

## 0. 资源地址

---

1. 线上 DEMO 项目地址: <http://www.escook.cn:8086/>
2. 项目的 API 接口地址: [https://www.showdoc.cc/escook?page\\_id=3707158761215217](https://www.showdoc.cc/escook?page_id=3707158761215217)

## 1. 项目前期的准备工作

---

### 1.1 初始化项目结构

1. 将 `素材` 目录下的 `assets` 和 `home` 文件夹, 拷贝到 `code` 目录下
2. 在 `code` 目录下新建 `login.html` 和 `index.html` 页面

### 1.2 使用GitHub管理大事件的项目

1. 在 `code` 目录中运行 `git init` 命令
2. 在 `code` 目录中运行 `git add .` 命令
3. 在 `code` 目录下运行 `git commit -m "init project"` 命令
4. 新建 Github 仓库 `web_bigevent`
5. 将本地仓库和Github仓库建立关联关系
6. 将本地仓库的代码推送到Github仓库中
7. 运行 `git checkout -b login` 命令, 创建并切换到 `login` 分支

### 1.3 安装VSCode的Live Server插件辅助开发

1. 在插件市场, 搜索 `Live Server` 并安装
2. 在页面上鼠标右键, 选择 `Open With Live Server` 即可快速使用 http 协议访问页面

## 2. 登录注册

---

### 2.1 绘制login页面的基本结构

1. 编写 HTML 结构:

```
1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <meta charset="UTF-8" />
5      <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6    </head>
7    <title>大事件-登录/注册</title>
8    <!-- 导入 LayUI 的样式 -->
9    <link rel="stylesheet" href="/assets/lib/layui/css/layui.css" />
10   <!-- 导入自己的样式表 -->
```

```

10     <link rel="stylesheet" href="/assets/css/login.css" />
11 </head>
12 <body>
13     <!-- 头部的 Logo 区域 -->
14     <div class="layui-main">
15         
16     </div>
17
18     <!-- 登录注册区域 -->
19     <div class="loginAndRegBox">
20         <div class="title-box"></div>
21     </div>
22 </body>
23 </html>
24

```

## 2. 美化样式:

```

1  html,
2  body {
3      margin: 0;
4      padding: 0;
5      height: 100%;
6      width: 100%;
7      background: url('/assets/images/login_bg.jpg') no-repeat center;
8      background-size: cover;
9  }
10
11  .loginAndRegBox {
12      width: 400px;
13      height: 310px;
14      background-color: #fff;
15      position: absolute;
16      left: 50%;
17      top: 50%;
18      transform: translate(-50%, -50%);
19  }
20
21  .title-box {
22      height: 60px;
23      background: url('/assets/images/login_title.png') no-repeat center;
24  }

```

## 2.2 实现登录和注册的按需切换

### 1. 编写 HTML 结构:

```

1     <!-- 登录注册区域 -->
2     <div class="loginAndRegBox">
3         <div class="title-box"></div>
4         <!-- 登录的div -->
5         <div class="login-box"></div>
6         <!-- 注册的div -->
7         <div class="reg-box"></div>
8     </div>

```

### 2. 编写样式:

```

1  .reg-box {
2      display: none;
3  }

```

### 3. 编写 JavaScript 代码:

```

1  $(function() {
2      // 点击“去注册账号”的链接
3      $('#link_reg').on('click', function() {
4          $('.login-box').hide()
5          $('.reg-box').show()
6      })
7
8      // 点击“去登录”的链接
9      $('#link_login').on('click', function() {
10         $('.login-box').show()
11         $('.reg-box').hide()
12     })
13 })

```

## 2.3 绘制登录表单的基本结构

### 1. 编写 HTML 结构:

```

1      <!-- 登录的div -->
2      <div class="login-box">
3          <!-- 登录的表单 -->
4          <form class="layui-form" action="">
5              <!-- 用户名 -->
6              <div class="layui-form-item">
7                  <i class="layui-icon layui-icon-username"></i>
8                  <input type="text" name="username" required lay-verify="required"
placeholder="请输入用户名" autocomplete="off" class="layui-input" />
9              </div>
10             <!-- 密码 -->
11             <div class="layui-form-item">
12                 <i class="layui-icon layui-icon-password"></i>
13                 <input type="password" name="password" required lay-verify="required" placeholder="请输入密码" autocomplete="off" class="layui-input" />
14             </div>
15             <!-- 登录按钮 -->
16             <div class="layui-form-item">
17                 <!-- 注意: 表单提交按钮和普通按钮的区别, 就是 lay-submit 属性 -->
18                 <button class="layui-btn layui-btn-fluid layui-btn-normal" lay-submit>登录</button>
19             </div>
20             <div class="layui-form-item links">
21                 <a href="javascript:;" id="link_reg">去注册账号</a>
22             </div>
23         </form>
24     </div>

```

## 2.4 美化登录表单的样式

### 1. 编写样式:

```

1  .layui-form {
2      padding: 0 30px;
3  }
4
5  .links {
6      display: flex;
7      justify-content: flex-end;
8  }
9
10 .links a {
11     font-size: 12px;
12 }

```

## 2.5 绘制文本框前面的小图标

1. 在用户名的文本框之前，添加如下的标签结构：

```
1  <i class="layui-icon layui-icon-username"></i>
```

2. 在密码框之前，添加如下的标签结构：

```
1  <i class="layui-icon layui-icon-password"></i>
```

3. 美化样式：

```

1  .layui-form-item {
2      position: relative;
3  }
4
5  .layui-icon {
6      position: absolute;
7      left: 10px;
8      top: 10px;
9  }
10
11 .layui-input {
12     padding-left: 32px;
13 }

```

## 2.6 快速绘制注册的表单

1. 将登录的表单复制一份，并修改为注册的表单即可：

```

1      <!-- 注册的div -->
2      <div class="reg-box">
3          <!-- 注册的表单 -->
4          <form class="layui-form" action="">
5              <!-- 用户名 -->
6              <div class="layui-form-item">
7                  <i class="layui-icon layui-icon-username"></i>
8                  <input type="text" name="username" required lay-verify="required"
placeholder="请输入用户名" autocomplete="off" class="layui-input" />
9              </div>
10             <!-- 密码 -->
11             <div class="layui-form-item">
12                 <i class="layui-icon layui-icon-password"></i>

```

```

13         <input type="password" name="password" required lay-
verify="required" placeholder="请输入密码" autocomplete="off" class="layui-
input" />
14     </div>
15     <!-- 密码确认框 -->
16     <div class="layui-form-item">
17         <i class="layui-icon layui-icon-password"></i>
18         <input type="password" name="repassword" required lay-
verify="required" placeholder="再次确认密码" autocomplete="off"
class="layui-input" />
19     </div>
20     <!-- 注册按钮 -->
21     <div class="layui-form-item">
22         <!-- 注意：表单提交按钮和普通按钮的区别，就是 lay-submit 属性 -->
23         <button class="layui-btn layui-btn-fluid layui-btn-normal" lay-
submit>注册</button>
24     </div>
25     <div class="layui-form-item links">
26         <a href="javascript:;" id="link_login">去登录</a>
27     </div>
28 </form>
29 </div>

```

## 2.7 实现登录表单的验证

1. 导入 layui 的 js 文件：

```
1 <script src="/assets/lib/layui/layui.all.js"></script>
```

2. 为需要验证的表单项添加 `lay-verify` 属性，同时指定具体的校验规则即可。

## 2.8 自定义校验规则

1. 从 layui 中获取 form 对象：

```
1 var form = layui.form
```

2. 通过 `form.verify()` 函数自定义校验规则：

```

1 form.verify({
2     // 自定义了一个叫做 pwd 校验规则
3     pwd: [/^\s{6,12}$/, '密码必须6到12位，且不能出现空格'],
4     // 校验两次密码是否一致的规则
5     repwd: function(value) {
6         // 通过形参拿到的是确认密码框中的内容
7         // 还需要拿到密码框中的内容
8         // 然后进行一次等于的判断
9         // 如果判断失败，则return一个提示消息即可
10        var pwd = $('<code>.reg-box [name=password]</code>').val()
11        if (pwd !== value) {
12            return '两次密码不一致！'
13        }
14    }
15 })

```

3. 按需为表单项添加校验规则：

```

1      <input type="password" name="repassword" required lay-
verify="required|pwd|repwd" placeholder="再次确认密码" autocomplete="off"
class="layui-input" />

```

## 2.9 发起注册用户的Ajax请求

1. 为注册表单添加id:

```

1      <!-- 注册的表单 -->
2      <form class="layui-form" id="form_reg"></form>

```

2. 监听提交事件:

```

1      // 监听注册表单的提交事件
2      $('#form_reg').on('submit', function(e) {
3          // 1. 阻止默认的提交行为
4          e.preventDefault()
5          // 2. 发起Ajax的POST请求
6          var data = {
7              username: $('#form_reg [name=username]').val(),
8              password: $('#form_reg [name=password]').val()
9          }
10         $.post('http://ajax.frontend.itheima.net/api/reguser', data,
function(res) {
11             if (res.status !== 0) {
12                 return layer.msg(res.message)
13             }
14             layer.msg('注册成功, 请登录! ')
15             // 模拟人的点击行为
16             $('#link_login').click()
17         })
18     })

```

## 2.10 使用layer提示消息

1. 导入 layer:

```

1      var layer = layui.layer

```

2. 调用 `layer.msg()` 提示消息:

```

1      layer.msg('注册成功, 请登录! ')

```

## 2.11 发起登录的Ajax请求

1. 为登录表单添加id:

```

1      <form class="layui-form" id="form_login"></form>

```

2. 监听提交事件:

```

1      // 监听登录表单的提交事件
2      $('#form_login').submit(function(e) {
3          // 阻止默认提交行为
4          e.preventDefault()
5          $.ajax({
6              url: '/api/login',

```

```

7         method: 'POST',
8         // 快速获取表单中的数据
9         data: $(this).serialize(),
10        success: function(res) {
11            if (res.status !== 0) {
12                return layer.msg('登录失败!')
13            }
14            layer.msg('登录成功!')
15            // 将登录成功得到的 token 字符串, 保存到 localStorage 中
16            localStorage.setItem('token', res.token)
17            // 跳转到后台主页
18            location.href = '/index.html'
19        }
20    })
21 })

```

## 2.12 在ajaxPrefilter中统一拼接请求的根路径

1. 在 `/assets/js` 目录中新建 `baseAPI.js`
2. 编写如下代码:

```

1    // 注意: 每次调用 $.get() 或 $.post() 或 $.ajax() 的时候,
2    // 会先调用 ajaxPrefilter 这个函数
3    // 在这个函数中, 可以拿到我们给Ajax提供的配置对象
4    $.ajaxPrefilter(function(options) {
5        // 在发起真正的 Ajax 请求之前, 统一拼接请求的根路径
6        options.url = 'http://ajax.frontend.itheima.net' + options.url
7    })
8

```

## 2.13 提交login分支的代码到GitHub

1. 运行 `git add .` 命令
2. 运行 `git commit -m "完成了登录和注册的功能"` 命令
3. 运行 `git push -u origin login` 命令
4. 运行 `git checkout master` 命令
5. 运行 `git merge login` 命令
6. 运行 `git push` 命令
7. 运行 `git checkout -b index` 命令

# 3. 后台主页

## 3.1 快速实现后台主页的布局效果

1. 从 layUI 官方文档中粘贴布局的主要代码, 并修改如下:

```

1    <!DOCTYPE html>
2    <html lang="en">
3        <head>
4            <meta charset="UTF-8" />

```

```

5      <meta name="viewport" content="width=device-width, initial-scale=1.0"
    />
6      <title>后台主页</title>
7      <!-- 导入 layui 的样式表 -->
8      <link rel="stylesheet" href="/assets/lib/layui/css/layui.css" />
9  </head>
10 <body class="layui-layout-body">
11   <div class="layui-layout layui-layout-admin">
12     <div class="layui-header">
13       <div class="layui-logo">
14         
15       </div>
16       <!-- 头部区域（可配合layui已有的水平导航） -->
17       <ul class="layui-nav layui-layout-right">
18         <li class="layui-nav-item">
19           <a href="javascript:;">
20             
21             个人中心
22           </a>
23           <dl class="layui-nav-child">
24             <dd><a href="">基本资料</a></dd>
25             <dd><a href="">更换头像</a></dd>
26             <dd><a href="">重置密码</a></dd>
27           </dl>
28         </li>
29         <li class="layui-nav-item"><a href="">退出</a></li>
30       </ul>
31     </div>
32
33     <div class="layui-side layui-bg-black">
34       <div class="layui-side-scroll">
35         <!-- 左侧导航区域（可配合layui已有的垂直导航） -->
36         <ul class="layui-nav layui-nav-tree" lay-filter="test">
37           <li class="layui-nav-item layui-nav-itemed">
38             <a class="" href="javascript:;">所有商品</a>
39             <dl class="layui-nav-child">
40               <dd><a href="javascript:;">列表一</a></dd>
41               <dd><a href="javascript:;">列表二</a></dd>
42               <dd><a href="javascript:;">列表三</a></dd>
43               <dd><a href="">超链接</a></dd>
44             </dl>
45           </li>
46           <li class="layui-nav-item">
47             <a href="javascript:;">解决方案</a>
48             <dl class="layui-nav-child">
49               <dd><a href="javascript:;">列表一</a></dd>
50               <dd><a href="javascript:;">列表二</a></dd>
51               <dd><a href="">超链接</a></dd>
52             </dl>
53           </li>
54           <li class="layui-nav-item"><a href="">云市场</a></li>
55           <li class="layui-nav-item"><a href="">发布商品</a></li>
56         </ul>
57       </div>
58     </div>
59
60     <div class="layui-body">
61       <!-- 内容主体区域 -->

```



```

62         <div style="padding: 15px;">内容主体区域</div>
63     </div>
64
65     <div class="layui-footer">
66         <!-- 底部固定区域 -->
67         © layui.com - 底部固定区域
68     </div>
69 </div>
70 <!-- 导入 layui 的JS文件 -->
71 <script src="/assets/lib/layui/layui.all.js"></script>
72 </body>
73 </html>

```

## 3.2 修改侧边栏的结构

```

1  <div class="layui-side layui-bg-black">
2      <div class="layui-side-scroll">
3          <!-- 左侧导航区域（可配合layui已有的垂直导航） -->
4          <ul class="layui-nav layui-nav-tree">
5              <li class="layui-nav-item">
6                  <a href="/">首页</a>
7              </li>
8              <li class="layui-nav-item">
9                  <a class="" href="javascript:;">文章管理</a>
10                 <dl class="layui-nav-child">
11                     <dd>
12                         <a href="javascript:;">文章类别</a>
13                     </dd>
14                     <dd>
15                         <a href="javascript:;">文章列表</a>
16                     </dd>
17                     <dd>
18                         <a href="javascript:;">发布文章</a>
19                     </dd>
20                 </dl>
21             </li>
22             <li class="layui-nav-item">
23                 <a href="javascript:;">个人中心</a>
24                 <dl class="layui-nav-child">
25                     <dd>
26                         <a href="javascript:;">基本资料</a>
27                     </dd>
28                     <dd>
29                         <a href="javascript:;">更换头像</a>
30                     </dd>
31                     <dd>
32                         <a href="javascript:;">重置密码</a>
33                     </dd>
34                 </dl>
35             </li>
36         </ul>
37     </div>
38 </div>

```

## 3.3 使用lay-shrink实现左侧菜单互斥效果

```

1      <div class="layui-side layui-bg-black">
2          <div class="layui-side-scroll">
3              <!-- 左侧导航区域（可配合layui已有的垂直导航） -->
4              <ul class="layui-nav layui-nav-tree" lay-shrink="all">
5                  <!-- 省略其他代码 -->
6              </ul>
7          </div>
8      </div>

```

### 3.4 为菜单项添加图标

1. 导入第三方的图标库：

```

1      <!-- 导入第三方图标库 -->
2      <link rel="stylesheet" href="/assets/fonts/iconfont.css" />

```

2. 修改左侧菜单的结构：

```

1      <div class="layui-side layui-bg-black">
2          <div class="layui-side-scroll">
3              <!-- 左侧导航区域（可配合layui已有的垂直导航） -->
4              <ul class="layui-nav layui-nav-tree" lay-shrink="all">
5                  <li class="layui-nav-item">
6                      <a href=""><span class="iconfont icon-home"></span>首页</a>
7                  </li>
8                  <li class="layui-nav-item">
9                      <a class="" href="javascript:;"><span class="iconfont icon-16">
</span>文章管理</a>
10                     <dl class="layui-nav-child">
11                         <dd>
12                             <a href="javascript:;"><i class="layui-icon layui-icon-
app"></i>文章类别</a>
13                         </dd>
14                         <dd>
15                             <a href="javascript:;"><i class="layui-icon layui-icon-
app"></i>文章列表</a>
16                         </dd>
17                         <dd>
18                             <a href="javascript:;"><i class="layui-icon layui-icon-
app"></i>发布文章</a>
19                         </dd>
20                     </dl>
21                 </li>
22                 <li class="layui-nav-item">
23                     <a href="javascript:;"><span class="iconfont icon-user"></span>
个人中心</a>
24                     <dl class="layui-nav-child">
25                         <dd>
26                             <a href="javascript:;"><i class="layui-icon layui-icon-
app"></i>基本资料</a>
27                         </dd>
28                         <dd>
29                             <a href="javascript:;"><i class="layui-icon layui-icon-
app"></i>更换头像</a>
30                         </dd>
31                         <dd>

```

```

32         <a href="javascript:;"><i class="layui-icon layui-icon-
    app"></i>重置密码</a>
33     </dd>
34 </dl>
35 </li>
36 </ul>
37 </div>
38 </div>

```

3. 修改头部“退出”按钮的结构：

```

1     <li class="layui-nav-item">
2         <a href=""><span class="iconfont icon-tuichu"></span>退出</a>
3     </li>

```

4. 导入自己的样式表文件：

```

1     <link rel="stylesheet" href="/assets/css/index.css" />

```

5. 编写自己的样式：

```

1     .layui-footer {
2         text-align: center;
3         font-size: 12px;
4     }
5
6     .iconfont {
7         margin-right: 8px;
8     }
9
10    .layui-icon {
11        margin-right: 8px;
12        margin-left: 20px;
13    }

```

### 3.5 使用iframe标签在内容主体区域显示网页内容

1. 在页面主体的 div 中添加 `iframe`：

```

1     <div class="layui-body">
2         <!-- 内容主体区域 -->
3         <iframe name="fm" src="" frameborder="0"></iframe>
4     </div>

```

2. 为 `首页` 链接添加 `href` 和 `target` 属性：

```

1     <a href="/home/dashboard.html" target="fm"><span class="iconfont icon-
    home"></span>首页</a>

```

3. 美化样式：

```

1  iframe {
2      width: 100%;
3      height: 100%;
4  }
5
6  .layui-body {
7      overflow: hidden;
8  }

```

### 3.6 解决3个小问题

1. 为 `iframe` 指定默认页面:

```

1  <iframe name="fm" src="/home/dashboard.html" frameborder="0"></iframe>

```

2. 为 `首页` 对应的导航 Item 项添加 `layui-this` 属性:

```

1  <li class="layui-nav-item layui-this">
2      <a href="/home/dashboard.html" target="fm"><span class="iconfont icon-home"></span>首页</a>
3  </li>

```

3. 强制清除 `<a>` 链接的 CSS3 动画:

```

1  a {
2      transition: none !important;
3  }

```

### 3.7 渲染图片头像和文字头像

1. 修改头部区域的的头像结构如下:

```

1  <a href="javascript:;" class="userinfo">
2      
3      <span class="text-avatar">A</span>
4      个人中心
5  </a>

```

2. 在左侧导航区域的 `ul` 之前添加如下头像结构:

```

1  <div class="userinfo">
2      
3      <span class="text-avatar">A</span>
4      <span id="welcome">欢迎 ***</span>
5  </div>

```

3. 添加样式美化 UI 结构:

```

1  .userinfo {
2      height: 60px;
3      line-height: 60px;
4      text-align: center;
5      font-size: 12px;
6      user-select: none;
7  }
8
9  .layui-side-scroll .userinfo {

```

```
10     border-bottom: 1px solid #282b33;
11 }
12
13 .layui-nav-img {
14     width: 40px;
15     height: 40px;
16 }
17
18 .text-avatar {
19     display: inline-block;
20     width: 40px;
21     height: 40px;
22     background-color: #009688;
23     border-radius: 50%;
24     line-height: 40px;
25     text-align: center;
26     font-size: 20px;
27     color: #fff;
28     position: relative;
29     top: 4px;
30     margin-right: 10px;
31 }
```