CricLive - Full HTML, CSS & JavaScript Code

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>CricLive - Live Cricket Scores & Updates</title>
    <style>
        * {
            margin: 0;
            padding: 0;
            box-sizing: border-box;
            font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
        }
        body {
            background-color: #f0f2f5;
            color: #333;
            line-height: 1.6;
        }
        header {
            background: linear-gradient(135deg, #0c622e, #1a863a);
            color: white;
            padding: 1rem;
            text-align: center;
            box-shadow: 0 4px 6px rgba(0,0,0,0.1);
        }
        header h1 {
            font-size: 2rem;
            margin-bottom: 0.5rem;
        }
        nav {
            display: flex;
            justify-content: center;
            background: #14532d;
            padding: 0.5rem;
        }
        nav a {
            color: white;
            text-decoration: none;
            margin: 0 15px;
            font-weight: bold;
            transition: color 0.3s ease;
        }
        nav a:hover {
            color: #facc15;
        }
        .container {
            max-width: 1200px;
            margin: 20px auto;
            padding: 20px;
        }
        .card {
            background: white;
            border-radius: 10px;
            padding: 20px;
            margin: 15px 0;
            box-shadow: 0 4px 12px rgba(0,0,0,0.1);
            transition: transform 0.3s ease;
        }
        .card:hover {
            transform: translateY(-5px);
        }
        h2 {
            color: #0c622e;
            margin-bottom: 15px;
        }
        .match {
            padding: 15px;
            border-bottom: 1px solid #eee;
        }
```

```
.match:last-child {
    border-bottom: none;
}
.teams {
    display: flex;
    justify-content: space-between;
    align-items: center;
}
.score {
    font-size: 1.2rem;
    font-weight: bold;
    color: #1a863a;
}
.status {
    margin-top: 8px;
    font-size: 0.9rem;
    color: #666;
}
.news-item {
    padding: 10px 0;
    border-bottom: 1px solid #eee;
}
.news-item:last-child {
    border-bottom: none;
}
.footer {
    background: #14532d;
    color: white;
    text-align: center;
    padding: 15px;
    margin-top: 30px;
}
.live-indicator {
    display: inline-block;
    width: 10px;
    height: 10px;
    background: red;
    border-radius: 50%;
    margin-right: 8px;
    animation: blink 1s infinite;
}
@keyframes blink {
    50% { opacity: 0; }
}
.api-status {
    background: white;
    border-radius: 8px;
    padding: 15px;
    margin: 15px 0;
    box-shadow: 0 4px 12px rgba(0,0,0,0.1);
}
.api-key {
    font-family: monospace;
    background: #f8f9fa;
    padding: 5px 10px;
    border-radius: 4px;
    margin: 5px 0;
}
.api-status-success {
    color: #1a863a;
}
.api-status-error {
    color: red;
}
.api-rotation-status {
    background: white;
    border-radius: 8px;
    padding: 15px;
    margin: 15px 0;
    box-shadow: 0 4px 12px rgba(0,0,0,0.1);
}
.api-status-item {
```

```
            display: flex;
            justify-content: space-between;
            padding: 8px 0;
            border-bottom: 1px solid #eee;
        }
        .api-status-item:last-child {
            border-bottom: none;
        }
        .current-api-indicator {
            background-color: #e6f7ee;
            padding: 3px 8px;
            border-radius: 4px;
            font-size: 0.8rem;
            color: #0c622e;
            margin-left: 10px;
        }
    </style>
</head>
<body>
    <header>
        <h1>■ CricLive</h1>
        <p>Live Cricket Scores, Updates & News</p>
    </header>
    <nav>
        <a href="#live-matches">Live Matches</a>
        <a href="#upcoming">Upcoming</a>
        <a href="#news">News</a>
        <a href="#api-status">API Status</a>
    </nav>
    <div class="container">
        <section id="live-matches" class="card">
            <h2><span class="live-indicator"></span> Live Matches</h2>
            <div id="liveMatches"></div>
        </section>
        <section id="upcoming" class="card">
            <h2>Upcoming Matches</h2>
            <div id="upcomingMatches"></div>
        </section>
        <section id="news" class="card">
            <h2>Latest Cricket News</h2>
            <div id="newsFeed">
                <div class="news-item">Loading latest news...</div>
            </div>
        </section>
        <section id="api-status" class="card">
            <h2>API Status & Key Rotation</h2>
            <div class="api-rotation-status" id="apiRotationStatus"></div>
        </section>
    </div>
    <footer class="footer">
        <p>© 2025 CricLive. All Rights Reserved.</p>
    </footer>
    <script>
        const apiKeys = [
            "api_key_1",
            "api_key_2",
            "api_key_3",
            "api_key_4"
        ];
        let currentKeyIndex = 0;
        const baseUrl = "https://api.cricapi.com/v1";
        async function fetchWithRotation(endpoint) {
            let attempts = 0;
            while (attempts < apiKeys.length) {
                const apiKey = apiKeys[currentKeyIndex];
                try {
                    const response = await fetch(`${baseUrl}/${endpoint}&apikey=${apiKey}`);
                    if (!response.ok) throw new Error("API Error");
                    const data = await response.json();
                    updateApiStatus(apiKey, true);
                    return data;
                } catch (error) {
```

```
            updateApiStatus(apiKey, false);
            currentKeyIndex = (currentKeyIndex + 1) % apiKeys.length;
            attempts++;
        }
    }
    throw new Error("All API keys failed");
}
function updateApiStatus(apiKey, success) {
    const statusContainer = document.getElementById("apiRotationStatus");
    const statusItem = document.createElement("div");
    statusItem.className = "api-status-item";
    statusItem.innerHTML = `
        <span class="api-key">${apiKey}</span>
        <span class="${success ? "api-status-success" : "api-status-error"}">
            ${success ? "Working" : "Failed"}
        </span>
        ${apiKeys[currentKeyIndex] === apiKey ? '<span class="current-api-indicator">Current</sp
    `;
    statusContainer.appendChild(statusItem);
}
async function loadLiveMatches() {
    try {
        const data = await fetchWithRotation("currentMatches?offset=0");
        const container = document.getElementById("liveMatches");
        container.innerHTML = "";
        if (data.data && data.data.length > 0) {
            data.data.forEach(match => {
                const div = document.createElement("div");
                div.className = "match";
                div.innerHTML = `
                    <div class="teams">
                        <span>${match.teams[0]} vs ${match.teams[1]}</span>
                        <span class="score">${match.score ? match.score : "TBD"}</span>
                    </div>
                    <div class="status">${match.status}</div>
                `;
                container.appendChild(div);
            });
        } else {
            container.innerHTML = "<p>No live matches at the moment.</p>";
        }
    } catch (error) {
        document.getElementById("liveMatches").innerHTML = "<p>Error loading live matches.</p>";
    }
}
async function loadUpcomingMatches() {
    try {
        const data = await fetchWithRotation("matches?offset=0");
        const container = document.getElementById("upcomingMatches");
        container.innerHTML = "";
        if (data.data && data.data.length > 0) {
            data.data.forEach(match => {
                const div = document.createElement("div");
                div.className = "match";
                div.innerHTML = `
                    <div class="teams">
                        <span>${match.teams[0]} vs ${match.teams[1]}</span>
                        <span>${new Date(match.dateTimeGMT).toLocaleString()}</span>
                    </div>
                    <div class="status">${match.status}</div>
                `;
                container.appendChild(div);
            });
        } else {
            container.innerHTML = "<p>No upcoming matches available.</p>";
        }
    } catch (error) {
        document.getElementById("upcomingMatches").innerHTML = "<p>Error loading upcoming matche
    }
}
loadLiveMatches();
loadUpcomingMatches();
```

```
        setInterval(loadLiveMatches, 60000);
        setInterval(loadUpcomingMatches, 300000);
    </script>
</body>
</html>
```