Basic Endpoints

\*\*Trend Endpoints:\*\*

1. \*\*GET /api/trends\*\*: Retrieve a list of all fashion trends.

2. \*\*GET /api/trends/{id}\*\*: Retrieve details of a specific fashion trend by its ID.

3. \*\*POST /api/trends\*\*: Create a new fashion trend. - Fashion Companies and Admin only

4. \*\*PUT /api/trends/{id}\*\*: Update an existing fashion trend by its ID. - Fashion Companies and Admin only

\*\*Designer Endpoints:\*\*

6. \*\*GET /api/designers\*\*: Retrieve a list of all fashion designers.

7. \*\*GET /api/designers/{id}\*\*: Retrieve details of a specific fashion designer by their ID.

8. \*\*POST /api/designers\*\*: Create a new fashion designer profile. - Fashion Companies and Admin only

9. \*\*PUT /api/designers/{id}\*\*: Update an existing fashion designer profile by their ID. - Fashion Companies and Admin only

10. \*\*DELETE /api/designers/{id}\*\*: Delete a fashion designer profile by their ID. - Fashion Companies and Admin only

\*\*User Endpoints (if applicable for user authentication):\*\*

11. \*\*POST /api/register\*\*: Register a new user account.

12. \*\*POST /api/login\*\*: Authenticate a user and generate an access token. – Admin -Extra If possible?

13. \*\*GET /api/users\*\*: Retrieve a list of all users (admin-only access).

14. \*\*GET /api/users/{id}\*\*: Retrieve details of a specific user (admin-only access).

15. \*\*PUT /api/users/{id}\*\*: Update user details.

16. \*\*DELETE /api/users/{id}\*\*: Delete a user account.

\*\*Trend Analysis Endpoints:\*\*

17. \*\*GET /api/trends/{id}/analysis\*\*: Retrieve historical trend analysis data for a specific trend.

18. \*\*POST /api/trends/{id}/analysis\*\*: Create new trend analysis data for a specific trend. – individual user accounts

19. \*\*PUT /api/trends/{id}/analysis/{analysisId}\*\*: Update trend analysis data for a specific trend.

20. \*\*DELETE /api/trends/{id}/analysis/{analysisId}\*\*: Delete trend analysis data for a specific trend.

**Complex Endpoints**

Given the time constraints and complexity considerations, here are some complex endpoints that can be manageable within your project:

1. \*\*Top Trending Trends:\*\*

- Endpoint: /trends/sort/popularity?order={order}`

- Description: Returns a list of the most popular or trending fashion trends based on user likes or comments. This can be determined by tracking user interactions.

2. \*\*Search Trends by Keyword:\*\*

- Endpoint: `/trends/search?keyword={keyword}`

- Description: Enables users to search for fashion trends containing specific keywords or terms in their titles or descriptions.

3. \*\*Trend Statistics:\*\*

- Endpoint: `/trends/statistics`

- Description: Provides statistical data about trends, such as the total number of trends, trends per season, and user engagement metrics.

1. Updates:  
     
   Trend Sorting by Popularity:

* Endpoint: /trends/sort/popularity?order={order}
* Description: Allows users to sort fashion trends by their popularity, either in ascending or descending order.

1. Trend Sorting by Date:

* Endpoint: /trends/sort/date?order={order}
* Description: Enables users to sort fashion trends based on their publication date, either in ascending or descending order.

1. Trend Sorting by Designer:

* Endpoint: /trends/sort/designer?order={order}
* Description: Lets users sort fashion trends by the designer or fashion house associated with them, in ascending or descending order.

1. Trend Sorting by Season:

* Endpoint: /trends/sort/season?order={order}
* Description: Allows users to sort fashion trends by the season or collection they belong to, in ascending or descending order.

Category Endpoints:

1. GET /api/categories: Retrieve a list of all fashion trend categories.
2. GET /api/categories/{id}: Retrieve details of a specific fashion trend category by its ID.
3. POST /api/categories: Create a new fashion trend category. - Admin only
4. PUT /api/categories/{id}: Update an existing fashion trend category by its ID. - Admin only
5. DELETE /api/categories/{id}: Delete a fashion trend category by its ID. - Admin only

Product Endpoints:

1. GET /api/products: Retrieve a list of all fashion products.
2. GET /api/products/{id}: Retrieve details of a specific fashion product by its ID.
3. POST /api/products: Create a new fashion product. - Fashion Companies and Admin only
4. PUT /api/products/{id}: Update an existing fashion product by its ID. - Fashion Companies and Admin only
5. DELETE /api/products/{id}: Delete a fashion product by its ID. - Fashion Companies and Admin only

Complex Endpoints:

1. Top Trending Categories:

* Endpoint: /categories/top
* Description: Returns a list of the most popular or trending fashion trend categories based on user likes or comments. This can be determined by tracking user interactions.

1. Search Categories by Keyword:

* Endpoint: /categories/search?keyword={keyword}
* Description: Enables users to search for fashion trend categories containing specific keywords or terms in their names or descriptions.

1. Category Statistics:

* Endpoint: /categories/statistics
* Description: Provides statistical data about fashion trend categories, such as the total number of categories, categories per season, and user engagement metrics.

1. Advanced Category Sorting:

* Endpoint: /categories/sort?criteria={criteria}&order={order}
* Description: Allows users to sort fashion trend categories dynamically based on criteria such as name, popularity, or season.

1. Get Products by Category:

* Endpoint: /categories/{id}/products
* Description: Retrieve a list of fashion products associated with a specific fashion trend category by its ID.

1. Search Products by Keyword:

* Endpoint: /products/search?keyword={keyword}
* Description: Enables users to search for fashion products containing specific keywords or terms in their names or descriptions.

1. Product Statistics:

* Endpoint: /products/statistics
* Description: Provides statistical data about fashion products, such as the total number of products, products per designer, and user engagement metrics.

1. Advanced Product Sorting:

* Endpoint: /products/sort?criteria={criteria}&order={order}
* Description: Allows users to sort fashion products dynamically based on criteria such as name, price, or designer.

white_check_markeyesraised_hands