Requirements

The Fashion Trend Analysis Dashboard project has the following requirements:

User Management:

**User Registration and Authentication:**

Users must be able to register accounts with unique usernames and valid email addresses.

User authentication must be implemented to secure user data and provide personalized experiences.

**User Profile Management:**

Users should have the ability to create and update their profiles.

Profile information may include first name, last name, date of birth, gender, address, and contact details.

Trend Data Management:

**Trend Data Entry and Management:**

The system must allow authorized users to add new fashion trends.

Users can edit and update existing trend data, including trend names, descriptions, and associated categories.

**Categorisation of Trends:**

Fashion trends should be categorised into meaningful groups, such as colours, fabrics, styles, themes, designers, locations, and seasons.

Trends can belong to multiple categories simultaneously.

User Interaction Features:

**User Interaction and Engagement:**

Users can like, comment on, and share fashion trends.

Trend popularity metrics, such as the number of likes and comments, should be tracked and displayed.

**Trend Analytics and Reporting:**

The dashboard must provide trend analytics and statistics.

Users should be able to view trend analysis graphs, including trend frequency, category distribution, and user engagement metrics.

Search and Visualisation:

**Search and Filtering:**

Users should have search and filtering options to find specific trends based on keywords, categories, designers, or other attributes.

**Data Visualisation:**

The dashboard should visually represent fashion trends through graphs, charts, and images.

Graphical representations of trend data should be interactive and informative.

Security and Data Integrity:

**Security and Data Protection:**

Implement security measures to protect user data and ensure data integrity.

User authentication and authorisation should be implemented securely.

Backend Development:

**Backend Components:**

Create Java-based backend components, including controllers, services, and DAOs (Data Access Objects), to handle user interactions and data management.

Implement RESTful API endpoints for data retrieval and manipulation.

**DTO (Data Transfer Objects):**

Develop DTOs to facilitate data exchange between the frontend and backend components.

DTOs should represent data entities and provide a structured format for data transmission.

Deliverables

To successfully complete the Fashion Trend Analysis Dashboard project, the following deliverables are expected:

**User Interface (UI):**

Design and develop the web-based dashboard with an intuitive and user-friendly interface.

Ensure responsive design for optimal user experience on different devices.

**Backend Implementation:**

Create Java classes and components for handling user registration, authentication, trend data management, categorization, analytics, and user interactions.

Implement RESTful API endpoints for data retrieval and manipulation.

**Database Setup:**

Develop the database schema to store user profiles, fashion trends, user interactions, and analytics data.

Define appropriate tables, columns, and relationships.

Populate the database with sample data for testing and demonstration.

**Documentation:**

Generate documentation for the REST API, including detailed descriptions of endpoints, request and response formats, and sample API calls.

Create documentation for the project, including an overview, architecture, and user guide.

**Testing and Quality Assurance:**

Write unit tests to ensure the functionality and reliability of backend components.

Conduct integration testing to validate the interaction between frontend and backend components.

Perform UI testing to verify the correctness of user interface features.

**DAO, Controller, and Service Components:**

Implement DAOs, controllers, and service classes to manage data access and user interactions.

Ensure proper handling of requests and responses.

**DTO Implementation:**

Develop DTOs to facilitate data transfer between frontend and backend components.

Ensure that DTOs accurately represent data entities.

**Deployment:**

Deploy the Fashion Trend Analysis Dashboard to a web server or hosting platform for public access.

Configure the deployment environment for security and performance.

**User Training and Support:**

Provide user training materials or resources for using the dashboard effectively.

Offer support channels for user inquiries and issues.

**Project Presentation:**

Prepare a presentation or demonstration to showcase the project's features, functionality, and benefits.

By delivering these components and meeting the specified requirements, the Fashion Trend Analysis Dashboard will be a fully functional and user-centric platform for exploring fashion trends and analytics.