

# **Lab05**

## **Tổng hợp nội dung**

### **1.Lab02**

#### **- Use Case Description**

##Use Case Description: Quản trị CMS

 Tên Use Case: Quản trị CMS

-Mục đích: Giúp Admin có thể quản lý toàn bộ nội dung và các khía cạnh liên quan trên hệ thống, đảm bảo nội dung được kiểm soát chặt chẽ trước khi xuất bản.

-Tác nhân chính: Admin

-Điều kiện tiên quyết: Admin đã đăng nhập vào hệ thống với quyền quản trị.

-Luồng sự kiện chính:

+Admin đăng nhập vào hệ thống.

+Hệ thống hiển thị giao diện quản trị CMS.

-Admin thực hiện các chức năng sau:

-Quản lý bài viết, hình ảnh, video: Admin có thể tạo, chỉnh sửa, xóa các bài viết, hình ảnh, và video.

-Duyệt bài trước khi xuất bản: Admin xem xét và phê duyệt các nội dung do Content Manager tạo trước khi chúng được hiển thị công khai.

-Quản lý bình luận: Admin xem và quản lý các bình luận từ người dùng, có thể xóa hoặc ẩn các bình luận không phù hợp.

-Phân quyền người dùng: Admin tạo và gán quyền cho các tài khoản người dùng khác nhau (ví dụ: gán vai trò Content Manager).

+Admin đăng xuất khỏi hệ thống.

+Luồng sự kiện thay thế:

+Nếu Admin không phê duyệt bài viết: Bài viết sẽ ở trạng thái chờ duyệt và không được hiển thị công khai.

## ##Use Case Description: Xuất bản đa kênh

⌚ Tên Use Case: Xuất bản đa kênh

-Mục đích: Giúp Content Manager có thể dễ dàng đăng tải và quản lý nội dung trên nhiều nền tảng khác nhau từ một giao diện duy nhất.

-Tác nhân chính: Content Manager

-Điều kiện tiên quyết: Content Manager đã đăng nhập vào hệ thống.

-Luồng sự kiện chính:

+Content Manager đăng nhập vào hệ thống.

+Hệ thống hiển thị giao diện quản lý nội dung.

+Content Manager tạo một nội dung mới (bài viết, hình ảnh, video).

+Content Manager chọn các nền tảng muốn xuất bản (ví dụ: Facebook, Zalo, TikTok, YouTube).

+Content Manager có thể lập lịch xuất bản nội dung vào một thời điểm cụ thể.

+Hệ thống tự động đăng nội dung lên các nền tảng đã chọn theo lịch trình.

+Nếu là video, Content Manager có thể nhúng video từ YouTube/TikTok vào website.

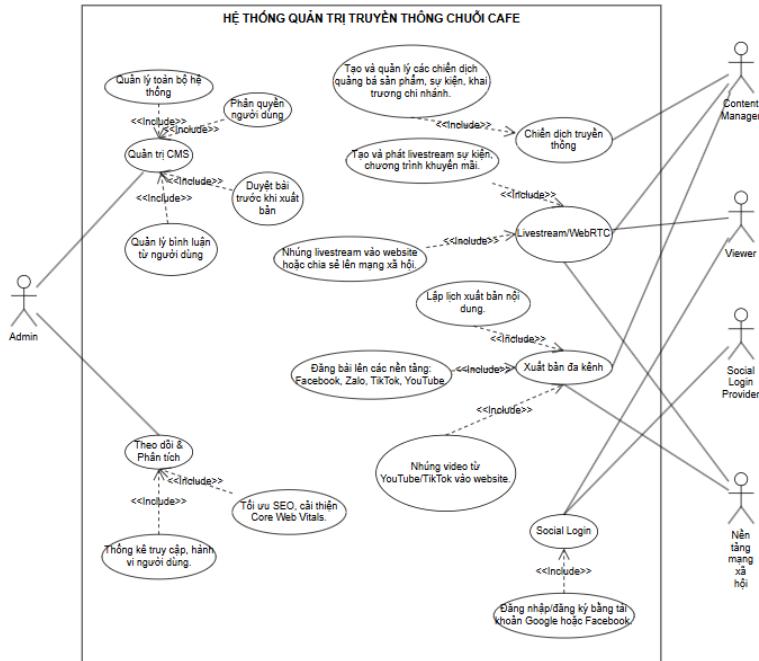
+Content Manager đăng xuất khỏi hệ thống.

+Luồng sự kiện thay thế:

+Nếu kết nối với nền tảng mạng xã hội bị lỗi: Hệ thống sẽ thông báo lỗi và Content Manager có thể thử lại hoặc xử lý thủ công.

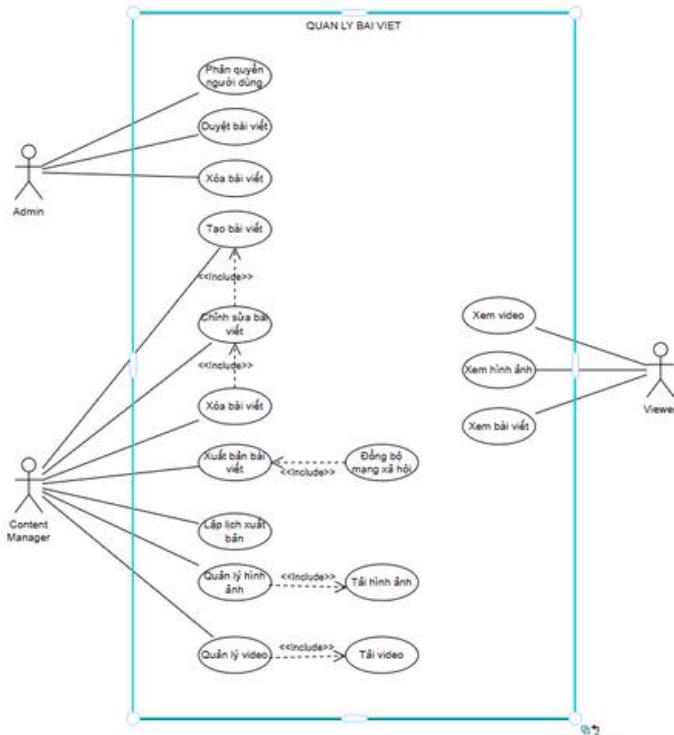
+Nếu Content Manager không lập lịch: Nội dung sẽ được xuất bản ngay lập tức sau khi hoàn thành.

- Sơ đồ Use Case Diagram

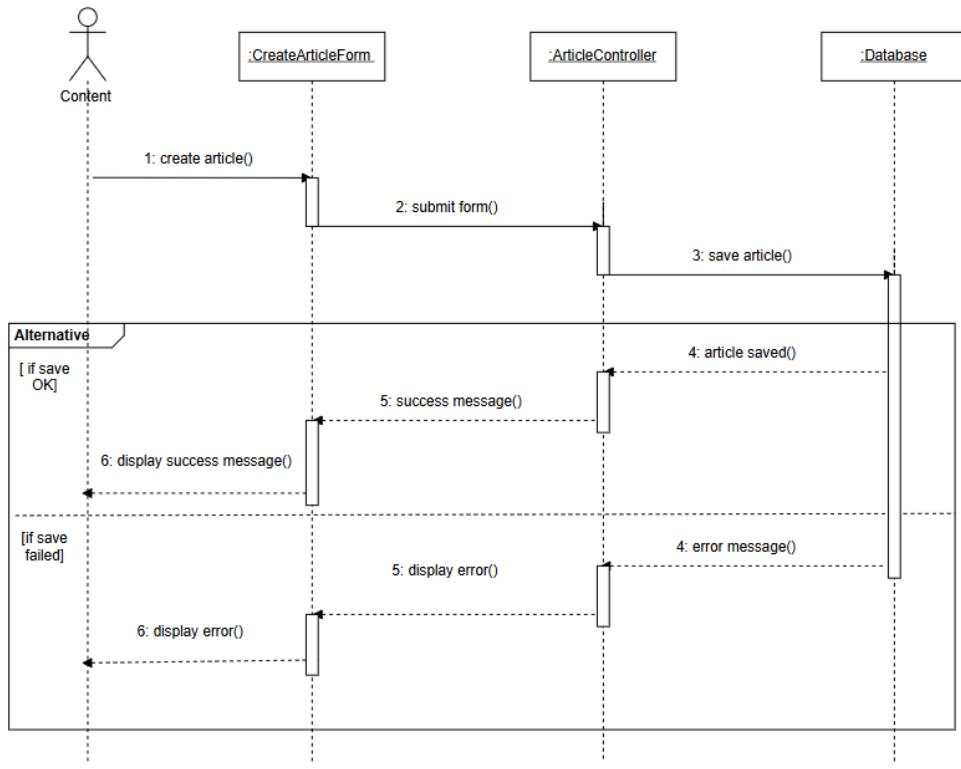


## 2.Lab03

### - Sơ đồ Use Case Diagram



## - Sơ đồ Sequence Diagram



## - Giải thích:

# 📄 Use Case Descriptions

## 1. Quản lý Bài viết

### Use Case: Tạo bài viết

\* \*\*Mục đích\*\*: Cho phép Content Manager tạo mới một bài viết.

\* \*\*Tác nhân chính\*\*: Content Manager

\* \*\*Điều kiện tiên quyết\*\*: Content Manager đã đăng nhập vào hệ thống.

\* \*\*Luồng sự kiện chính\*\*:

1. Content Manager chọn chức năng “Tạo bài viết”.
2. Nhập tiêu đề, nội dung, gắn thẻ, thêm hình ảnh hoặc video minh họa.
3. Lưu bài viết dưới dạng “nháp” hoặc gửi phê duyệt.

\* \*\*Luồng thay thế\*\*:

\* Nếu dữ liệu nhập không hợp lệ (thiếu tiêu đề, nội dung trống) → Hệ thống báo lỗi.

\* \*\*Điều kiện kết thúc\*\*: Bài viết được lưu thành công và sẵn sàng để chỉnh sửa/xuất bản.

### ### Use Case: Chỉnh sửa bài viết

\* \*\*Mục đích\*\*: Cho phép Content Manager chỉnh sửa bài viết đã tạo.

\* \*\*Tác nhân chính\*\*: Content Manager

\* \*\*Precondition\*\*: Bài viết tồn tại trong hệ thống.

\* \*\*Main Flow\*\*:

1. Content Manager mở danh sách bài viết.
2. Chọn một bài viết → sửa tiêu đề, nội dung, hình ảnh.
3. Lưu lại bản chỉnh sửa.

\* \*\*Alternative Flow\*\*:

\* Nếu không có quyền chỉnh sửa → Hệ thống từ chối.

\* \*\*Postcondition\*\*: Bài viết được cập nhật.

### ### Use Case: Xóa bài viết

\* \*\*Mục đích\*\*: Xóa bài viết không còn cần thiết.

\* \*\*Tác nhân chính\*\*: Content Manager hoặc Admin

\* \*\*Precondition\*\*: Bài viết tồn tại.

\* \*\*Main Flow\*\*:

1. Actor chọn “Xóa bài viết”.
2. Hệ thống hiển thị cảnh báo.
3. Actor xác nhận → Bài viết bị xóa.

\* \*\*Alternative Flow\*\*:

\* Nếu Actor hủy thao tác → Bài viết vẫn giữ nguyên.

\* \*\*Postcondition\*\*: Bài viết bị xóa khỏi hệ thống.

### ### Use Case: Lập lịch xuất bản

\* \*\*Mục đích\*\*: Đặt thời gian tự động xuất bản bài viết.

\* \*\*Tác nhân chính\*\*: Content Manager

\* \*\*Precondition\*\*: Bài viết đã tồn tại.

\* \*\*Main Flow\*\*:

1. Content Manager chọn “Lập lịch xuất bản”.

2. Nhập ngày giờ xuất bản.

3. Hệ thống lưu lịch hẹn.

\* \*\*Alternative Flow\*\*:

- \* Nếu thời gian không hợp lệ (quá khứ) → Thông báo lỗi.

\* \*\*Postcondition\*\*: Lịch xuất bản được thiết lập.

### ### Use Case: Xuất bản bài viết

\* \*\*Mục đích\*\*: Đưa bài viết ra công khai cho Viewer xem.

\* \*\*Tác nhân chính\*\*: Content Manager

\* \*\*Precondition\*\*: Bài viết đã được duyệt.

\* \*\*Main Flow\*\*:

1. Chọn chức năng “Xuất bản”.

2. Hệ thống kiểm tra duyệt.

3. Bài viết được đăng công khai.

4. <<include>> Đồng bộ lên mạng xã hội.

\* \*\*Alternative Flow\*\*:

- \* Nếu chưa được Admin duyệt → Bài viết không xuất bản.

\* \*\*Postcondition\*\*: Bài viết công khai trên hệ thống (và mạng xã hội).

### ### Use Case: Đồng bộ mạng xã hội (Include)

\* \*\*Mục đích\*\*: Tự động đăng bài viết sang các nền tảng MXH.

\* \*\*Tác nhân chính\*\*: Hệ thống (include khi xuất bản).

\* \*\*Precondition\*\*: Hệ thống kết nối với MXH thành công.

\* \*\*Main Flow\*\*:

1. Sau khi xuất bản, hệ thống gửi dữ liệu sang Facebook, Zalo, TikTok...

2. Bài viết hiển thị trên các nền tảng.

\* \*\*Alternative Flow\*\*:

\* Nếu kết nối lỗi → Hệ thống thông báo, yêu cầu Content Manager xử lý thủ công.

\* \*\*Postcondition\*\*: Bài viết được đồng bộ đa kênh.

## 2. Quản lý Hình ảnh

### Use Case: Tải hình ảnh lên

\* \*\*Mục đích\*\*: Cho phép Content Manager upload hình ảnh.

\* \*\*Tác nhân chính\*\*: Content Manager

\* \*\*Precondition\*\*: Đã đăng nhập.

\* \*\*Main Flow\*\*:

1. Chọn chức năng “Tải hình ảnh”.

2. Chọn file ảnh từ thiết bị.

3. Nhập metadata (tiêu đề, mô tả).

4. Lưu vào hệ thống.

\* \*\*Alternative Flow\*\*:

\* Nếu file quá lớn hoặc định dạng sai → Báo lỗi.

\* \*\*Postcondition\*\*: Hình ảnh được lưu trữ

### Use Case: Quản lý hình ảnh

\* \*\*Mục đích\*\*: Cho phép Content Manager xem, chỉnh sửa, đổi tên, xóa ảnh.

\* \*\*Tác nhân chính\*\*: Content Manager

\* \*\*Precondition\*\*: Có ảnh trong hệ thống.

\* \*\*Main Flow\*\*:

1. Mở “ThU viện hình ảnh”.
2. Thực hiện thao tác chỉnh sửa (đổi tên, sửa metadata, xóa).
3. Lưu thay đổi.

\* \*\*Alternative Flow\*\*:

\* Nếu ảnh bị khóa (đang dùng trong bài viết) → Không cho xóa.

\* \*\*Postcondition\*\*: Hình ảnh được cập nhật hoặc xóa.

### ## 3. Quản lý Video

### Use Case: Tải video lên

\* \*\*Mục đích\*\*: Upload video vào hệ thống.

\* \*\*Tác nhân chính\*\*: Content Manager

\* \*\*Precondition\*\*: Đã đăng nhập.

\* \*\*Main Flow\*\*:

1. Chọn “Tải video”.
2. Upload file từ thiết bị.
3. Nhập thông tin mô tả.
4. Lưu lại.

\* \*\*Alternative Flow\*\*:

\* Nếu dung lượng vượt giới hạn → Báo lỗi.

\* \*\*Postcondition\*\*: Video lưu trữ thành công.

### ### Use Case: Quản lý video

\* \*\*Mục đích\*\*: Quản lý video đã upload (xem, đổi tên, chỉnh sửa metadata, xóa).

\* \*\*Tác nhân chính\*\*: Content Manager

\* \*\*Precondition\*\*: Video tồn tại.

\* \*\*Main Flow\*\*:

1. Chọn “Quản lý video”.
2. Thực hiện thao tác mong muốn.
3. Lưu thay đổi.

\* \*\*Alternative Flow\*\*:

\* Nếu video đang gắn vào bài viết → Không cho xóa.

\* \*\*Postcondition\*\*: Video được cập nhật hoặc xóa.

#### ## 4. Viewer (Người dùng cuối)

### Use Case: Xem bài viết / Xem hình ảnh / Xem video

\* \*\*Mục đích\*\*: Viewer có thể xem nội dung công khai.

\* \*\*Tác nhân chính\*\*: Viewer

\* \*\*Precondition\*\*: Nội dung đã được xuất bản.

\* \*\*Main Flow\*\*:

1. Viewer truy cập website.
2. Chọn danh mục (bài viết / ảnh / video).
3. Xem nội dung chi tiết.

\* \*\*Alternative Flow\*\*:

\* Nếu nội dung đã bị gỡ bỏ → Hệ thống báo lỗi.

\* \*\*Postcondition\*\*: Viewer xem được nội dung công khai.

#### ## Giải thích Sequence Diagram: Tạo bài viết##

👉 Liên hệ với Use Case “Tạo bài viết”:

Sequence Diagram minh họa chi tiết luồng tương tác giữa các đối tượng (Content Manager, giao diện nhập bài, bộ điều khiển, và cơ sở dữ liệu) trong quá trình tạo và lưu một bài viết.

## \*\*Các bước tương tác trong sơ đồ\*\*

\*Content Manager bắt đầu hành động bằng việc chọn chức năng create article() trên hệ thống.\*

→ Đây tương ứng với bước “Content Manager chọn chức năng Tạo bài viết” trong UC.

\*CreateArticleForm (giao diện nhập liệu) xuất hiện và cho phép Content Manager điền thông tin. Sau đó, form gửi dữ liệu bằng hành động submit form() đến ArticleController.\*

→ Liên hệ với UC: nhập tiêu đề, nội dung, hình ảnh/video rồi nhấn nút Submit. ArticleController nhận dữ liệu, thực hiện gọi hàm save article() để lưu xuống Database.

\*Tại Database, có hai khả năng xảy ra:\*

1. Trường hợp thành công: Database phản hồi article saved().
2. Trường hợp thất bại: Database trả về error message().

Nếu lưu thành công, ArticleController gửi lại success message() về CreateArticleForm.

Giao diện hiển thị thông báo “Bài viết đã được lưu thành công”.

Đây chính là luồng Main flow trong UC.

Nếu lưu thất bại, ArticleController gửi lại display error() cho CreateArticleForm.

Giao diện hiển thị thông báo lỗi.

Đây chính là luồng Alternative flow trong UC (lưu thất bại → người dùng được thông báo lỗi).

### 3. Lab04

#### \*HTML

```
<!DOCTYPE html>
<html lang="vi">
<head>
    <meta charset="UTF-8">
    <title>Công truyền thông số - Chuỗi Cafe T.G </title>
    <link rel="stylesheet" href="style.css">
</head>
<body>
    <!-- Form Login -->
    <div id="login-container" class="container">
        <form id="loginForm" class="login-box">
            <!-- Logo -->
            <div class="login-container">
                <div class="logo-wrapper">
                    
                </div>
                <form class="loginForm">
                    <h2>Đăng nhập hệ thống</h2>
                    <label for="username">Tên đăng nhập</label>
                    <input type="text" id="username" >
```

```
<label for="password">Mật khẩu</label>
<input type="password" id="password" >

<div class="remember">
  <input type="checkbox" id="rememberMe">
  <label for="rememberMe">Ghi nhớ đăng nhập</label>
</div>

<div class="buttons">
  <button type="submit">Login</button>
  <button type="button" id="cancelBtn">Cancel</button>
</div>
</form>
</div>
</form>
</div>

<!-- Dashboard -->
<div id="dashboard" class="hidden">
  <aside class="sidebar">
    <!-- Avatar admin -->
    <div class="admin-profile">
      
      <div class="admin-info">
        <p class="admin-name">  Cafe T.G</p>
```

```
<a href="#" class="admin-settings">Quản lý tài khoản</a>
</div>
</div>

<ul>
    <li><a href="#" onclick="showSection('posts')">Quản lý Bài
viết</a></li>
    <li><a href="#" onclick="showSection('menu')">Quản lý Menu</a></li>
    <li><a href="#" onclick="showSection('customers')">Quản lý Khách
hàng</a></li>
    <li><a href="#" onclick="showSection('stats')">Báo cáo & Thông
kê</a></li>
</ul>
<button onclick="logout()">Đăng xuất</button>
</aside>

<main class="content">
    <h2 id="section-title">Chào mừng bạn đến với hệ thống</h2>
    <div id="section-content">
        <p>Vui lòng chọn một chức năng từ menu bên trái để quản trị.</p>
    </div>
</main>
</div>

<script src="script.js"></script>
</body>
```

```
</html>
```

### \*Script.js

```
document.getElementById("loginForm").addEventListener("submit",  
function(e){  
    e.preventDefault();  
  
    const user = document.getElementById("username").value;  
    const pass = document.getElementById("password").value;  
    const remember = document.getElementById("rememberMe").checked;  
  
    if(user === "" || pass === ""){  
        alert("Vui lòng nhập đầy đủ Username và Password!");  
        return;  
    }  
  
    if(pass.length < 8){  
        alert("Mật khẩu phải có ít nhất 8 ký tự!");  
        return;  
    }  
  
    if(remember){  
        localStorage.setItem("savedUser", user);  
    } else {  
        localStorage.removeItem("savedUser");  
    }  
  
    // Chuyển sang dashboard  
    document.getElementById("login-container").classList.add("hidden");
```

```
document.getElementById("dashboard").classList.remove("hidden");
document.getElementById("section-title").innerText = "Xin chào, " + user;
});

// Nút Cancel
document.getElementById("cancelBtn").addEventListener("click", function(){
    document.getElementById("username").value = "";
    document.getElementById("password").value = "";
});

// Hiện nội dung Dashboard theo menu
function showSection(section){
    const title = document.getElementById("section-title");
    const content = document.getElementById("section-content");

    switch(section){
        case "posts":
            title.innerText = "Quản lý Bài viết";
            content.innerHTML = "<p>Danh sách bài viết quảng bá cafe sẽ hiển thị ở đây...</p>";
            break;
        case "menu":
            title.innerText = "Quản lý Menu";
            content.innerHTML = "<p>Danh sách sản phẩm đồ uống sẽ hiển thị ở đây...</p>";
            break;
    }
}
```

```
case "customers":  
    title.innerText = "Quản lý Khách hàng";  
    content.innerHTML = "<p>Thông tin khách hàng thân thiết sẽ hiển thị ở  
đây...</p>";  
    break;  
  
case "stats":  
    title.innerText = "Thống kê";  
    content.innerHTML = "<p>Biểu đồ doanh thu, lượt check-in sẽ hiển thị ở  
đây...</p>";  
    break;  
}  
}  
  
// Logout  
function logout(){  
    document.getElementById("dashboard").classList.add("hidden");  
    document.getElementById("login-container").classList.remove("hidden");  
}  
  
const users = [  
    { username: "admin1", role: "Admin Tổng" },  
    { username: "manager1", role: "Quản lý Chi nhánh" },  
    { username: "editor1", role: "Biên tập nội dung" },  
    { username: "campaign1", role: "Quản lý Chiến dịch" }  
];  
  
// Hiển thị quyền sau khi đăng nhập
```

```
function showRole(username) {  
  const user = users.find(u => u.username === username);  
  if (user) {  
    document.querySelector('.admin-name').textContent = user.role;  
  }  
}  
  
let activityLog = [];  
  
function logActivity(username, action) {  
  const time = new Date().toLocaleString();  
  activityLog.push({ username, action, time });  
  console.log(`#${time} - ${username}: ${action}`);  
}  
  
logActivity("admin1", "Đăng nhập hệ thống");  
logActivity("editor1", "Chỉnh sửa bài viết #123");  
  
*style.css
```

```
* {  
  margin: 0; padding: 0;  
  box-sizing: border-box;  
  font-family: sans-serif ;  
}  
  
.admin-name {  
  font-size: 20px;  
  font-weight: bold;
```

```
margin: 5px 0;}
```

/\* Login Form \*/

```
/* Styling cho container tổng của trang đăng nhập */
.login-container {
    width: 350px;
    margin: 50px auto;
    padding: 20px;
    background-color: #fff;
    border-radius: 8px;
    box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
    text-align: center;
}

/* Styling cho vùng chứa logo */
.logo-wrapper {
    margin-bottom: 20px;
    text-align: center;
}

.logo-wrapper img {
    max-width: 150px;
    height: auto;
```

```
    display: block;  
    margin: 0 auto;  
}  
  
  
  
  

```

```
.login-form h2 {  
    margin-bottom: 20px;  
    color: #333;  
    text-align: center;  
}
```

```
.input-group {  
    margin-bottom: 15px;  
    text-align: left;  
}
```

```
.input-group label {  
    display: block;  
    margin-bottom: 5px;  
    font-weight: bold;  
    color: #555;  
    text-align: left;  
}
```

```
.input-group input {
```

```
width: calc(100% - 20px);  
padding: 10px;  
border: 1px solid #ddd;  
border-radius: 4px;  
}
```

```
.login-form button {  
width: 100%;  
padding: 10px;  
background-color: #007bff;  
color: white;  
border: none;  
border-radius: 4px;  
cursor: pointer;  
font-size: 16px;  
margin-top: 10px;  
}
```

```
.login-form button:hover {  
background-color: #0056b3;  
}
```

```
.container {  
display: flex;  
justify-content: center;  
align-items: center;
```

```
color:#4E1F00;  
width: 100%;  
min-height: 100vh;  
  
background: url("https://i.postimg.cc/rwJGd6VP/cc.jpg") no-repeat center  
center;  
background-size: cover;  
}  
  
/* Smartphone */  
@media (max-width: 600px) {  
.login-box {  
width: 90%;  
padding: 15px;  
}  
}  
  
/* Tablet */  
@media (min-width: 601px) and (max-width: 1024px) {  
.login-box {  
width: 70%;  
}  
}  
  
/* Laptop & Desktop */  
@media (min-width: 1025px) {
```

```
.login-box {  
    width: 300px;  
}  
}  
  
.login-box {  
    width: clamp(250px, 80%, 400px);  
    padding: 2rem;  
    background: #F8F4E1;  
    border-radius: 10px;  
    box-shadow: 0 5px 15px rgba(0,0,0,0.3);  
}  
  
.login-box label {  
    display: block;  
    margin: 10px 0 5px;  
    text-align: left;  
}  
  
.login-box input[type="text"],  
.login-box input[type="password"] {  
    width: 100%;  
    padding: 8px;  
}
```

```
border: 1px solid #aaa;  
border-radius: 5px;  
}  
.remember {  
margin: 10px 0;  
text-align: left;  
}  
.buttons {  
display: flex;  
justify-content: space-between;  
}  
  
.buttons button {  
padding: 8px 15px;  
border: none;  
border-radius: 5px;  
cursor: pointer;  
}  
  
.buttons button:first-child {  
background: #4caf50;  
color: white;  
}  
  
.buttons button:last-child {
```

```
background: #f44336;
color: white;
}

/* Dashboard */

.hidden { display: none !important; }

#dashboard {
display: none;
height: 100vh;
}

#dashboard:not(.hidden) {
display: flex;
}

.sidebar {
width: 270px;
background: #4E1F00;
color: white;
padding: 20px;
}

.sidebar h3 {
text-align: center;
margin: 20px 0;
```

}

```
.sidebar ul {  
    list-style: none;  
    margin-bottom: 20px;  
}  
  
{
```

```
.sidebar ul li {  
    margin: 15px 0;  
    padding-bottom: 10px;  
    border-bottom: 1px solid rgba(255,255,255,0.2);}  
  
{
```

```
.sidebar ul li:last-child {  
    border-bottom: none;  
}  
  
{
```

```
.sidebar ul li a {  
    color: #F8F4E1;  
    text-decoration: none;  
}  
  
{
```

```
.sidebar ul li a:hover {  
    color: white;  
}  
  
{
```

```
.sidebar button {  
    width: 100%;  
    padding: 10px;  
    background: #f44336;  
    border: none;  
    border-radius: 5px;  
    color: white;  
    cursor: pointer;  
}
```

```
.content {  
    flex: 1;  
    padding: 30px;  
    background: #F8F4E1;  
}
```

```
.content h2 {  
    margin-bottom: 15px;  
}
```

```
/* Admin Profile */  
.admin-profile {  
    display: flex;  
    align-items: center;  
    margin-bottom: 20px;
```

```
padding: 10px;  
background: rgba(255, 255, 255, 0.05);  
border-radius: 10px;  
}  
  
}
```

```
.admin-avatar {  
width: 50px;  
height: 50px;  
border-radius: 50%;  
margin-right: 10px;  
border: 2px solid #ffccbc;  
object-fit: cover;  
}
```

```
.admin-info .admin-name {  
font-weight: bold;  
color: #fff;  
}  
.admin-info .admin-settings {  
font-size: 0.85rem;  
color: #ffccbc;  
text-decoration: underline;  
cursor: pointer;  
}
```

```
.admin-info .admin-settings:hover {  
    color: #ffffff;  
}  
}
```

