据给与用户一定的反馈。

---

## 登录页

```
[GET] handle : "/loginPage" 或 handle : "/"
```

在开发时可以将本页面同注册页合并设计。

需求：要求提供用户登陆的页面，主要是设计一个post请求将登陆数据发送到后端。

需要提供的内容有：

- 学号
- 密码

需要对数据进行验证

后端接收的接口：

```
[POST] action="/login"
```

同样会对提交的数据进行检查，返回一个对象，定义如下：

```
alert.loginErr = 用户名或密码错误
```

根据返回的数据给予用户一定的反馈

### 顶部信息栏设计

对于已经登陆过的用户会在后续的页面中在每个页面上添加一个导航栏的功能，下面就其功能做描述：

首页：返回首页的按钮

登录：回到登录页的按钮

上传中心：跳转到上传中心页面

下载资源：跳转到下载中心

我的消息：跳转到用户消息页面

头像：最右侧有当前用户的头像，点击即可跳转回用户个人页面。后端会对每个用户返回唯一的地址，调用接口：

```
src = "root/" + {{ .userHeadImageDir }} + "headImage.jpg"
```

每个页面的跳转地址与具体的GET请求地址一致

**请在每一个页面顶部设计该导航栏，并在生成页面时渲染生成头像**

---

## 下载中心页

```
[GET] handle : "/downloadCenter"
```

需求：提供用户一个下载中心的导航页面。

[第一版]需要有的内容有：

- 一个地址可以转到用户总上传的文件页面
- 一个地址可以转到服务器总下载页面

每个地址需要提供一个a标签或者其他页面转移功能，地址内容由后端提供。

后端返回一个对象，定义如下：

```
msg.usersUploadFiles = /xxxxx
```

```
msg.hostFiles = /xxxxxx
```

根据后端返回的对象成员构造本页面功能。

---

## 上传中心页

```
[GET] handle : uploadPage
```

需求：提供给用户上传本地文件的功能。

需要有的内容：

- input 标签 读取本地文件准备上传
- submit 按钮，提供上传

后端接收的接口：

```
[POST] action = "/usersUpload"
```

后端返回的对象，定义如下：

```
msg.ifm = "成功"/"上传失败"
```

根据返回的内容来反馈给用户。

---

## 用户个人页面

```
[GET] handle : /userSpace
```

需求：本页面是用户直接查看和管辖自己信息的页面，也是用户个人页面簇的第一个页面。

需要有的内容：

- 公告板，用于显示当前服务器公示信息等，调用内容见接口。
- 设置头像的表单，一个上传文件的form表单。
- 个人信息完成度，标识当前用户信息的完整程度。

后端接收的接口（头像）：

```
[POST] action = "/userSpace/message/headImage"
```

在第一次载入时或重设头像后会返回一个对象，定义如下：

```
msg.finish = 一个小数，需要对其表示为百分数
```

```
msg.board = 公告信息
```

```
msg.headImageIfm = 设置成功/失败
```

关于页面的排版布局/样式不做要求，可随意发挥。

---

## 用户消息页面

```
[GET] handle : userSpace/message
```

需求：本页面是用户个人页面下的子页面，用于发送消息/查看最近接收的消息。

需要的内容有：

- 显示当前用户名

- 一个发送器，用于对指定的用户发送消息，需要有接收者、消息内容这两个主要部分，通过form表达提交。
- 一个消息列表，用于显示发送给当前用户的所有消息，大体如下：

发送者	已读	发送时间	内容
张三	未读	2020-6-20	张三你好帅

大致设计如此，如若内容部分盛不下可以用别的方法设计（点击弹窗/跳转页面显示等方法），这里不做限制，只要实现了功能，美观即可。

后端接收发送消息的接口：[POST] `action = "/userSpace/message/sendMSG"`

后端回传的数据接口：

`Ifm.msg` = 一个JSON数组，存有所有的消息数据，对JSON对象定义如下：

```
{"Sender":"","Receiver":"","Content":"","SendTime":"","READ":""}
```

其中receiver虽然传过来了但是可以根据需要决定。其他部分内容均需要显示

对于这个JSON数组，需要转换为对象数组并在列表当中显示。

`Ifm.user` = 用户名