Product Search API Design Document

API Structure

```
GET /products/search
Query Parameters:
- q: Search query string (required, min 2 chars)
- limit: Number of results per page (optional, default: 10)
- skip: Number of results to skip (optional, default: 0)
```

Data Schema

```
// Product Object
{
  "id": number,
  "title": string,
  "brand": string,
  "category": string,
  "price": number
}

// Response Schema
{
  "products": Product[],
  "total": number,
  "limit": number,
  "skip": number
}
```

Implementation Details

Search & Pagination Logic

1. **Search Process**:
    - Convert query to lowercase
    - Score products based on matches:
        - Title starts with: +3 points
        - Title contains: +2 points
        - Brand starts with: +2 points
        - Brand contains: +1 point
    - Filter products with score > 0
    - Sort by score (highest first)

2. **Pagination**:

  - Apply after scoring and sorting
  - Slice results using skip and limit
  - Calculate total from all matching products
  - Return paginated results with metadata

Error Handling

- **400 Bad Request**:
  - Query < 2 characters
  - Invalid limit/skip values
- **500 Internal Server Error**:
  - File read errors
  - JSON parse errors

Limitations

- In-memory JSON storage
- Case-insensitive search only
- No fuzzy matching
- No authentication/rate limiting
- Linear search time O(n)

Future Improvements

1. **Search**:

   - Fuzzy matching
   - Full-text search
   - Category/price filtering

2. **Performance**:

   - Database integration
   - Caching
   - Indexing
   - Rate limiting

3. **Features**:

   - Authentication
   - Product management
   - Category/brand management