Creating an advanced social media management system with the features you've described is an ambitious but exciting project. Here's a high-level overview of how you can approach building such a system using HTML, CSS, JavaScript, PHP, and possibly other technologies.

**Key Features and Components**

1. **User Authentication and Multi-Account Management**
   * User login/signup (with support for multiple accounts).
   * OAuth integration for different social media platforms (e.g., YouTube, Twitter, Instagram).
   * Dashboard for managing multiple accounts.
2. **Trend Analysis and Data Tracking**
   * API integration to fetch the latest trends from various social media platforms.
   * Custom date range selection to view trending data (e.g., last 20 days).
   * Filtering options to view channels or accounts based on criteria like views, subscribers, likes, etc.
3. **Channel/Account Analytics**
   * Fetch and display detailed analytics for any social media account.
   * Visual representations (charts, graphs) of data like growth, engagement, demographics, etc.
4. **Content Analysis and Advanced Keyword Generator**
   * Analyze content performance based on engagement metrics.
   * Generate advanced keywords based on Google search trends and platform-specific searches.
   * Content recommendation system based on trending topics and keywords.
5. **Caption Generator and Content Scheduling**
   * AI-powered caption generator for different social media posts.
   * Content calendar and scheduling tool to plan and publish posts.

**Tech Stack**

* **Frontend**: HTML, CSS, JavaScript (possibly using frameworks like React.js or Vue.js for more dynamic interfaces).
* **Backend**: PHP (you can also consider Node.js, Python/Django, or Ruby on Rails for more scalability).
* **Database**: MySQL or PostgreSQL.
* **APIs**: Integration with social media APIs (YouTube Data API, Twitter API, Instagram Graph API, etc.).
* **Libraries/Tools**: Chart.js or D3.js for data visualization, OAuth libraries for authentication.

**Step-by-Step Development Guide**

1. **Setup Environment**
   * Set up a local development environment using XAMPP or MAMP for PHP and MySQL.
   * Initialize a version control system (e.g., Git).
2. **User Authentication and Account Management**
   * Create a user authentication system with login, signup, and password recovery.
   * Implement OAuth for connecting social media accounts.
   * Design a user dashboard for managing multiple accounts.
3. **API Integration and Data Fetching**
   * Integrate with various social media APIs to fetch data (e.g., trends, channel analytics).
   * Store fetched data in your database for efficient querying and analysis.
4. **Frontend Development**
   * Design a responsive user interface using HTML, CSS, and JavaScript.
   * Use libraries like Chart.js to create interactive charts and graphs for displaying analytics.
5. **Content Analysis and Keyword Generation**
   * Implement content analysis tools to evaluate post performance.
   * Develop a keyword generator using external APIs (Google Trends, social media search APIs).
6. **Caption Generator and Scheduling**
   * Integrate AI/ML models for generating captions (e.g., OpenAI's GPT-3 via API).
   * Build a content calendar and scheduling tool to plan and automate posts.
7. **Testing and Deployment**
   * Thoroughly test all features and fix any bugs.
   * Deploy your application to a web server (e.g., using services like AWS, DigitalOcean, or Heroku).