Bug Report

A bug report is a document or section in our project report that describes problems (bugs) we encountered during development, how they affected the system, and what actions we took to resolve or mitigate them. During the development and testing phases of our project, we encountered several technical and logical issues. Below is a summary of the key bugs, their impacts, and how we resolved them.

- **Broken Image Links After Web Scraping**
  Description: Some product images did not load because the scraped URLs were either relative paths or missing.
  Impact: Affected visual presentation of the product cards.
  Solution: Implemented fallback image handling and added validation during scraping to skip entries with missing URLs.

- **Inconsistent Filtering Results**
  Description: Applying multiple filters (such as price and brand together) produced incorrect or incomplete results.
  Impact: Misled users and reduced search accuracy.
  Solution: Refactored backend filtering logic to combine query conditions using logical AND instead of independent checks.

- **Login Authentication Delay**
  Description: Logging in took longer than expected due to redundant database queries.
  Impact: Slower login experience for users.
  Solution: Optimized login logic and used session-based caching for frequently accessed static data.

- **Comparison Table Crashing on Null Values**
  Description: The comparison feature failed if a product lacked certain specifications.
  Impact: Page crash or layout break during comparisons.
  Solution: Added null checks and used default placeholder values for missing data fields.

- **Comment Section Vulnerable to Script Injection**
  Description: User-submitted comments were not sanitized properly, allowing potential XSS attacks.
  Impact: Security risk due to possible execution of malicious scripts.
  Solution: Sanitized all input fields using middleware and applied HTML escaping on the frontend before rendering.

- **Source Website Unavailable During Maintenance**
  Description: When the original website (from which data was scraped) was under maintenance or offline, our scraper could not retrieve data.
  Impact: Caused failures in updating or displaying product data.
  Solution: Implemented error handling to detect unavailability and display cached or previously stored data from our database. Future improvement may include a fallback backup source.