E-Commerce platform in **consumer electronics**

You have to create the frontend for an e-commerce platform specializing in **consumer electronics** with NextJS. The platform should allow users to:

- Browse a catalog of products.
- Search for specific products using keywords.
- Apply filters to narrow down results based on criteria like category, price range, and rating.
- View detailed information for individual products.

Your solution should emphasize performance, scalability, and user experience.

- Build a **Product Listing Page** with:
  - Grid layout for desktops and single-column layout for mobile devices.
  - Display product details such as name, price, rating, and stock status.
  - Implement pagination or infinite scrolling for seamless product loading.
- Create **Dynamic Product Detail Pages** with:
  - URL structure: /products/[id].
  - High-resolution images optimized using next/image.
  - Detailed product descriptions, price, ratings, and "Add to Cart" functionality.
  - Visual confirmation for "Add to Cart" actions (e.g., toast notifications).
- Implement **Search and Filtering**:
  - Real-time search bar for filtering products by name or keywords.
  - Filters for:
    - Categories (e.g., laptops, smartphones, accessories).
    - Price range (using a slider or input range).
    - Ratings (e.g., 4+ stars).
  - Ensure search and filtering functionalities work together dynamically.
- Use **Static Site Generation (SSG)** with **Incremental Static Regeneration (ISR)**:
  - Pre-generate the product listing page to enhance SEO and load times.
  - Dynamically regenerate pages when new products are added or updated.
  - Apply SSG for individual product detail pages for fast loading and performance.
- Integrate APIs:
  - Use mock APIs or Next.js API routes to fetch product data, handle search, and filter functionality.
  - Simulate external API calls for fetching product details and catalog data.
- Manage State Using:
  - Context API, Redux, or Zustand to track:
    - Active search and filter criteria.
    - Shopping cart details with item count and product specifics.
- Ensure **Responsive Design**:
  - Optimize for desktop and mobile users.
  - Include collapsible search and filter options on mobile devices.
- Apply **Performance Optimizations**:
  - Optimize images using next/image for lazy loading and resizing.
  - Use dynamic imports to lazy-load non-critical components.
  - Prefetch routes to enhance navigation speed.
- Add Optional Enhancements:
  - Wishlist functionality to save favorite products.
  - Sorting options for price, popularity, or ratings.
  - Graceful error handling for empty search results or unavailable products.
- Deliverables:
  - GitHub repository with well-structured code and a README file including:
    - Overview of the application.
    - Setup and usage instructions.
  - Deployed application URL (e.g., on Vercel).
  - Documentation explaining:
    - Implementation approach.
    - Challenges faced and solutions applied.
    - Suggestions for future improvements or additional features.