**整体介绍：**  
  
这个网站建设目的是让大家分享好的ai图片生成咒语

具体的功能有 首页瀑布流无限加载展示

详情页有点赞和评论功能

开放注册入口所有人都可以上传图片

登录后可以对自己上传的图片进行编辑，并且删除自己图片的评论

可以通过参数控制评论开关，可以指定超级管理员账号，维护所有图片。

**页面讲解：**

http://localhost/pubu/pubu.php

它通过引入Bootstrap和自定义的CSS样式表来设置页面的外观和响应式设计。

导航栏和初始内容：页面顶部有一个导航栏，包括Logo、注册和登录按钮。初始内容区域可能是空的，用于后续动态加载瀑布流内容。

JavaScript脚本：该文件包含JavaScript代码，负责处理瀑布流的动态加载。它监听滚动事件，当用户滚动到页面底部时，触发数据加载的功能。

JavaScript部分整体介绍

在pubu.php文件中，JavaScript代码主要负责实现瀑布流布局的动态加载功能。核心功能包括：

**初始化变量**：设置初始页面和分类变量，用于跟踪用户在浏览内容时的位置和选择。

**发送AJAX请求**：使用jQuery的AJAX方法向服务器请求数据。请求的URL包括当前页码和分类信息。

**处理返回数据**：接收并处理从script.php返回的数据。数据通常以JSON格式返回，包含需要展示的图片和相关信息。

**动态内容加载**：将返回的数据动态地添加到页面中，创建新的HTML元素以展示图片和相关信息。

**无限滚动逻辑**：实现当用户滚动到页面底部时自动加载下一页内容的功能。

以下是该JavaScript代码段的逐行解释：

**Javascript Copy code**

let currentPage = 0; // 定义一个变量 currentPage 来跟踪当前请求的页面

初始化currentPage变量为0，用于记录当前加载到的页面数。

**Javascript Copy code**

let currentCategory = <?php echo $category = isset($\_GET['category']) ? $\_GET['category'] : 0; ?> // 新增变量存储当前分类

currentCategory变量用于存储当前选择的分类。这个值是从PHP代码中获取的，表示当前URL参数中的分类。

**Javascript Copy code**

const container = $('#container'); // 获取页面中 id 为 container 的元素

获取页面中ID为container的元素，该元素将用于放置动态加载的内容。

**Javascript Copy code**

const showMoreItems = function () {

$.ajax({ // 使用 jQuery 的 ajax 方法来发送请求

url: `**script.php**?page=${currentPage}&category=${currentCategory}`, // 请求的 url，其中包含当前页面和分类

method: "GET", // 请求的方法为 GET

dataType: "json", // 返回的数据类型为 json

定义函数showMoreItems，用于发送AJAX请求到script.php。请求包含当前页码和分类信息，预期返回的数据类型是JSON。

**Javascript Copy code**

success: function (data) { // 请求成功时，执行的回调函数

data.forEach(item => { // 遍历返回的数据

当请求成功时，执行回调函数。函数内部遍历返回的数据，每个item代表一条图片信息。

**Javascript Copy code**

let newElement = $( // 利用 jQuery 创建一个新的 HTML 元素

`<div class="item">

<h3>${item.spell}</h3>

<a href="detail-page.php?infoid=${item.infoID}">

<img src="${item.imageAddr}" alt="Image of ${item.spell}">

</a>

</div>`

);

container.append(newElement); // 将新元素添加到容器中

});

currentPage++; // 更新当前页面变量

}

});

};

对于每个项目，创建一个新的HTML元素，其中包含图片和标题，并将其添加到container中。

完成数据添加后，currentPage自增，为下一次加载做准备。

这段代码实现了一个典型的瀑布流动态加载功能，使用户能够在滚动页面时不断加载和查看新内容。这种方式提升了用户体验，同时对服务器的压力进行了有效分散。

**接下来打开上面提到的script.php页面做讲解**

script.php页面的主要作用是从后端数据库中获取并返回给前端需要展示的数据，以便在pubu.php页面上实现瀑布流布局的动态加载。下面是对script.php的整体功能和流程的概述：

**初始化和数据库连接**：script.php首先包含必要的函数文件（如functions.php），这通常包含数据库连接设置和其他通用函数。然后，它建立与数据库的连接，准备执行查询。

**参数获取**：script.php接收来自前端的参数，主要是分类(category)。这些参数用于确定要从数据库中检索哪部分数据。

**构建和执行SQL查询**：根据接收到的参数，构建一个SQL查询，以从数据库中检索相应的图片和文字信息。这通常涉及到连接多个表，并应用特定的过滤条件（如检查图片是否被删除，图片的类型等）。

**数据处理和格式化**：从数据库检索到的数据被处理和格式化，通常转换为JSON格式，这是因为JSON易于在JavaScript中处理。

**数据返回**：处理后的数据通过HTTP响应发送回前端。pubu.php中的JavaScript将使用这些数据来动态地更新页面，展示新加载的图片和相关信息。

这个文件是实现瀑布流布局动态加载功能的关键，因为它提供了前端所需的数据。通过这种方式，网站能够在用户浏览时持续展示新内容，而无需重新加载整个页面。

**接下来我们打开内页detail-page.php做讲解：**

detail-page.php 页面在您的项目中扮演着展示特定图片及其详细信息的角色。这个页面通常用于在用户点击某个图片时展示更多的详情。以下是对detail-page.php的整体功能和流程的概述：

页面结构和样式：detail-page.php 包含HTML代码，用于定义页面布局和样式。这可能包括头部、图片展示区域、评论区域等。页面的样式可能通过引入CSS文件来定义。

图片和信息展示：页面的核心功能是展示选中图片的详细信息。这通常包括图片本身、相关的“咒语”（即图片生成时的关键词或描述）、以及可能的额外信息，如图片类型、上传者等。

动态内容加载：如果页面设计为动态加载图片信息（例如，根据传递的图片ID从数据库中检索信息），则可能会包含PHP代码来执行数据库查询，并将查询结果嵌入到HTML中。

用户互动：页面可能包含用户互动元素，如点赞按钮、评论区域。这些功能可以通过JavaScript和后端PHP脚本实现，以便在用户进行操作时更新数据库（如记录点赞、存储评论）。

响应式设计：与pubu.php类似，detail-page.php也可能采用响应式设计，以确保在不同大小的屏幕上均能良好显示。

导航和链接：页面可能包含返回主页或其他部分的导航链接，以便用户在查看完详情后继续浏览其他内容。

综上所述，detail-page.php是一个用于展示单个图片及其详细信息的页面，它不仅提供了丰富的内容展示，还可能包含与用户互动的功能。这个页面是提升用户体验的关键部分，因为它允许用户深入了解他们感兴趣的图片。

代码中17行-57行是获取图片和点赞数量的代码，为了确保安全性，使用了PHP的预处理语句（Prepared Statements）来从数据库中安全地检索数据，18行 创建了一个预处理语句。预处理语句是一种特殊的SQL语句，它被发送到数据库并编译，之后可以多次执行。在这个语句中，? 是一个占位符，代表稍后将绑定的参数。19行的 bind\_param 函数用于绑定实际的参数值到预处理语句的占位符上。在这里，"i" 表示绑定的变量（$infoid）是一个整数类型。这种类型指定提高了安全性，因为它确保了传递给数据库的数据类型是预期的类型。

208行开始 页面加载完成后执行以下代码 展示评论 以及 获得最新点赞量

fetchLatestComments是获取最新评论的函数，submitComment是提交评论的函数

414行附近的 allowComments 是 参数里面的控制是否评论

接下来讲一下注册regedit.html页面

就是用了insert\_user函数插入信息到数据库，代码在 functions.php 文件里面 所有页面共同调用

接下来是登录页面

重点就是账户密码正确后写入session用于登录判断

session\_start();

$\_SESSION['username'] =$username;

$\_SESSION['UserID'] =$row['UserID'];

现在讲解下application.php页面

现在登录系统打开admin-guanli.php页面

所有后台页面会if (isset($\_SESSION['username'])) {判断是否登录

后台管理页面使用get方式获取canshu参数的值来实现内容页面的变化

105-135部分循环输出图片列表，增加编辑按钮

canshu=sczy展示上传图片页面

图片上传函数是在upload\_process.php里面，27行定义了会根据当天日期创建文件夹，16行也会判断文件类型是否符合标准，最后成功了跳转回后台列表页面

其他管理评论编辑图片等就是基本的数据库操作，可简单介绍下，结束。